PLUS: MC-10 Programs

HOT CoCo®

A WAYNE GREEN PUBLICATION
March 1984 USA $2.95

THE MAGAZINE FOR TRS-80 COLOR COMPUTER®, MC-10®, AND DRAGON™ USERS.

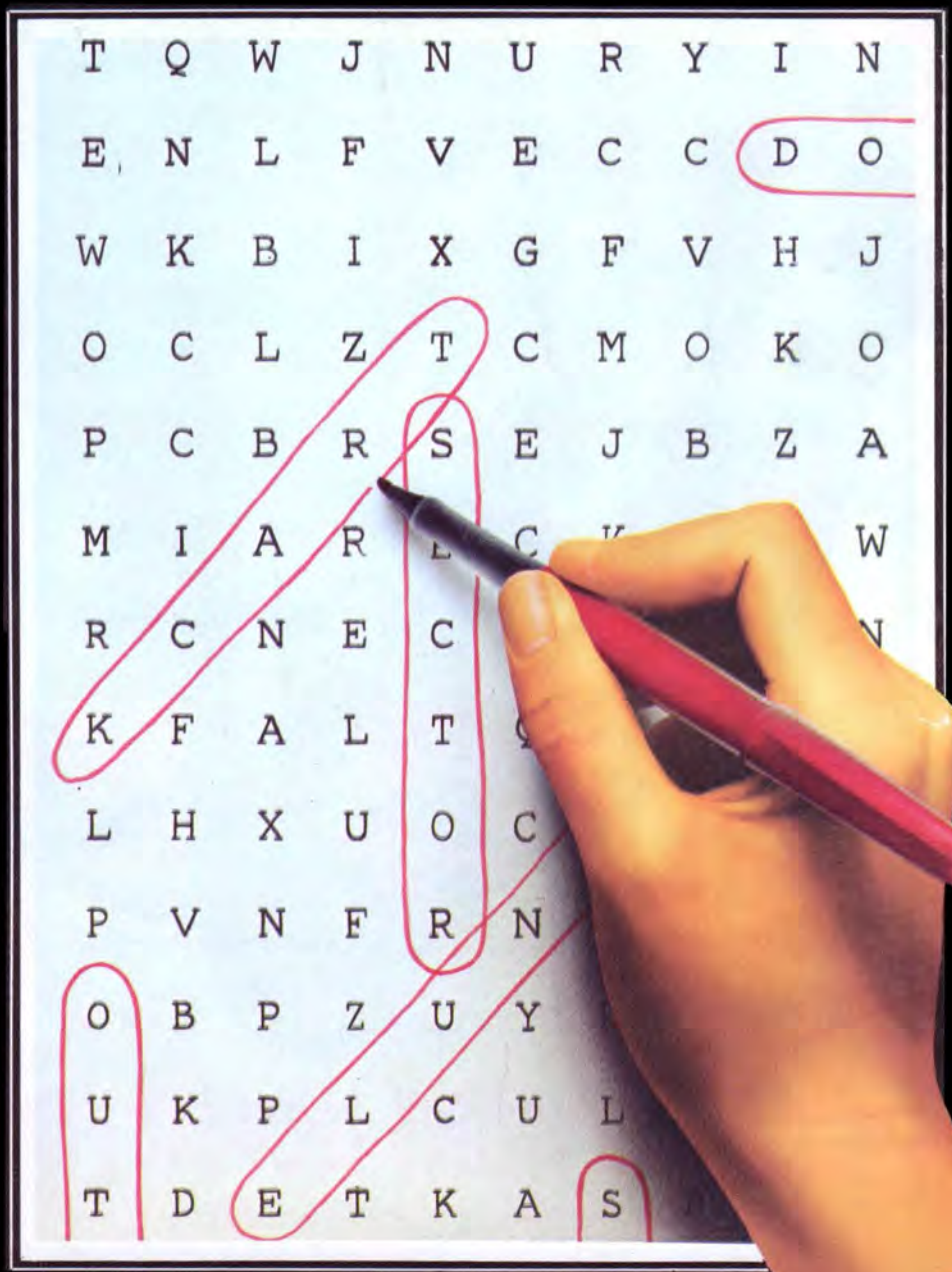
Know Your Disk System

What Do All Those Terms Mean?

Create Your Ow Wordsearch Puzzles

Build a Real-World Interfacing Device

Scott Norman Finds Free FLEX Software



Possum Run:
Great Fun and
Superb Graphics

The Basic Beat Shows ou How To Convert Model I/III/4 to the CoCo

03
0 74470 12067 A

REMEMBER.

ELEPHANT™ NEVER FORGETS.

A full line of top-quality floppies, in virtually every 5 1/4" and 8" model, for compatibility with virtually every computer on the market. Guaranteed to meet or exceed every industry standard, certified 100% error-free and problem-free, and to maintain its quality for at least 12 million passes (or over a lifetime of heavy-duty use).

Contact Dennison Computer Supplies, Inc., 55 Providence Highway, Norwood, MA 02062 or call toll-free 1-800-343-8413. In Massachusetts, call collect (617) 769-8150. Telex 951-624.

✓472

Dennison

ColorMate™

The Color Computer's better half.

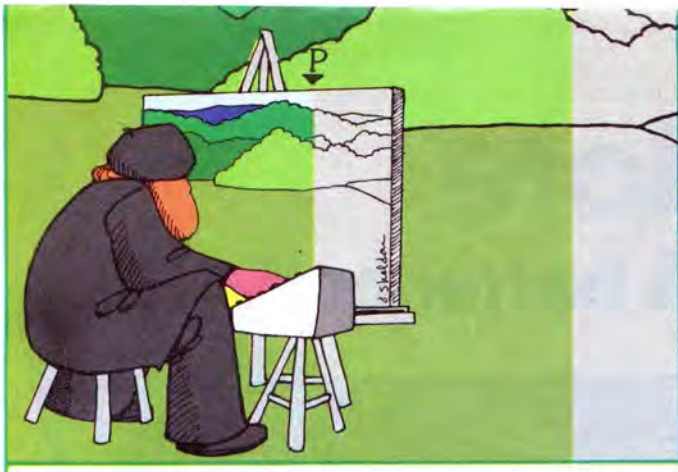
COLORMATE unleashes the power of the Color Computer! It expands any standard CoCo with 16K and floppy disk to allow operation of SDOS, a fast, time proven operating system, and powerful program development tools like the BASIC Compiler. SDOS handles floppies and optional Winchester disk transparently. BASIC provides very fast code, long names, true subroutines and powerful disk file I/O.

Screen above is unretouched photograph of ColorMate display. Disk Extended BASIC is not required.

COLORMATE is $495.00. Includes 400+ pages of documentation. Radio Shack floppy controller and floppy disk drive not included. Word processing, accounting, and Winchester disk drives are available. Write or call for details. Dealer inquiries invited.

COMPUTER SYSTEMS DISTRIBUTORS
P.O. Box 9769
Anaheim, California 92802
(714) 772-1390

✓393



Video Van Gogh—p. 92

HOT CoCo

ARTICLES

This symbol indicates the program's placement on the Instant CoCo loader, available on cassette or disk. See our Instant CoCo ad for details.

Cover art by Eric Ingraham

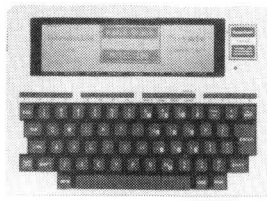
DEPARTMENTS

Vol. 1 No. 10 March 1984

HOT CoCo (ISSN 0740-3186) is published 12 times a year by Wayne Green Inc., 80 Pine St., Peterborough, NH 03458. Phone: 603-924-9471. Second-class postage pending at Peterborough, NH, and additional mailing offices. Subscription rates in U.S. are $25 for one year, $38 for two years, and $53 for three years. In Canada and Mexico, $27.97—one year only, U.S. funds. Nationally distributed by International Circulation Distributors. Foreign subscriptions (surface mail), $44.97—one year only, U.S. funds drawn on a U.S. bank. Foreign subscriptions (air mail), please inquire. In South Africa contact *HOT CoCo*, P.O. Box 782815, Sandton, South Africa 2146. **All subscription correspondence should be addressed to *HOT CoCo*, Subscription Department, P.O. Box 975, Farmingdale, NY 11737. Please include your address label with any correspondence.** Postmaster: Send address changes to *HOT CoCo*, Subscription Services, P.O. Box 975, Farmingdale, NY 11737.

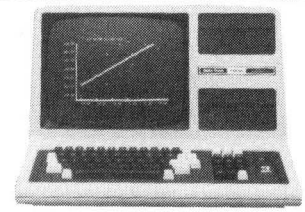
From Computer Plus to YOU . . .

PLUS *after* PLUS *after* PLUS

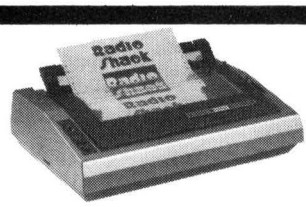


Model 100 8K $679
Model 100 24K $835

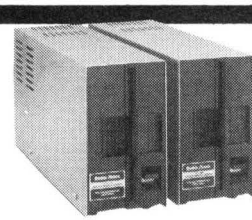
Color Computer II 16K $185
w/16K Ext. Basic $245



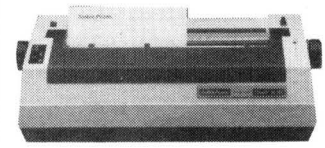
Model 4 16K $849
Model 4 64K
2 Disk & RS232 $1699



DMP120 $395
DMP200 $520



CoCo Drive 0 $329
CoCo Drive 1 $235



DWP210 $629

BIG SAVINGS ON A FULL COMPLEMENT OF RADIO SHACK COMPUTER PRODUCTS

COMPUTERS	
Model 4 Portable 64K w/2 Drives	1525
Color Computer 64K ext. basic	305
Pocket Computer 2	165
Model 12 1 Drive	2699
Model 16B 1Dr 256K	4249
MODEMS	
Hayes Smartmodem II	235
R.S. AC-3	129
R.S. Modem I	89
R.S. Modem II	160
PRINTERS	
Silver Reed EXP500 D.W. Ser.	455
Silver Reed EXP550 D.W. Ser.	665
CGP115	159
CGP220 Ink Jet	545
DMP100	315
DMP420	735
Gemini 10X	315
Delta-10	515
Gemini 15X	399
CITOH Prowriter	375
CITOH Prowriter II	649
Okidata	CALL
Epson	CALL
ETC.	
Disk Drive Controller	139
Extended Basic Kit	69
Botek Ser/Par Conv.	69
64K Ram Chips	75
R.S. Deluxe Keyboard	35.95
Superpro Keyboard	69.95
CCR-81 Recorder	52
R.S. Deluxe Joystick (each)	35.95
R.S. Joysticks (pair)	22
Video Plus (monitor adapter)	24.95
Amdek Color 1+ Monitor	299
SOFTWARE	(Tape Version)
Zaxxon	39.95
The King	26.95
Trap Fall	27.95
Screen Print	19.95
Buzzard Bait	27.95
Devil Assault	27.95
Colorpede	29.95
Juniors Revenge	28.95
Pac Attack	24.95
Block Head	26.95
Froggie	24.95
Lunar Rover Patrol	24.95
Lancer	21.95
Color Zap	9.95
Typing Tutor	19.95
Galagon	24.95
Scott Adams Adventures	19.95
Sea Dragon	34.95
Colorcome	49.95
Telewriter 64	49.95
FHL Flex (disk)	69.95
O-Pak (disk)	34.95
Key-264K	35.95
Elite-Calc	59.95
VIP Writer	59.95
VIP Calc	59.95
VIP Terminal	49.95
VIP Database (disk)	59.95

Order any 2 software pieces listed and take 10% off their listed price.
All R.S. software 10% off list.
Send for complete list.

CALL TOLL FREE
1-800-343-8124

- *LOWEST POSSIBLE PRICES*
- *BEST POSSIBLE WARRANTY*
- *KNOWLEDGEABLE SALES STAFF*
- *TIMELY DELIVERY*
- *SHOPPING CONVENIENCE*

VISA AMERICAN EXPRESS MasterCard

computer plus

P.O. Box 1094
480 King Street
Littleton, MA 01460

SINCE 1973

IN MASSACHUSETTS CALL (617) 486-3193

18

TRS-80 is a registered trademark of Tandy Corp.

HOT CoCo

EDITOR-IN-CHIEF
Michael E. Nadeau
MANAGING EDITOR
Janet Fiderio
REVIEW EDITOR
Mark E. Reynolds
NEW PRODUCTS EDITOR
Cynthia Smith
TECHNICAL EDITORS
Peter Paplaskas,
Guier Wright,
Amee Eisenberg (Instant CoCo)
EDITORIAL DESIGN MANAGER
Susan Gross
EDITORIAL DESIGNER
Susan Hays
LAYOUT EDITORS
Joan Ahern, Phillip Geraci,
Maurelle Godoy, Judy Oliver,
Glenn Suokko
PROOFREADERS
Peter Bjornsen,
Harold Bjornsen,
Robin Florence
RESEARCH ASSISTANT
Celeste Wrenn

PUBLISHER/EDITOR
Wayne Green
VICE PRESIDENT/
GENERAL MANAGER
Debra Wetherbee
VICE PRESIDENT/FINANCE
Roger Murphy
EDITORIAL MANAGER
Jeffrey DeTray
ASSISTANT TO PRESIDENT
Matt Smith
ASSISTANT TO VP/FINANCE
Dominique Smith
DIRECTOR OF MARKETING
AND SALES
David Schissler
CIRCULATION DIRECTOR
William P. Howard
603-924-9471
BULK & NEWSSTAND
SALES MANAGER
Ginnie Boudrieau
1-800-343-0728
ADVERTISING, 603-924-7138
Director: Stephen Twombly
Sales Representatives: Bev Poirier,
F. Lincoln Bedell, Raino E. Wirein
Ad Coordinator: Suzanne DesRochers
PUBLIC RELATIONS
Jim Leonard

DIGRESSIONS

SHOULD YOU BE AFRAID OF BUYING MAIL ORDER?

I have a friend who, as a child, ordered a pet monkey from an ad in a comic book. Several weeks later the monkey arrived—dead. Of course, the ad didn't say that the monkey was *alive*.

You can understand if I tell you my friend has bad feelings about shopping mail order. Perhaps you, too, have had bad experiences shopping by mail, and you hesitate to buy anything for your Color Computer unless you can get it at your local Radio Shack store.

Now, Radio Shack has some great stuff for the CoCo, but for every good product that Radio Shack sells, there are dozens of others a phone call or letter away. And much of the risk involved buying long distance can be reduced if you are aware of the laws and proper ordering etiquette. (This month we've published a short piece on p. 8 detailing your rights as a mail-order consumer and what to do if you think you are getting a raw deal. Please read it.)

The Color Computer "cottage industry," or non-Radio Shack software and hardware vendors, has had an unusually good record in regard to the rest of the mail-order computer business. You just don't hear stories of bogus companies taking people's money and then disappearing in the Color Computer world.

If you have a problem with a mail-order vendor, it's probably one of three types: The vendor advertised something that was not quite finished, and it isn't brought to market at the promised date; your order "fell between the cracks" (vendors are human and sometimes lose track of things); or the vendor went out of business.

The first type is by far the most common, and this is true for the entire microcomputer support industry. Vendors advertise unfinished products because of the desire to beat out the competition. Many times, however, a product is advertised prematurely only because the vendor misjudged the lead time needed to get it out. Meanwhile, the buyer can wait or get his money back.

The second type speaks for itself, and this problem is easily worked out between the buyer and the vendor. The third type of buyer/vendor problem is the most unfortunate: The vendor overestimated the market for his product and couldn't make enough money to continue producing it. Luckily, this is the least-common type of problem.

Many of the CoCo vendors are enthusiasts like yourself. They have sincere interest in your getting the most out of their products. Many are willing to talk you through problems over the phone, and some offer upgrade services for free or at cost. A few even send out newsletters with tips and other newsworthy items.

Most of the vendors I've met are not out to make a killing off Color Computer users. They like what they are doing, and they get a big kick out of someone telling them, "Gee, I really liked your program."

If you have a problem with a vendor, politely try to work it out. You'll get satisfaction 99 percent of the time. If you are in the other 1 percent, let us know about it. Send us any correspondence you have and the problem's history. We'll do all we can to resolve the situation.

If you get exceptionally good service from a vendor, let's hear about that, too. We'll pass along your comments.

One more piece of advice: Don't buy monkeys from ads in comic books.—*M.N.* ■

Go on line in the world's fastest growing technology.

NEW! DATA COMMUNICATIONS TRAINING FROM NRI

Practical training includes computer, modem, test instruments, and access to exclusive NRI communications network.

Satellites . . . microwave . . . fiber optics . . . dedicated land lines. Suddenly the world is communicating in a new and different way, via digital data systems. People talking to computers . . . computers to computers . . . information is stored, retrieved, and relayed in nanoseconds.

Industry, opportunities to triple

Data and telecommunications is already a $150 billion industry and is expected to triple over the next five years. One typical company has grown from $85 million to $650 million . . . a 765% growth since 1978 alone. The need for qualified technicians to install, maintain, and service this enormous investment in high-tech equipment is tremendous even now. Opportunities and salaries can go nowhere but up and up.

NRI will train you at home

You can learn at home in your spare time to become a data communications technician with NRI at-home training. NRI will start you with the basics, build upon your knowledge with easy-to-follow, bite size lessons to take you into the world of digital data communications. You'll learn what it takes to work on satellite, microwave, fiber optic, and telephone data links.

And you'll learn at your own comfortable pace, without classroom pressures or evenings away from your family. Over the past 70 years, NRI has taught the latest high-tech skills to almost 2 million students to become the world's largest and most successful school of its kind.

Hands-on training includes computer, modem, breakout box and much more

NRI takes you far beyond "book learning." As part of your course, you receive plenty of practical hands-on training that gives you real-world skills. You get the Radio Shack Color Computer, with 16K memory to teach you the systems and language of data communications plus you get an operating modem to let you tie in with world-wide communications networks.

You build your own RS-232C interface breakout box, an indispensable installation and trouble-shooting instrument you'll use throughout your career. You receive a professional digital multimeter and the NRI Discovery Lab, where you construct solid-state circuits and demonstrate practical applications of the theory you've learned.

Exclusive NRI data network

You'll learn what data communications is all about by actually becoming part of an operating network. You'll go on line to "talk" to your instructor, take your final exam by computer link, communicate with other NRI students and leave messages on the NRI "bulletin board."

As part of your course, you'll also receive membership in THE SOURCE [sm], a regular $100 value. A phone call ties you into computers loaded with instant news, stock quotes, electronic mail, educational programs, games, even discount shopping and travel reservations.

Move into the future, send for Free Catalog

You can't find training like this anywhere else . . . only NRI trains you at home for an exciting and rewarding career in the brilliant new world of Data Communications. Mail the coupon right now for our big catalog of high-tech electronic careers showing all the equipment you get, detailed lesson descriptions, and career opportunities. Look it over and decide where you want your future to grow. Act now. There's a real need for trained data communications technicians.

Training includes all this equipment you keep . . . 16K computer, modem, breakout box, digital multimeter and the exclusive NRI Discovery Lab.

TRS-80 is a trademark of the Radio Shack division of Tandy Corp.
SM a service mark of Source Telecomputing Corp., a subsidiary of the Reader's Digest Association, Inc.

NRI McGraw-Hill

NRI Schools
McGraw-Hill Continuing Education Center
3939 Wisconsin Avenue
Washington, D.C. 20016
We'll give you tomorrow.

All Career courses approved under GI bill.
☐ Check for details

CHECK ONE FREE CATALOG ONLY

- ☐ Data Communications
- ☐ Computer Electronics with Microcomputers
- ☐ Color TV, Audio, and Video System Servicing
- ☐ Electronics Design Technology
- ☐ Digital Electronics
- ☐ Communications Electronics • FCC Licenses • Mobile CB • Aircraft • Marine
- ☐ Industrial Electronics
- ☐ Basic Electronics
- ☐ Small Engine Servicing
- ☐ Appliance Servicing
- ☐ Automotive Servicing
- ☐ Auto Air Conditioning
- ☐ Air Conditioning, Heating, Refrigeration, & Solar Technology
- ☐ Building Construction

Name (Please Print) Age

Street

City/State/Zip

Accredited by the Accrediting Commission of the National Home Study Council 194-034

Mail-Order Savvy

One of the ways *HOT CoCo* saves you money is through the many mail-order vendors for whom we run ads. By leafing through our pages, you can come up with some good deals on software and hardware.

Often, our readers have praised various vendors' customer-service policies. Sometimes, though, that isn't the case.

The Federal Trade Commission passed the Mail-Order Rule in 1975 to protect consumers from mail-order fraud. If you plan to order anything through the mail, you should know your rights.

If you've sent in payment, you must receive the merchandise when the vendor promised it.

If the vendor has not stated a specific delivery date, then he must ship your merchandise no later than 30 days after receiving your order. If you don't get your purchase shortly after the 30 days, you can cancel your order and get your money back.

A vendor must notify you if he can't meet the promised delivery date (or the 30-day limit). He must also tell you when he can ship the merchandise and give you the option of canceling your order for a full refund or agreeing to wait for the new shipping date. He must give you a free way to return your answer, as by a stamped card or envelope. However, if you don't answer, it means you accept the shipping delay.

When you cancel a prepaid order (unless you paid via credit card), the vendor must mail you a refund within seven business days. If there is a refund delay, the company must obtain your express consent.

If you cancel a mail order charged on your credit card, the vendor must credit your account within one billing cycle after receipt of your request. This rule does not apply to certain purchases, such as the following: mail-order photo finishing, orders for seeds and plants, magazine subscriptions and other deliveries in a series (except for the initial shipment), C.O.D. orders, and credit orders when you do not pay before the company mails the merchandise. Also, the rule does not generally apply if you order an item by phone, perhaps when using a vendor's toll-free 800 number.

So, if you have a problem, what should you do? First, contact the mail-order company. Be ready to give them all pertinent information, such as your order number, credit-card number, check number, order date, and the exact name you used when ordering. You'll make solving potential problems much easier if you keep records of all this information from the beginning.

If you've contacted the company and still aren't satisfied, contact the following organizations (and keep a record of each communication):

- Your local or state consumer protection office or Better Business Bureau.
- The consumer protection agency nearest to the vendor.
- Your local postmaster. Ask for the name and address of the appropriate postal inspector in charge.
- The book, magazine, or newspaper publisher that carried the advertisement. Publishers often try to resolve problems between readers and advertisers.
- The Direct Mail/Marketing Association, Mail-Order Action Line, 6 E. 43rd St., New York, NY 10017.
- The Federal Trade Commission, 6th St. and Pennsylvania Ave., Washington, DC 20580. The FTC doesn't resolve individual disputes, but your information can help them establish policies to protect consumers in the future. ■

If Our Programs Don't Work

Having trouble entering our listings from the magazine? Here are a few tips that might help.

First, we print all our Basic listings in the CoCo's 32-column format. This means that each line should appear the same on the screen as it does in the magazine. If a line on your screen does not match the same line in the magazine, reread what you typed; you might have made an error.

Second, make sure the program is for your computer. Read the System Requirements box. The information in this box represents the minimum system configuration needed to run that particular program. Also, read the article thoroughly before typing in the program. Sometimes the article contains instructions vital to making the typed-in listing work. For instance, some CoCos will not accept the high-speed POKE (POKE 65495,0). The article for a program using this POKE will tell you to change those POKEs to 65494,0 if your computer will not work at the faster speed.

Some CoCos are sensitive to spacing in the program lines. Occasionally a computer will read a line such as FORR=1TO20 incorrectly, interpreting the FOR not as a keyword, but as a variable. If you've removed spaces from a program listing to save space, and that program will not work, reinsert those spaces.

If everything is okay so far, check the published listing with what you've typed. Common typing errors include confusing a zero with the letter O, a one with the letter I, or a colon with a semicolon. DATA statements are particularly tricky because of the long lists of numbers. Be very careful with these.

Assembly listings usually require an editor/assembler to enter them into your CoCo. The two most common editor/assemblers are Radio Shack's EDTASM+ and The Micro Works' SDS80C. An Assembly listing assembled using the SDS80C will probably not run under EDTASM+. You can hand-assemble Assembly listings using a short Basic listing such as that found on page 135 of the November 1983 *HOT CoCo*. Hand-assembly is a tedious task best left to more experienced users.

If all the above fails, send us a printout or a detailed description of the problem you experience along with any error messages. We'll try to work it out for you. We cannot help you if you have modified the original program in any way. ■

PRODUCTION
Director: Nancy Salmon;
Michael Ford, Marjorie Gillies, Donna Hartwell, Anne Rocchio, Kenneth Sutcliffe, Theresa Verville, Robert M. Villeneuve, Phyllis Pittet, Lynne Simonson
Ad Coordinators: Patricia Bradley, Paula Ramsey; Assistant: Jean Southworth
Advertising Production: Fiona Davies, Bruce Hedin, Scott Philbrick, Jane Preston
CHIEF COPYWRITER
Steve Tripp
HOT CoCo, Gail Morrison
PHOTOGRAPHY
Supervisor: Thomas Villeneuve;
Sandra Dukette, Laurie Jennison, Nathaniel Haynes, Sturdy Thomas
TYPESETTING
Supervisor: Sara Bedell; Darlene Bailey, Marie Barker, Prem Gongaju, Lynn Haines, Cynthia Letourneau, Kimberly Nadeau, Debbie Nutting, Lindy Palmisano, Heidi Thomas, Susan Weller
DESIGN
Manager: Joyce Pillarella;
Design Consultant: Dion Owens
CREATIVE DIRECTOR
Christine Destrempes

The left bracket, [, replaces the up arrow used by Radio Shack to indicate exponentiation on our printouts. When entering programs published in *HOT CoCo*, you should make this change.

Article submissions from our readers are welcomed and encouraged. Inquiries should be addressed to: *HOT CoCo* Submissions Editor, 80 Pine Street, Peterborough, NH 03458. Include an SASE for a copy of our writer's guidelines. Payment for accepted articles is made at a rate of approximately $50 per printed page; all rights are purchased. Authors of reviews should contact the *HOT CoCo* Review Editor, 80 Pine Street, Peterborough, NH 03458.

Problems with Subscriptions: Send a description of the problem and your current and/or most recent address to: *HOT CoCo*, Subscription Department, P.O. Box 975, Farmingdale, NY 11737.

Change of Address: Send old label or copy of old address and new address to: *HOT CoCo*, P.O. Box 975, Farmingdale, NY 11737. Please give eight weeks advance notice.

Dealers: Contact Ginnie Boudrieau, Bulk Sales Manager, *HOT CoCo*, Pine St., Peterborough, NH 03458. (800) 343-0728.

Problems with Advertisers: Send a description of the problem and your current address to: Magazine, Rt. 101 & Elm Street, Peterborough, NH 03458. ATTN.: Rita B. Rivard, Customer Service Manager. If urgent, call 1-800-441-4403.

HOT CoCo is a member of the CW Communications/Inc. group, the world's largest publisher of computer-related information. The group publishes 42 computer publications in 18 major countries. Nine million people read one or more of the group's publications each month. Members of the publication group include: Australia: *Australasian Computerworld, Micro Magazine*; Argentina: *Computerworld/Argentina*; Brazil: *DataNews, MicroMundo*; Denmark: *Computerworld/Danmark, MikroData*; France: *Le Monde Informatique*; Germany: *ComputerWoche, MicroComputerWelt, PC Welt*; Italy: *Computerworld Italia*; Japan: *Computerworld Japan*; Mexico: *Computerworld/Mexico*; Norway: *Computerworld Norge, MikroData*; People's Republic of China: *China Computerworld*; Saudi Arabia: *Saudi Computerworld*; Spain: *Computerworld/Espana, MicroSistemas*; Sweden: *ComputerSweden, MikroDatorn, Min Hemdator*; United Kingdom: *Computer Management, Computer Business Europe*; United States: *Computerworld, HOT CoCo, inCider, InfoWorld, ISO World, Microcomputing, PC World, 80 Micro, RUN,* and *jr.*

WORKBASE

MORE THAN JUST A DATABASE

- CREATE
- UPDATE
- REORGANIZE

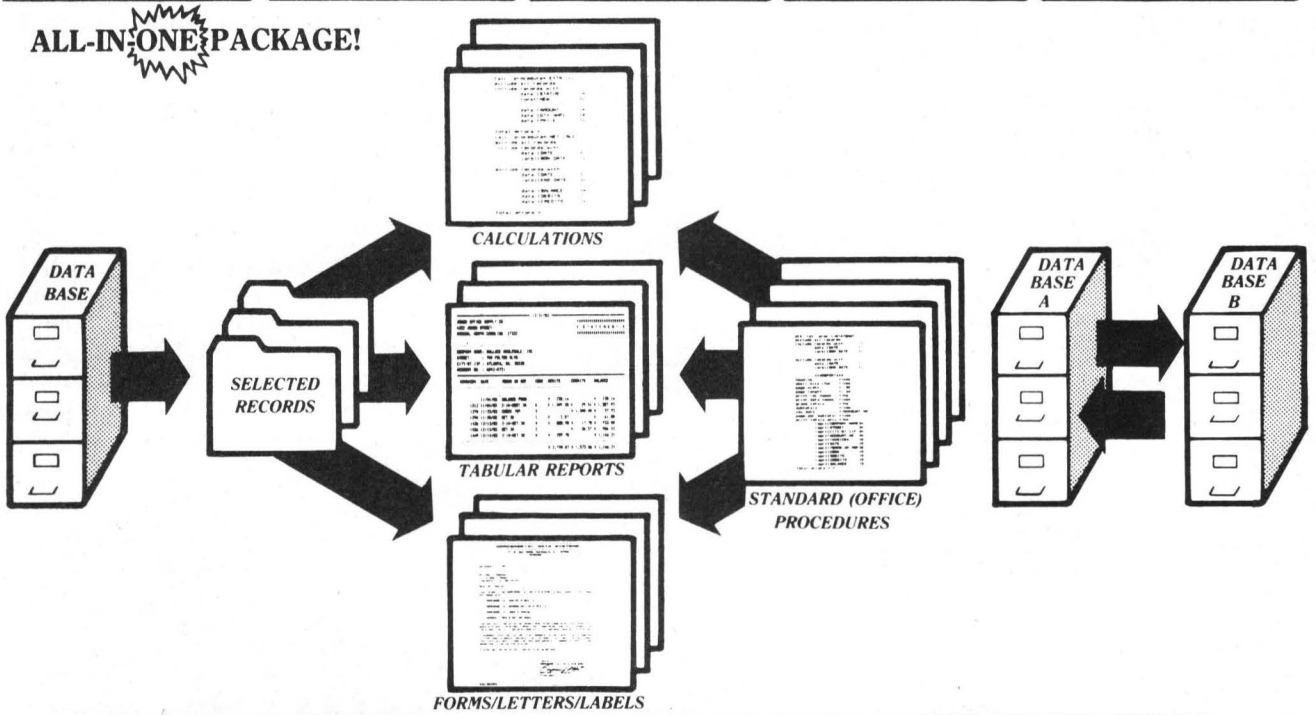
- SELECT AND
- SORT

- CALCULATIONS
- REPORTS

- DEFINE & SAVE CALCULATIONS AND REPORTS

- MOVE DATA BETWEEN FILES

ALL-IN-ONE PACKAGE!



- CALCULATOR MODE
- ADD, DELETE, OR REARRANGE DATA FIELDS

- SORT ASCENDING OR DESCENDING
- SELECT OR SORT USING ANY DATA FIELD
- MULTIPLE SELECT MODES

- DEFINE SEVERAL CALCULATIONS OR REPORTS WITH SELECTION CRITERIA
- USE CALCULATION RESULTS AS SELECTION CRITERIA OR OTHER CALCULATIONS
- MERGE DATABASE WITH TEXT

- CHANGE SELECTION/CALCULATION VARIABLES OR ENTER DATA AT EXECUTION TIME
- LOAD & EXECUTE USING ANY FILE
- ALSO SAVE SORT AND SELECTION CRITERIA

- MERGE FILES OR RECORDS
- COPY FILES FOR BACK-UP
- SEVERAL UTILITIES

WORKBASE™ DOES NOT REQUIRE PROGRAMMING, SPECIAL SOFTWARE, OR ANY HARDWARE MODIFICATIONS.

WORKBASE™ COMES COMPLETE WITH MANUAL, TUTORIAL, DEMONSTRATION FILES, MASTER DISK AND WARRANTY.

WORKBASE™ I $ 95.00 400 RECORDS: REQUIRES 32K COLOR COMPUTER & 1 DISK DRIVE.

WORKBASE™ II $115.00 1200 RECORDS: REQUIRES 32K COLOR COMPUTER & 2 DISK DRIVES.

*** PRICE INCLUDES SHIPPING & HANDLING WITHIN U.S.A. *** NC RESIDENTS ADD 4% SALES TAX. SHIPPED FROM STOCK WITHIN 24 HRS. ***

TO ORDER SEND CHECK OR MONEY ORDER TO:

CREDIT CARD ORDERS:
CALL TOLL FREE 1-800-334-0854 (EXT 887)
WITHIN NC CALL 1-800-672-0101 (EXT 887)

WORKBASE™ DATA SYSTEMS
P.O. BOX 3448
DURHAM, N.C. 27702

✓359

WORKBASE™ IS A TRADEMARK OF WORKBASE DATA SYSTEMS, A SUBSIDIARY OF SMALL BUSINESS SYSTEMS OF DURHAM, N.C. - 1 (919) 544-5408, TRS-80 IS A TRADEMARK OF TANDY CORP.

✓See List of Advertisers on page 130

TRS-80® Color

Come to Radio Shack

16K Standard Color Computer 2

159.95

26-3026

Was $239.95 in Cat. RSC-10

USE YOUR Radio Shack CITILINE

16K Extended Color Computer 2

199.95

26-3027

Was $319.95 in Cat. RSC-10

Great Savings. We've cut the price of our most popular family computer by 33% over last year's price! With its compact, white case and electric typewriter-quality keyboard, the TRS-80 16K Standard Color Computer 2 is truly your best buy!

Ready-to-Use. Just attach your Color Computer to any TV and plug in a Program Pak™ to set up a household budget, track investments, help your kids sharpen math and reading skills—or play exciting space, sports and logic games!

Learn to Program. It's easy to teach yourself to program with the help of our entertaining beginner's manual. Simplified commands make it easy to program in eight vivid colors and sound. In no time at all you'll be producing

TP-10 Thermal Printer

99.95

26-1261

USE YOUR Radio Shack CITILINE

CGP-115 Color Graphics Printer

199.95

26-1192

Prices apply at participating Radio Shack stores and dealers. TV not included.

Computers at New Low Prices!

for Printers, Software and More for Your Color Computer

great-looking drawings, charts and diagrams! The Standard BASIC language includes data and string handling capabilities, dimensioned arrays, math functions and 9-digit accuracy.

Easy to Expand. The Color Computer expands when you're ready with a cassette recorder, joysticks, color mouse controller, printer and telephone modem. You can add more memory and up to four disk drives, too!

Extend Your Abilities. Want to create high-resolution graphics or write sophisticated programs? Then the 16K Extended Color Computer 2 is the answer. You'll find it's easy to create elaborate geometrical shapes and drawings—even simple animation! You can even teach yourself how to create your own arcade-type games. You get all the great features of the 16K Standard version, plus Extended Color BASIC's powerful programming features, such as PEEK, POKE and USR commands, multi-character variable names, string arrays up to 255 characters, full-featured editing and tracing. You also get both the Standard and Extended BASIC tutorial manuals, so you can become an expert programmer in no time at all.

CGP-115 Color Graphics Printer. A perfect match for the Color Computer. It easily prints charts, graphs, computer-generated "doodles," programmed pictures and more in red, blue, green and black on 4 1/2"-wide paper. Commands built into this "smart" printer simplify drawing and plotting. Text mode prints 40 or 80 characters per line at 12 characters per second. Software controls additional character sizes and rotation. The CGP-115 uses easily replaceable ink cartridges. Measures only 2 15/16 × 8 1/4 × 8 1/2" and weighs just 1 3/4 pounds!

TP-10 Thermal Printer. The affordable way to get program listings, text and data printouts. Prints non-color graphics, too. And TP-10's thermal operation is so quiet you'll hardly know it's there! Text mode prints 32 characters per line at 30 characters per second. A special repeat function makes graphics programming easier than ever! Uses 4 1/8"-wide paper. Measures just 3 × 8 × 5".

Available Nationwide. Find out how your family can benefit from owning the exciting Color Computer 2. Visit your nearby Radio Shack Computer Center, participating store or dealer today and ask for a "hands-on" demonstration.

New! Reactoids* Tests Your Skill

19.95 26-3092

Stop the Atomic Threat

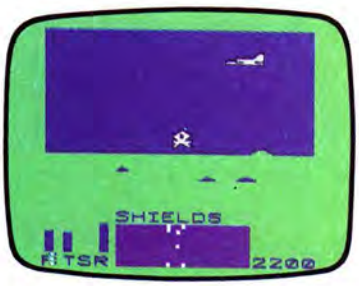
Meltdown at a fusion reactor! Take control of the reactor's computer system to try and contain the wandering atoms! A real-time game to test your skill and dexterity.

New! Dungeons of Daggorath™ —The Ultimate in Adventure

29.95 26-3093

Find the Hidden Treasures

You're pitted against a succession of awesome beasts. Each victory brings you closer to your enemy—the wizard!



New! Star Blaze* for Galactic Fun

19.95 26-3094

"Captain, What Are Your Orders?"

Your mission: protect the Milky Way. Seek and destroy alien vessels, then check the radar for the next wave. There's no let-up in the excitement.

New! Baseball* for Great On-the-field Sports Fun

24.95 26-3095

Hit 'Em! Field 'Em!

You're in full control of this exciting sports simulation. Lets you bring the World Series home!

Your Nearby Radio Shack Has the Latest in Color Software

*Joysticks required

Send me your free TRS-80 Computer Catalog.

Mail To: Radio Shack, Dept.
300 One Tandy Center, Fort Worth, Texas 76102

NAME _____

ADDRESS _____

CITY _____ STATE ____ ZIP _____

TELEPHONE _____

Radio Shack®

The biggest name in little computers®

A DIVISION OF TANDY CORPORATION

The HJL-57 Keyboard

Compare it with the rest. Then, buy the best.

If you've been thinking about spending good money on a new keyboard for your Color Computer, why not get a good keyboard for your money?

Designed from scratch, the HJL-57 Professional Keyboard is built to unlock ALL the potential performance of your Color Computer. Now, you can do real word processing and sail through lengthy listings...with maximum speed; minimum errors.

At $79.95, the HJL-57 is reasonably priced, but you can find other CoCo keyboards for a few dollars less. So, before you buy, we suggest that you compare.

Compare Design.

The ergonomically-superior HJL-57 has sculptured, low profile keycaps; and the three-color layout is identical to the original CoCo keyboard.

Compare Construction.

The HJL-57 has a rigidized aluminum baseplate for solid, no-flex mounting. Switch contacts are rated for 100 million cycles minimum, and covered by a spill-proof membrane.

Compare Performance.

Offering more than full-travel, bounce-proof keyswitches, the HJL-57 has RFI/EMI shielding that eliminates irritating noise on displays; and four user-definable function keys (one latchable), specially-positioned to avoid inadvertent actuation.

Free Function Key Program

Your HJL-57 kit includes usage instructions and decimal codes produced by the function keys, plus a free sample program that defines the function keys as follows: F1 = Screen dump to printer. F2 = Repeat key (latching). F3 = Lower case upper case flip (if you have lower case capability). F4 = Control key; subtracts 64 from the ASCII value of any key pressed. Runs on disc or tape; extended or standard Basic.

Compare Installation.

Carefully engineered for easy installation, the HJL-57 requires no soldering, drilling or gluing. Simply plug it in and drop it right on the original CoCo mounting posts. Kit includes a new bezel for a totally finished conversion.

Compare Warranties.

The HJL-57 is built so well, it carries a full, one-year warranty. And, it is sold with an exclusive 15-day money-back guarantee.

Compare Value.

You know that a bargain is a bargain only so long as it lasts. If you shop carefully, we think you will agree...The HJL-57 is the last keyboard your CoCo will ever need. And that's **real** value.

Order Today.

Only $79.95, the HJL-57 is available for immediate shipment for either the original Color Computer (sold prior to October, 1982) or the F-version and TDP-100 (introduced in October, 1982), and the new 64K CoCo. **Now also available for CoCo 2.**

Order by Phone Anytime

716-235-8358

24 hours, 7 days a week

Ordering Information: Specify model (Original, F-version, or CoCo 2). Payment by C.O.D., check, MasterCard or Visa. Credit card customers include complete card number and expiration date. Add $2.00 for shipping ($3.50 for Canada). New York state residents add 7% sales tax.
Dealer inquiries invited. For dealer information in Eastern U.S. and Canada, call collect: 617-586-7614, Advanced Computer Services (distributor), 74 Plain Street, Brockton, MA 02401.

440

HJL PRODUCTS INC.
955 Buffalo Road • P.O. Box 24954
Rochester, New York 14624

Feedback

Assembler Frustration

I enjoy the Assembly-language programs you publish and hope for more. However, I am continually frustrated by the differences between EDTASM+ and other assemblers, such as SDS80C.

When you publish articles like "Smashout" (*HOT CoCo*), November 1983, p. 80), I wish you would explain how to assemble them on EDTASM+, or at least let us know that the program requires an SDS80C assembler.

I would also appreciate an article explaining the differences and similarities among assemblers.

Joel A. Sherman
Watertown, CT

Sorry about our oversight. We try to be as specific as possible in our System Requirements box. Your article suggestion is a good one. We'll see what we can do. —eds.

Them Ol' PMODE 4 Blues

Many programmers seem to be using the PMODE 4 color artifacts lately, as does Radio Shack in Sands of Egypt. However, PMODE 4 does not work on TV sets in countries such as Australia and the UK that use the PAL TV system. This color artifact just results in black and white, or at best, blue and white.

Down here, games such as Donkey King and Space Shuttle are pale and colorless, but those such as Trapfall and Whirlybird Run that use different high-resolution color techniques work well.

I hope the programmers out there take note. I'd like to be able to see more of the color games on my CoCo.

Tony R. Davidson
Brisbane, Australia

Have You Hugged Your *HOT CoCo* Today?

I want to thank you for a great magazine—it's fantastic. My 15-year-old son and I have learned so much from each article.

He has taken computer programming in school, where they have an Apple II. He doesn't find it any problem to switch from machine to machine, but we know the CoCo is the best.

Being a homemaker, I don't have a lot of time to spend learning and programming, but watching my son teaches me the basics.

The checksums that Howard Batie put into his Galaxy Trek Adventures are a big help—I wish all programs had them. I make so many typos and they make it so much easier to find the errors. I'd like to thank Mr. Batie for some wonderful games.

Another fine feature is Elmer's Arcade. I'd like to program Robert Dukette's drawing in the October issue as a title page for each of Richard Ramella's games. Perhaps someone has already done it, but it would require more memory than just 4K.

I also follow the Basic Beat each month. Again, thanks for an exciting magazine.

Jeanne Raynor
Ford, WA

Clubhouse

Dayton, OH

We have recently formed the Dayton CoCo Users' Group. Interested parties should contact me for information.

Dayton CoCo Users' Group
c/o Joe Evans, Jr.
609 Applehill Drive
West Carrollton, OH 45449

Southern Illinois

The Southern Illinois CoCo Klub (SICK) meets monthly at a member's home or business. We announce our meetings in our monthly newsletter.

SICK
c/o John Dalhaus
2815 Orchard Court
Highland, IL 62249

Better Resolution

I built the monochrome monitor ("Monochrome Monitor Driver," *HOT CoCo*, July 1983, p. 36), which I find very satisfactory, but I would appreciate better resolution. How about publishing an article on building an 80-column card (if such a thing is possible), preferably internally connected so the ROM-pack port remains free for my Super Color Writer ROM pack.

Keep up the good work; I enjoy *HOT CoCo*.

David Elliot
Toronto, Ontario

On-Line

Falcon Color-80

The Falcon Color-80 is a new 24-hour bulletin board system (BBS) in California. We're a CoCo board, but we welcome all computer users.

In addition to an electronic mail section, we have full upload and download capabilities. We welcome comments.

Craig, Keith, and Dan Daniel, Sysops
Fairfield, CA
707-437-3663 (BBS)

CoCo Corner BBS

The CoCo Corner BBS operates 24 hours a day at 300–1,200 baud and supports uploading and downloading.

Jim LeDoux, Sysop
Santa Barbara, CA
805-687-9400 (BBS)

Creme De CoCo

Creme De CoCo is a 24-hour, 7-day, BBS exclusively for the Color Computer.

Eric Thred, Sysop
Blis, IL
312-597-8485 (BBS)

Dang It, Let's Get It Right

The very chagrined staff of *HOT CoCo* would like to apologize to our reader's for the inadvertant sneak preview of the February Elmer's Arcade program listing in January. The correct listing, Dang It, appears below. (Sorry Richard, but Elmer made us do it.)

```
100 REM * DANG IT * TRS-80 EXTEN
DED COLOR BASIC
110 REM * ELMER'S ARCADE * JAN.
'84 * RICHARD RAMELLA
120 CLS
130 FOR A=1 TO 200
140 PRINT CHR$(34+RND(10));
150 B=RND(10)
160 IF B=5 THEN PRINT " DANG IT!
 ";: SOUND RND(8)*13,1
170 NEXT A
180 FOR A=1 TO 500
190 NEXT A
200 CLS(0)
210 SC=0
220 E=30
230 G$=CHR$(128)
240 R=1024
250 FOR A=32 TO 62
260 POKE R+A,207
270 POKE R+A+448,207
280 IF A=32 OR A=62 THEN FOR X=A
 TO A+448 STEP 32: POKE R+X,207:
NEXT X
290 NEXT A
300 PRINT @ 0,"SCORE";STRING$(25
," ");
310 PRINT @ 20,"HIGH";SG;
320 P$=CHR$(207)+STRING$(3,191)+
CHR$(207)
330 PRINT @ 480,STRING$(31,207);
340 PRINT @ 449,STRING$(29,128);
350 PRINT @ 0,"SCORE";SC;
360 E=E-.10
370 A=RND(7)
380 IF A=5 GOTO 370
390 A=A*16
400 H$=CHR$(143+A)
410 Z$=G$+H$+G$
420 B=RND(2)
430 IF B=1 THEN FOR C=506 TO 480
 STEP -1 ELSE FOR C=480 TO 506
440 PRINT @ C,P$;
450 PRINT @ C-31,Z$;
460 FOR T=1 TO E
470 NEXT T
480 IF INKEY$<>"" GOTO 520
490 NEXT C
500 SC=SC-1
510 GOTO 330
520 M=C-30
530 PRINT @ M,H$;
540 FOR K=M TO 63 STEP -32
550 PRINT @ K,H$;
560 IF PEEK(R+K-32)<>128 GOTO 59
0
570 PRINT @ K,G$;
580 NEXT K
590 L=A+143
600 IF K>415 THEN PRINT @ K,H$;:
 GOTO 300
610 IF PEEK(R+K-32)=L OR PEEK(R+
K-1)=L OR PEEK(R+K+1)=L GOTO 650
620 SC=SC+1
630 SOUND RND(8)*13,1
640 GOTO 330
650 IF SC>SG THEN SG=SC
660 PRINT @ 480,"GAME OVER";
670 SOUND 1,1
680 SOUND 100,1
690 IF INKEY$<>"" GOTO 200
700 GOTO 670
710 END
```

High Scores

Eds. note—We received several high scores following our query last November. If the response continues, we'll run them as a regular feature. Keep them coming in.

So far, the champions are as follows:

Name	Game	Score
Ray Gallantry Brampton, Ontario	Bloc Head	64,275
	Keys of the Wizard (Level 1)	632
Dan Shargel Arroyo Grande, CA	Whirlybird Run	78,450
Doug Burke Kenora, Ontario	Lancer	117,700
Greg Burke Kenora, Ontario	The King	156,900
	Colorpede	1,376,460
	Doodle Bug	1,470,200
	Zaksund	556,780
	Pyramid	200
	Ninja Warrior	74,500
	Frog Trek	14,700
	Ghost Gobbler	64,500
M.A. Brickles Allen Park, MI	Defense	103,660
	Scarfman	121,600
	Death Trap	60,838
	The Frog	20,340
	Solo Pool	80
Michael E. Nadeau Peterborough, NH	Buzzard Bait	138,000
Mark E. Reynolds Bennington, NH	Mudpies	113,800

Help!

Ray Gallantry also needs to know how to get out of the maze and obtain a spell in Madness and the Minotaur and how to load the gun in Keys of the Wizard. Any suggestions?

Life in the Fast Lane

I'm an avid game-player who loves car games. If anyone out there has written a good one, please send it in.

Scott March
6 Debra Cresent
Barrie, Ontario L4N 3T1

Coke and CoCo

Many of your advertisers are making a serious mistake when they show a computer or peripheral sitting with a cup of coffee beside it. A cup of liquid beside your machine is a disaster waiting to happen.

As a TV engineering supervisor, I work with electronic devices every day, and I'm well aware of the hazards presented by coffee, tea, or soda pop in a computer. Not only can they short out a unit that's on, but they also contain acid that can eat into things like circuit-board traces and fragile little wires.

Spilled soda pop is worst of all, because it contains sugar and corn syrup and gets sticky when it dries. When you try to clean it, it just thins and then rethickens when it dries again, so you must soak, rinse, and wipe damaged components repeatedly. Some of the cleaning agents are very expensive, as is the labor cost.

So, don't set liquid near your computer.

D.W. Hauer
Jackson, MI

Corrections: re Colormania And Doctor ASCII

The Colormania tutorial, *HOT CoCo*, January, p. 108, contains a mistake. The reference to Listing 2a (third column, second paragraph) should read Listing 2b. The next paragraph's reference to Listing 2b, should read Listing 2a.

Also, in the February issue, p. 128, Doctor ASCII instructed our readers to type CSAVE "name", <start address>, <end address>, <EXEC address> in response to Claude L. Perry, Sr.'s question. This should read CSAVEM "name", <start address>, <end address>, <EXEC address>.

Exeter, NH

I enjoy your magazine and am interested in bulletin boards and users' groups. If anyone is interested in starting a users' group, or knows of a bulletin board system in the Kingston/Exeter area, please let me know.

Fred Ahlberg
RFD 1
Kingston, NH 03848

Send your letters to Feedback, HOT CoCo, *80 Pine St., Peterborough, NH 03458.*

TEN MOST-ASKED QUESTIONS about DYNACALC™

THE ELECTRONIC SPREAD-SHEET FOR 6809 COMPUTERS

1. What is an electronic spread-sheet, anyway?
Business people use spread-sheets to organize columns and rows of figures. DYNACALC simulates the operation of a spread-sheet without the mess of paper and pencil. Of course, corrections and changes are a snap. Changing any entered value causes the whole spread-sheet to be re-calculated based on the new constants. This means that you can play, 'what if?' to your heart's content.

2. Is DYNACALC just for accountants, then?
Not at all. DYNACALC can be used for just about any type of job. Not only numbers, but alphanumeric messages can be handled. Engineers and other technical users will love DYNACALC's sixteen-digit math and built-in scientific functions. You can build worksheets as large as 256 columns or 256 rows. There's even a built-in sort command, so you can use DYNACALC to manage small data bases — up to 256 records.

3. What will DYNACALC do for ME?
That's a good question. Basically the answer is that DYNACALC will let your computer do just about anything you can imagine. Ask your friends who have VisiCalc™, or a similar program, just how useful an electronic spread-sheet program can be for all types of household, business, engineering, and scientific applications. Typical uses include financial planning and budgeting, sales records, bills of material, depreciation schedules, student grade records, job costing, income tax preparation, checkbook balancing, parts inventories, and payroll. But there is no limit to what YOU can do with DYNACALC.

4. Do I have to learn computer programming?
NO! DYNACALC is designed to be used by non-programmers, but even a Ph.D. in Computer Science can understand it. Even experienced programmers can get jobs done many times faster with DYNACALC, compared to conventional programming. Built-in HELP messages are provided for quick reference to operating instructions.

5. Do I have to modify my system to use DYNACALC?
Nope. DYNACALC uses any standard 6809 configuration, so you don't have to spend money on another CPU board or waste time learning another operating system.

6. Will DYNACALC read my existing data files?
You bet! DYNACALC has a beautifully simple method of reading and writing data files, so you can communicate both ways with other programs on your system, such as the Text Editor, Text Processor, Sort/Merge, STYLOGRAPH™ word processor, RMS™ data base system, or other programs written in BASIC, C, PASCAL, FORTRAN, and so on.

7. How fast is DYNACALC?
Very. Except for a few seldom-used commands, DYNACALC is memory-resident, so there is little disk I/O to slow things down. The whole data array (worksheet) is in memory, so access to any point is instantaneous. DYNACALC is 100% 6809 machine code for blistering speed.

8. Is there a version of DYNACALC for MY system?
Probably. You need a 6809 computer (32k minimum) with FLEX™, UniFLEX™, or OS-9™ operating system. You also need a decent crt terminal, one with at least 80 characters per line, and direct cursor addressing. If your terminal isn't smart enough for DYNACALC, you probably need a new one anyway. The UniFLEX and OS-9 versions of DYNACALC allow you to mix different brands of terminal on the same system. There's also a special version of DYNACALC for Color Computers equipped with FLEX (Frank Hogg or Data-Comp versions).

9. How much does DYNACALC cost?
The FLEX versions are just $200 per copy; UniFLEX version $395; OS-9 version (works with LEVEL ONE or LEVEL TWO) $250. Orders outside North America add $7 per copy for postage. We encourage dealers to handle DYNACALC, since it's a product that sells instantly upon demonstration. Call or write on your company letterhead for more information.

10. Where do I order DYNACALC?
See your local DYNACALC dealer, or order directly from CSC at the address below. We accept telephone orders from 10 am to 6 pm, Monday through Friday. Call us at 314-576-5020. Your VISA or MasterCard is welcome. Please specify diskette size for FLEX or OS-9 versions. Software serial number is required for the UniFLEX version.

Order your DYNACALC today!

Foreign Dealers:

Australia & Southeast Asia: order from Paris Radio Electronics, 161 Bunnerong Road (PO Box 380) Kingsford, 2032 NSW Australia. Telephone: 02-344-9111.

United Kingdom: order from Compusense, Ltd., PO Box 169, London N13 4HT. Telephone: 01-882-0681.

Scandinavia: order from Swedish Electronics hk AB, Murargatan 23-25, Uppsala S-754 37 Sweden. Telephone: 18-25-30-00.

Computer Systems Center
13461 Olive Blvd.
Chesterfield, MO 63017
(314) 576-5020

CSC

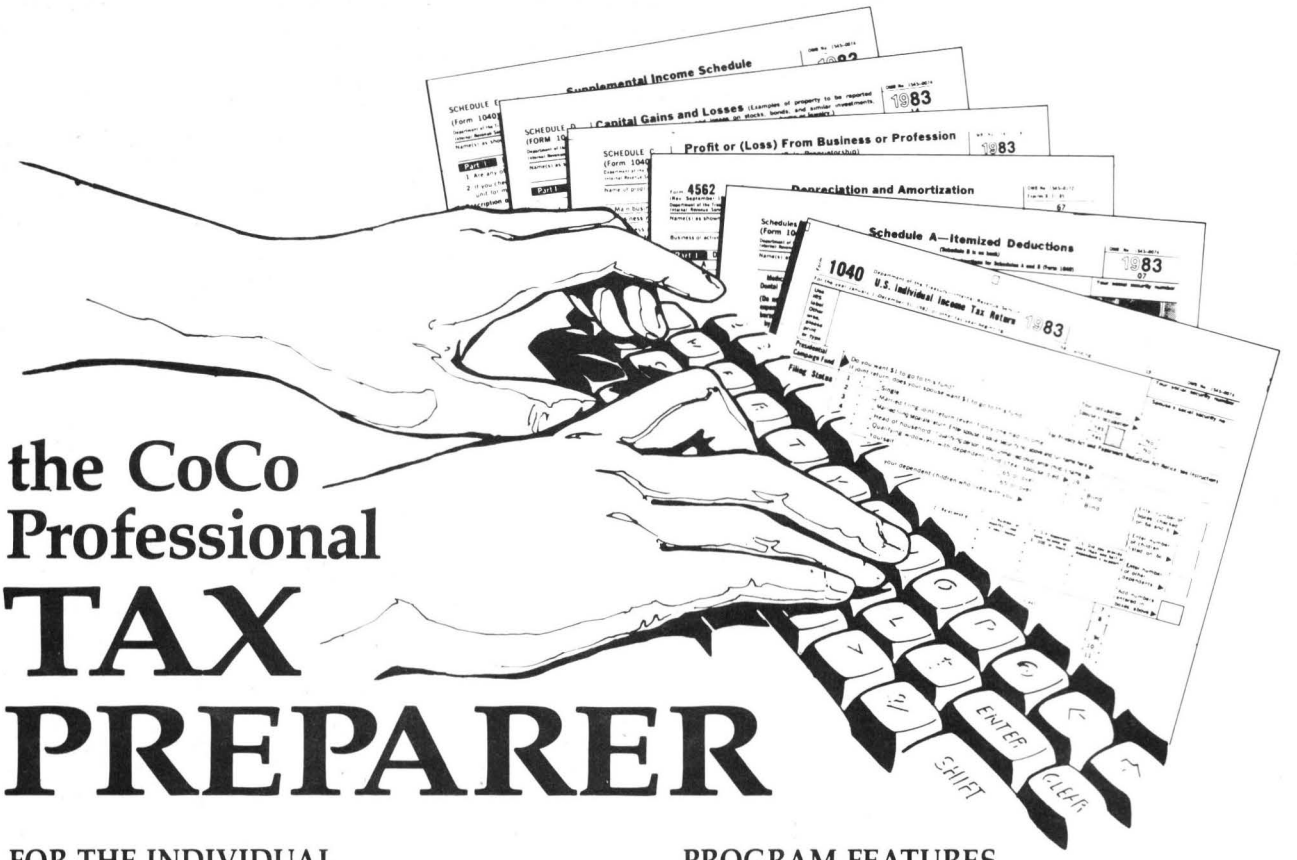
506

UniFLEX software prices include maintenance for the first year.

DYNACALC is a trademark of Computer Systems Center

VisiCalc is a trademark of VisiCorp.
STYLOGRAPH is a trademark of Great Plains Computer Co.
RMS is a trademark of Washington Computer Services.
FLEX and UniFLEX are trademarks of TSC.
OS-9 is a trademark of Microware and Motorola.

See List of Advertisers on page 130



the CoCo Professional TAX PREPARER

FOR THE INDIVIDUAL, IT ELIMINATES ANXIETY

File your taxes in confidence. The CoCo Professional Tax Preparer is accurate, thorough, and easy to use. Just answer the questions.

The CoCo Tax Preparer interviews you the way professionals in the large walk-in tax firms do. It takes you through each tax form in an organized manner. It knows which forms you need based on how you answer the questions it asks. And you can change data and make corrections - no hassle.

When you're done, the program prints your completed tax return on government-approved forms.

FOR THE PROFESSIONAL TAX PREPARER, IT SAVES TIME

Spend your time doing what you're supposed to do. *You* make the important business decisions, the CoCo Tax Preparer will do the rest. And you can simplify your filing — one diskette per client.

Interview your clients in a time-saving manner and get rid of your check-off sheets.

Produce complete tax returns on government-approved forms. The CoCo Tax Preparer lets you run continuous multicopy forms in the order you need, when you need them.

Mail to:
Micro Data Systems
6 Edward Drive
Ashland, MA 01721

Introductory offer: $99
Orders postmarked later than
February 28, 1984: $149.95

☐ Mastercard ☐ Visa ☐ Check or Money Order Enclosed

Card #_____ **Exp. Date** _____

Name _____

Address _____

City _____ **State** ____ **Zip** _____

Signature _____

I need the built-in sales tax table for _____ **(state).**

Mass. residents add 5% sales tax. Shipped post paid. Allow two weeks for delivery. Yearly update available.

PROGRAM FEATURES

Designed by a 15-year tax consultant, the program has built-in tax tables and tax rate schedules and supports the following forms:

1040
Schedules A, B, C, D, E, G, SE, W
Forms 2106, 2119, 3903, 4797
Office-at-Home
Installment Gain
Dependency Support
Credits and Other Taxes

$149.95

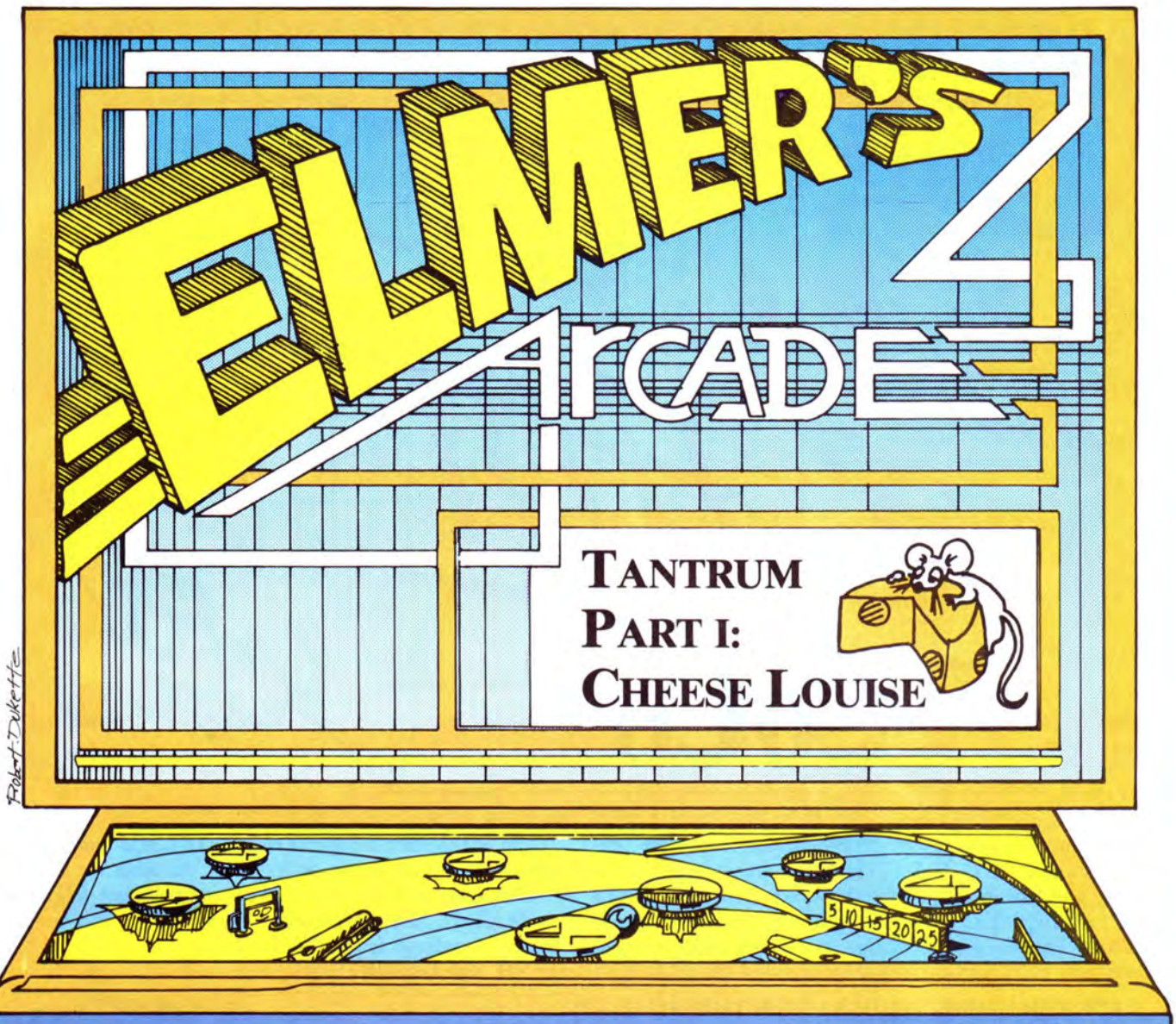
For a limited time only.
A $400 value

MORE FEATURES

- Over 170 full-screen menus displayed on command. Fully menu-driven screens — each appears only when required.
- Full reverse-screen scrolling and forward-screen block scroll.
- Calculator mode supports +, −, *, /, = on numeric data.
- Edit capability: any line at any time. Supports change, delete, hack, search, and insert commands. Eliminates the need for check-off sheets
- Runs on 32K extended Basic (one disk drive with change of diskette during program execution) or two disk drives. (A special-order version runs on 64K RAM units with one or two disk drives.) Comes with diskettes and operating manual that describes each screen presentation. Additional forms are available by special order.
- Full disk drive storage for all data and computations.
- Printed output on pin-fed or tractor-fed printers, for government-approved forms.
- Its combination of machine language and Basic is fast and it minimizes memory use.

VISA MasterCard

71



instant CoCo

by Richard Ramella

Elmer was scribbling something in a Big Chief tablet as I quietly entered his arcade not long ago. I could tell he was in the throes of creative expression by the agonized look on his face.

"What are you doing, *A History of The Penny Arcade*?" I asked.

"Eep!" he ulped, pocketing a smidgeon of a pencil and shoving the tablet under the counter.

"Come on, Elmer, let me see."

"See what?"

"What you're writing. I'm a writer. Maybe I can help."

"It's nothing," he said, but I detected shy interest in his voice. It took less than a minute to coax out the Big Chief tablet. "It's just the first draft of a book for little kids," my arcadian friend protested hopefully as I read a lengthy poem called Tantrum.

"This isn't bad at all," I lied when I finished.

To understand how bad it really was, look at the first few lines...

There were three cranky robots:
Kate, Ned, and Pete,
Who worked in a loft
Out on Cheesemaker Street.
And under the floor
Lived a mouse named Louise,
Whose pleasure in life
Was gathering up cheese...

It went on. And on and on. The rhyming got worse. To spare you torture and possible death by poetry, I'll simply tell the plot briefly.

The three robots threw a grand tantrum, which Elmer blamed on "a mindbending, spine-rending electrical surge." With the robots hopping all over the place, it was pretty tough for Louise to reach the cheese. In fact, she was in constant danger of being squished as she darted among metallic robot feet and arms in search of bits of de-Brie.

System Requirements

16K RAM
Color Basic

Some Like It HOT

HOT CoCo®

Guaranteed to make your TRS-80 Color Computer* sizzle with color, **HOT CoCo** magazine is informative, interesting, and *best of all* just for the Color Computer. You'll look forward to **HOT CoCo** month after month because it has something for everyone, from the novice right on up to the expert. **HOT CoCo** gives you:

- •PROGRAMMING TECHNIQUES & TUTORIALS—that promise to make you a superior programmer.
- •UTILITIES—to save you time and effort on all your routine tasks.
- •EXPERTLY WRITTEN COLUMNS—including *BASIC, GRAPHICS, FLEX and GAMES.*
- •HARDWARE & CONSTRUCTION—ideas on interfacing and enhancing to make building projects a breeze.
- •EDUCATIONAL APPLICATIONS—will stimulate and encourage imaginative thinking in your child.

PLUS

- •BUSINESS PROGRAMS—sure to make you a star at the office.
- •FEATURES ON COLOR APPLICATIONS—make your computer reach its full potential and get your money's worth from your machine.
- •BUYER'S GUIDES & PRODUCT REVIEWS—now you can stop running around comparing prices and products and start running your computer.
- •HOME & HOBBY APPLICATIONS—exciting ways to help your computer add enjoyment to your leisure time.
- •ANSWERS TO SPECIFIC QUESTIONS—it's like having your own private consultant—free!

Join in the color explosion with **HOT CoCo** today! Take advantage of our money saving offer, 12 issues for $24.97. A 13th issue is yours FREE with pre-payment (check or credit card). Use the attached order form, the coupon below, or call toll free 1-800-258-5473.
(In NH. call 1-924-9471)

YES! I want my Color Computer to sizzle with color. I understand that with payment enclosed or credit card order I will receive a free issue making a total of 13 issues for $24.97.

☐ CHECK/MO ☐ MC ☐ VISA ☐ AE ☐ BILL ME

card #_____ exp. date_____
signature_____
name_____
address_____
city_____ state_____ zip_____

Canada & Mexico $27.97/1 yr. only, U.S. funds drawn on U.S. bank.
Foreign Surface $44.97/1 yr. only, U.S. funds drawn on U.S. bank.
Please allow 6–8 weeks for delivery.

*This price voids all previous offers.

HOT CoCo • PO Box 975 • Farmingdale NY 11737

* TRS-80 Color Computer is a trademark of Radio Shack, a division of Tandy Corp. **343F4**

And it is this part of the plot that gave me the idea for a computer game.

In time, Louise can stand the situation no more. She eats a magic morsel, which Elmer attempts to rhyme with tragic weasel (bring the smelling salts; I'm faint!).

And the little mouse becomes a giant. Robots Ned and Kate run away, but Pete is trapped in a closet, through the openings of which elephantine Louise tosses cheese bombs until all the robots promise never again to give her a hard time when she's gathering food.

The End (and thank goodness).

It was the second part of this story that gave me an idea for another computer game.

"What are you going to do with this...this...poem?" I asked Elmer.

"I thought I'd sell movie rights first."

"Ever thought of making it into a video game?"

His face flushed. "Not for one moment!"

"Consider it, Elmer. Look at the merchandising boom of Pac-Man. If Louise becomes a star, maybe she could endorse Purina Rat Chow or something like that."

"You're putting me on," he said.

"Just let me give it a try."

"I want a contract," he said trustingly.

I put out my hand. "Elmer, a handshake's not good enough between friends?"

He grumbled but shook.

The rest is video-arcade history.

I wrote not one but two Tantrum programs following Elmer's plot line. This month, by special arrangement with the author, I will share *Tantrum Part I—Cheese Louise* with all CoCo owners with at least 16K Color Basic.

Next month the adventure continues with *Tantrum Part II—Cheese Bombs.* I just love sagas, don't you?

I took the CoCo to Elmer's Arcade and set up the games for anyone with a nickel. Elmer and I made more than five bucks the first day!

Here's how to play *Cheese Louise:*

Type RUN and tap the enter key. A title introduces the three robots. It's the only part of Elmer's poem I could stand to use. When the timer comes on at the bottom left of the screen, the game begins. For each round the timer starts at 1,000 and is reduced to zero by 10s. You will see a yellow pixel, the cheese, and a blue pixel, Louise the mouse. Tap or hold down the four arrow keys to take Louise to the cheese. When the mouse contacts the cheese, you win the points shown on the timer.

"The game continues forever or until you score at least 10,000 points. Then Louise has all the nibble-chow she needs and you are declared The Big Cheese."

There are, of course, some problems. On succeeding rounds of play, the cheese can be dropped at greater distances from the mouse. Also, the orange robots hop up and down, slowly but without warning. If a robot comes into contact with Louise, 250 points are lost. Also, at times it doesn't work to contact the cheese on a slanting approach. Example: The mouse is immediately northwest of the cheese, and taking it south can turn the cheese temporarily blue and lose points. Also, hitting a side of the room subtracts 250 points.

The game continues forever or until you score at least 10,000 points. Then Louise has all the nibble-chow she needs and you are declared The Big Cheese. If in early tries you find yourself with a hopelessly minus score, my advice is hit the break key and start over.

This month's program makes you the mouse, and the robots are the villains. Next month, the mouse becomes the villain, and you are a robot. I urge you to be present and accounted for, because Part II of Tantrum provides you with a robot whose movement you control.

Last, let me talk briefly about my willingness to help if you have trouble typing in an Elmer's Arcade program. You can reach me two ways: by mail or telephone.

Write to Richard Ramella, 1493 Mt. View Ave., Chico, CA 95926. If you want an answer, include a stamped, self-addressed envelope. If writing from another country, include coin of your nation equal to the stamps on the outgoing letter. If possible, include a printout of the program as it exists in your computer and tell the problem you are having. If you can't provide

See List of Advertisers on page 130

the printout, be as specific as possible in relating the error—line occurring, what happens, any clue at all. I surprise even myself with my ability to help in these cases.

However, I cannot attempt to debug a program which has been changed in any way from the version published.

Also, I don't mind if you call me, though it's easier to answer questions by mail. My number is 916-343-7658. Remember, it's earlier in California by three hours, if you're calling from the East. I don't like to talk about computer programs after 11:30 p.m. *my time*. I am most often home and amenable to telephone calls from 6 p.m. to 10 p.m Pacific Standard Time on week nights.

Of course, I will not accept a collect call. Also, if I am out when you call, I cannot afford to return your call at my expense. However, you may leave your number and specifically request that I return your call collect.

Lest that make me seem like a real sour apple, let me end by saying it is always a pleasure to hear from anyone with a question about programs I publish. Elmer's the sour apple, not me. ■

```
120 CLS 7
130 RT=10
140 PRINT @ 352,"TANTRUM (PART I
 - CHEESE LOUISE)";
150 FOR T=1 TO 4
160 SOUND 5,3
170 SOUND 5,3
180 SOUND 17,3
190 SOUND 17,3
200 NEXT T
210 CLEAR 300
220 FOR A=1 TO 10
230 Z$=Z$+CHR$(128)
240 NEXT A
250 Z=112
260 A$=CHR$(128)
270 D$=CHR$(131+Z)
280 E$=CHR$(132+Z)
290 F$=CHR$(133+Z)
300 I$=CHR$(136+Z)
310 K$=CHR$(138+Z)
320 M$=CHR$(140+Z)
330 N$=CHR$(141+Z)
340 O$=CHR$(142+Z)
350 P$=CHR$(143+Z)
360 A$(1)=Z$
370 B$(1)=Z$
380 C$(1)=Z$
390 D$(1)=A$+A$+A$+A$+F$+K$+A$+A
$+A$+A$
400 E$(1)=A$+A$+A$+P$+P$+P$+P$+A
$+A$+A$
410 F$(1)=A$+A$+A$+K$+P$+P$+F$+A
$+A$+A$
420 G$(1)=A$+A$+A$+A$+K$+F$+A$+A
$+A$+A$
430 H$(1)=A$+A$+A$+D$+K$+F$+D$+A
$+A$+A$
440 A$(2)=Z$
450 B$(2)=Z$
460 C$(2)=A$+A$+A$+A$+F$+K$+A$+A
$+A$+A$
470 D$(2)=M$+M$+M$+M$+P$+P$+M$+M
$+M$+M$
480 E$(2)=A$+A$+A$+A$+P$+P$+A$+A
$+A$+A$
490 F$(2)=A$+A$+O$+M$+M$+M$+M$+N
$+A$+A$
500 G$(2)=A$+E$+I$+A$+A$+A$+A$+E
$+I$+A$
510 H$(2)=Z$
520 A$(3)=A$+A$+A$+K$+A$+A$+F$+A
$+A$+A$
530 B$(3)=A$+A$+A$+K$+F$+K$+F$+A
$+A$+A$
540 C$(3)=A$+A$+A$+M$+P$+P$+M$+A
$+A$+A$
550 D$(3)=D$+A$+A$+A$+P$+P$+A$+A
$+A$+D$
560 E$(3)=M$+M$+M$+M$+M$+M$+M$+M
$+M$+M$
570 F$(3)=Z$
580 G$(3)=Z$
590 H$(3)=Z$
600 GOSUB 1340
610 K=10
620 PRINT @ 352,"THERE WERE THRE
E CRANKY ROBOTS."
630 FOR T=1 TO 10
640 SOUND 89,1
650 SOUND 125,1
660 SOUND 147,1
670 SOUND 176,1
680 NEXT
690 GOSUB 1340
700 FOR X=352 TO 383
710 PRINT @ X,CHR$(32);
720 NEXT
730 PRINT @ 356,"KATE";
740 R=33
750 X=1
760 GOSUB 1130
770 FOR T=1 TO 10
780 SOUND 232,1
790 SOUND 227,1
800 NEXT
810 PRINT @ 366,"NED";
820 R=43
830 X=1
840 GOSUB 1130
850 FOR T=1 TO 10
860 SOUND 229,1
870 SOUND 223,1
880 NEXT
890 PRINT @ 376,"PETE";
900 R=53
910 X=1
920 GOSUB 1130
930 FOR T=1 TO 20
940 SOUND 227,1
950 SOUND 218,1
960 NEXT
970 FOR S=352 TO 383
980 PRINT @ S,CHR$(143+96);
990 NEXT
1000 K=0
1010 Q=Q+1
1020 TI=1010
1030 IF Q/2=INT(Q/2) THEN B=RND(
4)+4: D=15 ELSE B=RND(3)+14: D=4
1040 A=RND(RT)+3
1050 C=4
1060 X=RND(3)
1070 H=RND(3)
1080 TI=TI-10
1090 PRINT @ 384,"TIMER:"TI;
1100 IF TI=0 THEN 1530
1110 IF H>1 THEN 1220
1120 R=RND(3)*10+23
1130 PRINT @ R,A$(X);
1140 PRINT @ R+32,B$(X);
1150 PRINT @ R+64,C$(X);
1160 PRINT @ R+96,D$(X);
1170 PRINT @ R+128,E$(X);
1180 PRINT @ R+160,F$(X);
1190 PRINT @ R+192,G$(X);
1200 PRINT @ R+224,H$(X);
1210 IF J=10 THEN RETURN
1220 IF K=0 THEN RESET(C,D)
1230 Y=C
1240 N=D
1250 IF (PEEK(341)AND8)=0 THEN P
OKE341,255:D=D-1
1252 IF (PEEK(342)AND8)=0 THEN P
OKE342,255:D=D+1
1254 IF (PEEK(343)AND8)=0 THEN P
OKE343,255:C=C-1
1256 IF (PEEK(344)AND8)=0 THEN P
OKE344,255:C=C+1
1260 IF C=0 THEN RETURN ELSE IF
POINT(C-1,D)<>0 THEN W=POINT(C-1
,D): GOTO 1370
1270 IF POINT(C+1,D)<>0 THEN W=P
OINT(C+1,D): GOTO 1370
1280 IF POINT(C,D-1)<>0 THEN W=
POINT(C,D-1): GOTO 1370
1290 IF POINT(C,D+1)<>0 THEN W=P
OINT(C,D+1): GOTO 1370
1300 SET(A,B,2)
1310 SET(C,D,3)
1320 SOUND RND(13)*10+80,1
1330 GOTO 1060
1340 FOR T=1 TO 1000
1350 NEXT
1360 RETURN
1370 IF W=2 THEN SET(C,D,3): SC=
SC+TI: FOR T=1 TO 3: SOUND 204,1
: SOUND 210,1: NEXT T: SOUND 218
,6: SOUND 218,6: GOTO 1390
1380 PRINT @ 352,"PENALTY: 250";
: FOR T=1 TO 10: SOUND 200,1: SO
UND 180,1: NEXT: FOR T=352 TO 36
3: PRINT @ T,CHR$(143+96);: NEXT
: SC=SC-250
1390 PRINT @ 400,"SCORE";SC;"  "
;
1400 IF SC>9999 THEN PRINT @ 0,"
W-I-N-N-E-R !": GOTO 1580
1410 J=10
1420 X=1
1430 R=33
1440 GOSUB 1130
1450 R=43
1460 GOSUB 1130
1470 R=53
1480 GOSUB 1130
1490 J=0
1500 IF RT<55 THEN RT=RT+5
1510 IF W=2 THEN 1010
1520 GOTO 1300
1530 PRINT @ 0,"LOST... MICE MUS
T RUN QUICKLY!"
1540 SOUND 200,1
1550 SOUND 180,1
1560 GOTO 1540
1570 END
1580 CLS2
1590 FOR A=1 TO 100
1600 PRINT @ RND(415),CHR$(128);
1610 NEXT A
1620 PRINT @ 428,"WINNER !";
1630 PRINT @ 485,"YOU ARE THE BI
G CHEESE";
1640 SOUND 204,2
1650 SOUND 204,2
1660 FOR T=1 TO 50
1670 NEXT
1680 SOUND 204,2
1690 SOUND 210,5
1700 SOUND 213,5
1710 SOUND 216,8
1720 FOR T=1 TO 50
1730 NEXT
1740 SOUND 204,4
1750 FOR T=1 TO 500
1760 NEXT
1770 GOTO 1640
1780 END
```

Program Listing. Cheese Louise

END

Telewriter-64™

the Color Computer Word Processor

- **3 display formats: 51/64/85 columns × 24 lines**
- **True lower case characters**
- **User-friendly full-screen editor**
- **Right justification**
- **Easy hyphenation**
- **Drives any printer**
- **Embedded format and control codes**
- **Runs in 16K, 32K, or 64K**
- **Menu-driven disk and cassette I/O**
- **No hardware modifications required**

THE ORIGINAL

Simply stated, Telewriter is the most powerful word processor you can buy for the TRS-80 Color Computer. The original Telewriter has received rave reviews in every major Color Computer and TRS-80 magazine, as well as enthusiastic praise from thousands of satisfied owners. And rightly so.

The standard Color Computer display of 32 characters by 16 lines without lower case is simply inadequate for serious word processing. The checkerboard letters and tiny lines give you no feel for how your writing looks or reads. Telewriter gives the Color Computer a 51 column by 24 line screen display with *true lower case characters*. So a Telewriter screen looks like a printed page, with a good chunk of text on screen at one time. In fact, more on screen text than you'd get with Apple II, Atari, TI, Vic or TRS-80 Model III.

On top of that, the sophisticated Telewriter full-screen editor is so simple to use, it makes writing fun. With single-letter mnemonic commands, and menu-driven I/O and formatting, Telewriter surpasses all others for user friendliness and pure power.

Telewriter's chain printing feature means that the size of your text is never limited by the amount of memory you have, and Telewriter's advanced cassette handler gives you a powerful word processor without the major additional cost of a disk.

...one of the best programs for the Color Computer I have seen...
— Color Computer News, Jan. 1982

TELEWRITER-64

But now we've added more power to Telewriter. Not just bells and whistles, but major features that give you total control over your writing. We call this new supercharged version Telewriter-64. For two reasons.

64K COMPATIBLE

Telewriter-64 runs fully in any Color Computer — 16K, 32K, or 64K, with or without Extended Basic, with disk or cassette or both. It automatically configures itself to take optimum advantage of all available memory. That means that when you upgrade your memory, the Telewriter-64 text buffer grows accordingly. In a 64K cassette based system, for example, you get about 40K of memory to store text. So you don't need disk or FLEX to put all your 64K to work immediately.

64 COLUMNS (AND 85!)

Besides the original 51 column screen, Telewriter-64 now gives you 2 additional high-density displays: 64 × 24 and 85 × 24!! Both high density modes provide all the standard Telewriter editing capabilities, and you can switch instantly to any of the 3 formats with a single control key command.

The 51 × 24 display is clear and crisp on the screen. The two high density modes are more crowded and less easily readable, but they are perfect for showing you the exact layout of your printed page, *all on the screen at one time*. Compare this with cumbersome "windows" that show you only fragments at a time and don't even allow editing.

RIGHT JUSTIFICATION & HYPHENATION

One outstanding advantage of the full-width screen display is that you can now set the screen width to match the width of your printed page, so that "what you see is what you get." This makes exact alignment of columns possible and it makes hyphenation simple.

Since short lines are the reason for the large spaces often found in standard right justified text, and since hyphenation is the most effective way to eliminate short lines, Telewriter-64 can now promise you some of the best looking right justification you can get on the Color Computer.

FEATURES & SPECIFICATIONS:

Printing and formatting: Drives any printer (LPVII/VIII, DMP-100/200, Epson, Okidata, Centronics, NEC, C. Itoh, Smith-Corona, Terminet, etc).

Embedded control codes give full dynamic access to intelligent printer features like: underlining, subscript, superscript, variable font and type size, dot-graphics, etc.

Dynamic (embedded) format controls for: top, bottom, and left margins; line length, lines per page, line spacing, new page, change page numbering, conditional new page, enable/disable justification.

Menu-driven control of these parameters, as well as: pause at page bottom, page numbering, baud rate (so you can run your printer at top speed), and Epson font. "Typewriter" feature sends typed lines directly to your printer, and Direct mode sends control codes right from the keyboard. Special Epson driver simplifies use with MX-80.

Supports single and multi-line headers and automatic centering. Print or save all or any section of the text buffer. Chain print any number of files from cassette or disk.

File and I/O Features: ASCII format files — create and edit BASIC, Assembly, Pascal, and C programs, Smart Terminal files (for uploading or downloading), even text files from other word processors. Compatible with spelling checkers (like Spell 'n Fix).

Cassette verify command for sure saves. Cassette auto-retry means you type a load command only once no matter where you are in the tape.

Read in, save, partial save, and append files with disk and/or cassette. For disk: print directory with free space to screen or printer, kill and rename files, set default drive. Easily customized to the number of drives in the system.

Editing features: Fast, full-screen editor with wordwrap, block copy, block move, block delete, line delete, global search and replace (or delete), wild card search, fast auto-repeat cursor, fast scrolling, cursor up, down, right, left, begin line, end line, top of text, bottom of text; page forward, page backward, align text, tabs, choice of buff or green background, complete error protection, line counter, word counter, space left, current file name, default drive in effect, set line length on screen.

Insert or delete text anywhere on the screen without changing "modes." This fast "free-form" editor provides maximum ease of use. Everything you do appears immediately on the screen in front of you. Commands require only a single key or a single key plus CLEAR.

...truly a state of the art word processor... outstanding in every respect.
— The RAINBOW, Jan. 1982

PROFESSIONAL WORD PROCESSING

You can no longer afford to be without the power and efficiency word processing brings to everything you write. The TRS-80 Color Computer is the lowest priced micro with the capability for serious word processing. And only Telewriter-64 fully unleashes that capability.

Telewriter-64 costs $49.95 on cassette, $59.95 on disk, and comes complete with over 70 pages of well-written documentation. (The step-by-step tutorial will have your writing with Telewriter-64 in a matter of minutes.)

To order, send check or money order to:

Cognitec
704 N. Nob St.
Del Mar, CA 92014 ↙ 121

Or check your local software store. If you have questions, or would like to order by Visa or Mastercard, call us at (619) 755-1258 (weekdays, 8AM-4PM PST). Dealer inquiries invited.

(Add $2 for shipping. Californians add 6% state tax. Allow 2 weeks for personal checks. Send self-addressed stamped envelope for Telewriter reviews from CCN, RAINBOW, 80-Micro, 80-U.S. Telewriter owners: send SASE or call for information on upgrading to Telewriter-64. Telewriter-compatible spelling checker (Spell 'n Fix) and Smart Terminal program (Colorcom/E) also available. Call or write for more information.)

The Basic Beat

THE FIRST STEPS TO BASIC PROGRAMMING LESSON 10

by James W. Wood

How difficult is changing a TRS-80 Model I, III, or 4 Basic listing to use on a CoCo? It depends. Many Basic commands perform exactly the same on a CoCo as on the other Radio Shack computers. The other models use black-and-white monitors, so the primary differences between their programs and the Color Computer's involve the graphics commands. Another problem area is the conversion of keyboard PEEKs, but converting Assembly-language subroutines is not within our scope.

You don't need to rewrite programs command for command. You can rewrite SET graphics as CHR$ graphics. It is more important to have the final product function correctly than it is to be sure all commands are the same.

Whoever uses the program will be more impressed with how well it works than with how closely the converted listing resembles the original. Most users probably won't even read the listing. There is no magic formula to convert software. I'll try to give some general guidelines for specific commands.

The PRINT command is exactly the same on any Radio Shack model, except that you must change the black-and-white (B&W) computer's LPRINT to PRINT#-2.

The CoCo's screen has 16 lines of 32 characters each. The B&W screen has 16 lines of 64 characters. A CoCo screen, therefore, cannot show as much information or writing at one time as a B&W screen. You must adjust PRINT lines to fit the smaller format by breaking one screen of printing down into two pages. The CoCo's small screen makes cutting words in half more difficult to avoid.

Program Listing 1 runs on B&W or a CoCo. On the CoCo, it splits words. Adjust for this by inserting enough spaces (Program Listing 2), or using two PRINT statements (Program Listing 3).

Following PRINT with a comma causes a B&W to print four columns, each 16 characters wide. Program Listing 4 is an example. The program displays two 16-character-wide columns on a CoCo. PRINT TAB is usually used to center headings. The CoCo's center is 16 spaces from the left, while the B&W's is 32 spaces from the left. To center a title on a B&W, use Program Listing 5. Use Program Listing 6 for a CoCo. You can usually make the conversion by subtracting 16 from the PRINT TAB number.

Lowercase does not exist on the screen of some Model I's. Model IIIs have true lowercase. Sentences with lowercase look good on a Model III, but appear unsightly on the CoCo. Unless it's important, change all letters in sentences to uppercase.

PRINT@ positions on a B&W range from 0–1,023, with each line increasing by 64. PRINT @ on a CoCo goes from 0–511. Generally, to convert PRINT@, divide the position by two.

```
10 PRINT"LENGTHY LINES MAKE LIFE
 INTERESTING."
```

Program Listing 1

```
10 PRINT"LENGTHY LINES MAKE LIFE
         INTERESTING."
```

Program Listing 2

```
10 PRINT"LENGTHY LINES MAKE LIFE
"
20 PRINT"INTERESTING."
```

Program Listing 3

```
10 PRINT"HELLO",
20 GOTO10
```

Program Listing 4

```
10 CLS
20 PRINTTAB(25)"CENTERED TITLE"
```

Program Listing 5

```
10 CLS
20 PRINTTAB(9)"CENTERED TITLE"
```

Program Listing 6

```
10 CLS
20 A$=CHR$(187)+CHR$(174)+CHR$(1
45)
30 PRINT@543,A$;
40 GOTO40
```

Program Listing 7

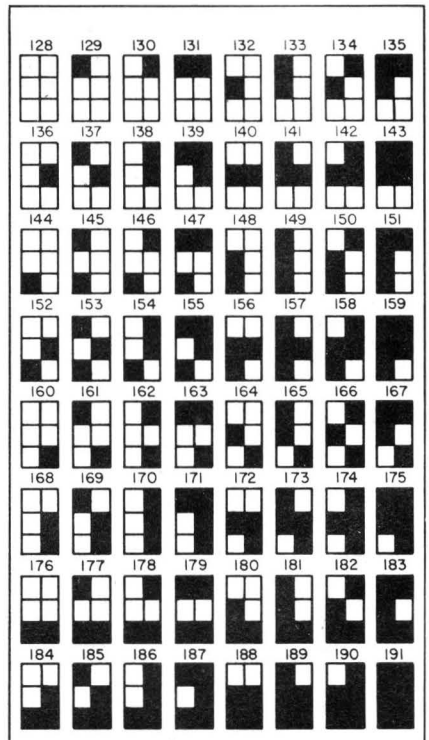


Fig. 1. The B&W's CHR$ and POKE Graphics Characters

Top Performance Demands Proper Tools!

JBM's MIDWARE line of quality software, is available on either five and one-quarter or eight inch diskettes. All of JBM's software packages come complete with comprehensive user's manuals.

JBM'S MIDWARE *for* OS9

OS9 is a registered trademark of Microware Corporation.

For more information, contact:

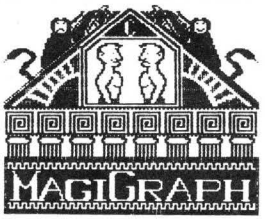
The JBM Group, Inc.
332 West Church Road
King of Prussia, PA USA 19406
TEL: 215-337-3138
TWX: 510-660-3999

the JBM group

VISA and MASTERCARD accepted.

190

See List of Advertisers on page 130



NEW GOOD STUFF FOR EVERY COLOR COMPUTER

Turn your Color Computer into a graphic design center with the ease of a keystroke! **MagiGraph** makes it simple to create highly detailed figures up to and including an entire high-resolution screen. Designed for those with some experience in Basic and Assembly Language programming, **MagiGraph** includes lots of special features:

- **A full set of logical and pixel manipulation functions** simplifies the development of complex figures.
- **An editor** lets you zoom in and work on every detail of your design. Toggle between the "macro" and "micro" screens for perspective on your creations.
- **Nine animation buffers** allow you to preview each sequence to ensure continuity and smooth flow.
- **Versatile I/O routines** store a graphic screen on cassette or floppy disk; recall it later for use by another program or revise it with **MagiGraph**.

If you're looking for the finest graphic development utility available for your Color Computer, THIS IS IT. Maximize your machine's potential, while you push your imagination to the limit — with **MagiGraph**!

By Kevin Dooley. Cassette **$34.95** (16K required); Disk **$39.95** (32K Extended Color BASIC required); Amdisk cartridge **$44.95.**

STOP WAITING AROUND FOR THE PRINTER!

SAVE TIME! Use the Micro Works **CSPOOL** printer spooler. Say you've just finished editing a five-page letter to the IRS and you start printing it out. Now you need to run your personal finance program to find out if you'll be able to afford to eat next month, but you have to WAIT for the @#?!* PRINTER! **CSPOOL** will let you KEEP COMPUTING while your printer is PLODDING. **CSPOOL** uses only 20 bytes of Basic's memory yet gives you 32K of printer buffer. It's like having two computers for the price of one! Even better yet, **CSPOOL** can be yours for FREE! When you buy your 64K RAM UPGRADE KIT from The Micro Works, we'll GIVE you a copy of this great little program for FREE! Or we will sell it to you, on cassette or diskette, for only **$19.95.** Requires 64K. Not for FLEX or OS9.

64K MEMORY UPGRADE KIT: For Rev. levels E, ET, NC, TDP-100s, and Color Computer II. Eight prime 64K RAM chips, instructions, and CSPOOL: **$64.95.**

SYSTEMS SOFTWARE

MACRO-80C: DISK-BASED EDITOR, ASSEMBLER AND MONITOR—With all the features the serious programmer wants, this package includes a powerful 2-pass macro assembler with conditional assembly, local labels, include files and cross referenced symbol tables. MACRO-80C supports the complete Motorola 6809 instruction set in standard source format. Incorporating all the features of our Rompack-based assembler (SDS-80C), MACRO-80C contains many more useful instructions and pseudo-ops which aid the programmer and add power and flexibility. The screen-oriented editor is designed for efficient and easy editing of assembly language programs. MACRO-80C allows global changes and moving/copying blocks of text. You can edit lines of assembly source which exceed 32 characters. DCBUG is a machine language monitor which allows examining and altering of memory, setting break points, etc.

Editor, assembler and monitor—along with sample programs—come on one Radio Shack compatible disk. Extensive documentation included. By Andy Phelps. **$99.95**

SDS-80C: SOFTWARE DEVELOPMENT SYSTEM—Our famous editor, assembler and monitor in Rompack. Like MACRO-80C, it allows the user to write, assemble and debug assembly language programs with no reloading, object patching or other hassles. Supports full 6809 instruction set. Complete manual included. **$89.95**

MICROTEXT: COMMUNICATIONS VIA YOUR MODEM! Now you **can** use your printer with your modem! Your computer can be an intelligent printing terminal. Talk to timeshare services or to other personal computers; print simultaneously through a second printer port; and re-display text stored in memory. Download text to Basic programs; dump to a cassette tape, or printer, or both. Microtext can be used with any printer or no printer at all. It features user-configurable duplex/parity for special applications, and can send any ASCII character. You'll find many uses for this general purpose module! ROMPACK includes additional serial port for printer. **$59.95**

MICRO WORKS COLOR FORTH

- Faster to program in than Basic
- Easier to learn than Assembly Language
- Executes in less time than Basic

The MICRO WORKS COLOR FORTH is a Rompack containing everything you need to run Forth on your Color Computer. COLOR FORTH consists of the standard Forth Interest Group (FIG) implementation of the language plus most of FORTH-79. It has a super screen editor with split screen display. Mass storage is on cassette. COLOR FORTH also contains a decompiler and other aids for learning the inner workings of this fascinating language. It will run on 4K, 16K, and 32K computers. And COLOR FORTH contains 10K of ROM, leaving **your** RAM for **your** programs! There are simple words to effectively use the Hi-Res Color Computer graphics, joysticks, and sound.

Includes a 112-page manual with a glossary of the system-specific words, a full standard FIG glossary and complete source listing.

MICRO WORKS COLOR FORTH . . . THE BEST! From the leader in FORTH, Talbot Microsystems. **$109.95**

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: $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 reload the monitor each time you use it. The EPROM plugs into the Extended Basic ROM Socket or the Romless Pack I. **CBUG ROM: $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: $49.95**

HARDWARE

PARALLEL PRINTER INTERFACE—Serial to parallel converter allows use of all standard parallel printers. PI80C plugs into the serial output port, leaving your Rompack slot free. You supply the printer cable. **PI80C: $59.95**

SUPER-PRO KEYBOARD—$69.95 (For computers manufactured after Oct. 1982, add $4.95)

ROMLESS PACKS for your custom EPROMS — call or write for information.

BOOKS

6809 ASSEMBLY LANGUAGE PROGRAMMING, by Lance Leventhal, **$16.95**

TRS-80 COLOR COMPUTER GRAPHICS, by Don Inman, **$14.95**

ASSEMBLY LANGUAGE GRAPHICS FOR THE TRS-80 COLOR COMPUTER, by Don Inman, **$14.95**

STARTING FORTH, by L. Brodie, **$17.95**

GAMES

ZAXXON—The real thing. Excellent. What more can we say? Cassette requires 32K. **$39.95**

STAR BLASTER—Blast your way through an asteroid field in this action-packed Hi-Res graphics game. Available in ROMPACK; requires 16K. **$39.95**

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

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

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

CAVE HUNTER—Experience vivid colors, bizarre sounds and eerie creatures as you wind your way through a cave maze in search of gold treasures. This exciting Hi-Res game by Mark Data Products requires 16K for cassette version. **$24.95**

THE MICRO WORKS

P.O. Box 1110-D
Del Mar, CA 92014
(619) 942-2400

California Residents add 6% Tax

Master Charge/Visa and COD Accepted

The Model III has special characters available from CHR$(192) to CHR$(255), including Greek letters, card suits, people, and rockets. These are not built into the Color Computer.

Graphics conversions are a little more complicated. PRINT CHR$ and POKE graphics vary in size and design. Figure 1 shows the B&W's characters for CHR$. Figure 2 displays characters available for a Color Computer. In Fig. 1 the dark areas are light on the screen, and in Fig. 2, dark areas are dark.

The B&Ws have a 3-by-2 pattern of SET positions within each character code. Color Basic's character codes occupy a 2-by-2 pattern of SET positions.

Program Listing 7 for B&W and Program Listing 8 for the CoCo both draw the same space ship. There is no easy formula for converting. The B&W's graphics are three spaces tall and, therefore, can print the ship as one string. Color Basic must use two strings to print the ship, since its CHR$ graphics are only two units high.

Converting POKEs involves the same problem as converting the CHR$ patterns, but you'll also have trouble locating video memory. Video memory on the B&W ranges from 15,360–16,383, for a total of 1,024 video locations. Each is one PRINT@ position. Color Basic's video is from 1,024–1,535, for a total of 512 locations.

Program Listing 9 makes the upper-left corner PRINT@ position on a B&W all white. Program Listing 10 makes the same corner buff on the CoCo.

Rather than devise a formula to convert, I usually watch the program on a B&W and then write a color program that has the same action. To change B&W video POKE positions to color, you can use my formula (Wood's Equation?):

Color video POKE position = (B&W video POKE position) ÷ 2 − 6656.

Remember that the color screen does not contain as many PRINT@ positions as B&W, nor does each PRINT@ position contain as many SET positions. Often in converting graphic programs it is necessary to simplify graphics due to the CoCo's smaller number of video positions.

Another graphics command is SET. All Radio Shack computer screens start at 0,0 in the upper-left corner. The B&W's lower-right corner has coordinates 127,47. The CoCo's lower-right corner has coordinates 63,31. As a general rule, divide the X coordinate by two and the Y coordinate by 1.5.

Program Listing 11 draws a border around the B&W computer's screen. Program Listing 12 does the same for a Color Computer. Remember that the CoCo's SET command needs a third number representing the color to be used.

The POINT command on a B&W computer returns the value of −1 if a SET position is on, and zero if it is off. The CoCo's POINT returns a zero if the SET position is black and the number (1–8) representing the color of a lighted position.

```
10 CLS0
20 A$=CHR$(205)+CHR$(199)+CHR$(2
00)
30 B$=CHR$(204)+CHR$(196)+CHR$(2
00)
40 PRINT@271,A$;
50 PRINT@303,B$;
60 GOTO60
```

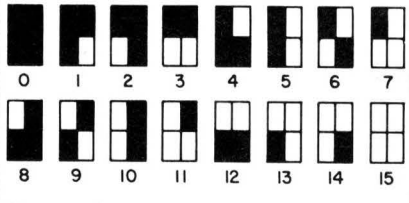
Program Listing 8

```
10 CLS
20 POKE15360,191
30 GOTO30
```

Program Listing 9

```
10 CLS0
20 POKE1024,207
30 GOTO30
```

Program Listing 10



Shape numbers

graphics character = 128 + shape number + color number

Color Numbers	
Green	0
Yellow	16
Blue	32
Red	48
Buff	64
Cyan	80
Magenta	96
Orange	112

Fig. 2. The Color Computer's CHR$ and POKE Graphics Characters

Address	Keys							
14337	@	A	B	C	D	E	F	G
14338	H	I	J	K	L	M	N	O
14340	P	Q	R	S	T	U	V	W
14344	X	Y	Z					
14352	0	1	2	3	4	5	6	7
14368	8	9	:	;	,	-	.	/
14400	ENT	CLR	BRK	↑	↓	←	→	SPC
14464	SHIFT							
	1	2	4	8	16	32	64	128

(Memory contents when key pressed)

Fig. 3. The B&W Keyboard Matrix

Address	Keys							
338	@	H	P	X	0	8	ENTER	
339	A	I	Q	Y	1	9	CLEAR	
340	B	J	R	Z	2	:		
341	C	K	S	↑	3	;		
342	D	L	T	↓	4	,		
343	E	M	U	←	5	-		
344	F	N	V	→	6	.		
345	G	O	W	SPACE	7	/		
	−1	−2	−4	−8	−16	−32	−64	

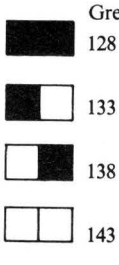
(Change of contents)

254	253	251	247	239	223	191

(Contents if pressed)

No key pressed, contents = 255

Fig. 4. The CoCo Keyboard Matrix



Yellow	+16
Blue	+32
Red	+48
Buff	+64
Cyan	+80
Magenta	+96
Orange	+112

Fig. 5. Color Video Locations

You must write lines that PEEK at the keyboard to see if a key is pressed. Figure 3 is the B&W keyboard's matrix, and Fig. 4 shows the Color Computer's.

Program Listing 13 is a B&W computer line to convert to color. Use Fig. 3 to determine which key Listing 13 needs. The row and column representing 14,337 and eight in Fig. 3 meet at C.

Figure 4 shows that pressing C on the CoCo keyboard results in memory location 341 containing 254. Thus, Program Listing 14 does for the CoCo what Listing 13 does for B&W. Sorry, there's no formula; you just need to know both machines.

When converting a program for color, you can add sound and joystick commands. Use whatever you have available to enhance the program.

And now, to change the topic. For those of you with Extended Color Basic, I'll look at semigraphics mode 8. This mode uses memory locations 1,024–3,071, so you need an Extended Color Basic machine with PCLEAR 1 or more graphics pages reserved or the video will overwrite your program.

Lines 10–40 of Program Listing 15 set up semigraphics mode 8. I got these lines from William Barden Jr.'s book, *Color Computer Graphics* (Ft. Worth, TX: Radio Shack, 1982, catalog no. 62-2076).

To turn the screen black, lines 50–60 POKE a 128 into all video locations. If you don't like black, select another color from Fig. 5.

Figure 6 is the video map. There are 32 positions across and 64 positions down. Each position has a left and right block similar to CHR$ graphic's division into four cells. Figure 5 lists the numbers used to color these blocks. The left and right block cannot be different colors unless one of the colors is black.

Listing 15 displays each pattern on the screen. To display pattern 154 onto the second row and three positions to the left, use the command POKE1058,154.

Program Listing 16 is my semigraphics mode 8 rendition of a man driving a car. Basic's speed makes the ride a little jerky.

Next month: a final exam on the full-size CoCo and the graduation ceremony. Don't be late. ■

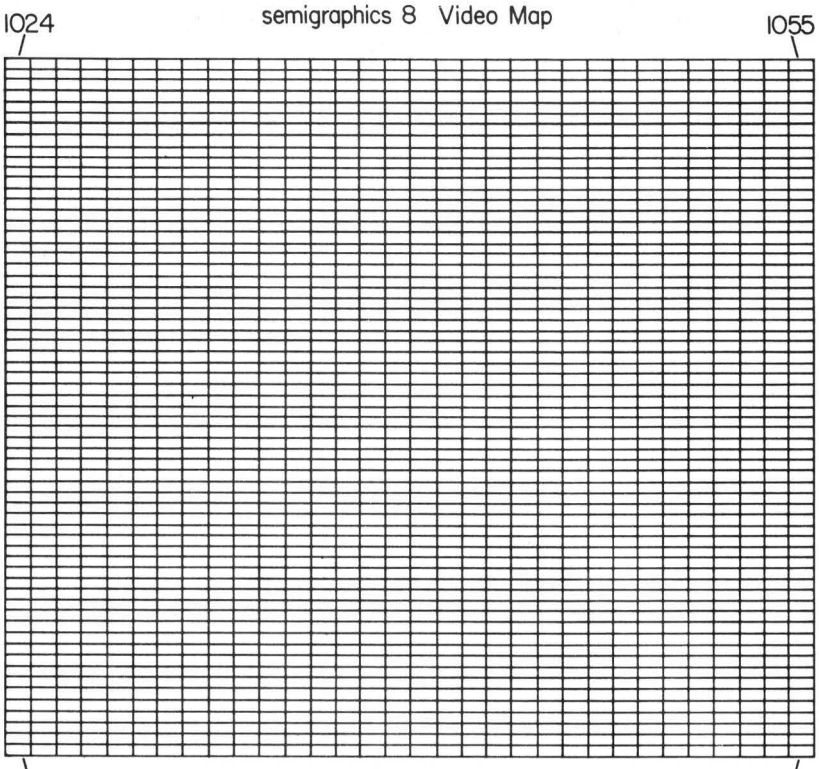


Fig. 6. Video Map

```
10 CLS
20 FOR X=0TO127
30 SET(X,0):SET(X,47)
40 NEXT X
50 FOR Y=0 TO 47
60 SET(0,Y):SET(127,Y)
70 NEXT Y
80 GOTO80
```

Program Listing 11

```
10 CLS0
20 FOR X=0 TO 63
30 SET(X,0,1):SET(X,31,1)
40 NEXT X
50 FOR Y=0 TO 31
60 SET(0,Y,1):SET(63,Y,1)
70 NEXT Y
80 GOTO80
```

Program Listing 12

```
10 CLS
20 IF PEEK(14337)=8 THEN 30 ELSE
 20
30 PRINT"FOUND CORRECT KEY"
```

Program Listing 13

```
10 CLS0
20 IF PEEK(341)=254 THEN30 ELSE
20
30 PRINT"FOUND CORRECT KEY"
```

Program Listing 14

```
10 A=PEEK(65314)
20 POKE 65314,(A AND 7)
30 POKE 65476,0:POKE 65475,1
40 POKE65472,0
50 FORA=1024 TO 3071
60 POKE A,128:NEXTA
65 P=1024
70 FOR B=0 TO 112 STEP 16
80 FOR A=128 TO 143 STEP 5
90 POKE P, A+B:P=P+2
100 IF P=1056 THEN P=P+32
110 NEXT A,B
120 GOTO120
```

Program Listing 15

```
10 A=PEEK(65314)
20 POKE 65314,(A AND 7)
30 POKE 65476,0:POKE 65475,1
40 POKE65472,0
50 FORA=1024 TO 3071
60 POKE A,128:NEXTA
70 FORA=1344 TO 1370
80 POKEA+1,128:POKEA+2,149
90 POKEA+32,128:POKEA+33,143:POK
EA+34,143:POKEA+35,143
100 POKEA+64,128:POKEA+65,133:PO
KEA+66,128:POKEA+67,138
110 FORT=1TO50:NEXTT
120 NEXTA
130 GOTO130
```

Program Listing 16

Send Your Child on the Rainbow Quest.

Fiction, Fantasy, and Computer Adventure for the Color Computer

Rainbow Quest will take your child on a space adventure of the future. The planet Rainbow is a faraway land of ancient and mysterious cities, mazes, and puzzling events for young readers to discover. **Rainbow Quest** by Richard Ramella is a book-and-cassette adventure for the Radio Shack Color Computer. Children read and play along as Molly and Sam meet pirates, robots, and strange creatures as they make their way across the planet Rainbow. To reach their goal, they must survive on their own and face the challenges they meet along the way. Readers will help Molly and Sam find their way through dark and confusing mazes, solve word and number puzzles, and conquer invaders in arcade-style games.

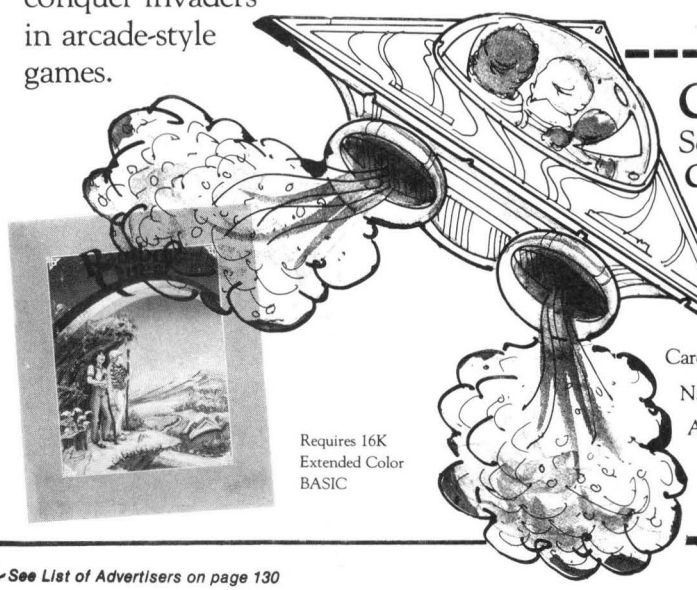
Each obstacle they meet is a program, on the **Rainbow Quest** cassette, ready to load and run.

Rainbow Quest has 25 programs in all. Book and cassette are sold together in a protective storage binder with complete instructions. Each **Rainbow Quest** package for the Color Computer is $24.97.

To order **Rainbow Quest**, call toll-free for credit card orders, 1-800-258-5473. (In New Hampshire, call 924-9471.) Or mail your order with payment or complete credit card information to: Wayne Green Inc., Attn. Book Sales, Peterborough, NH 03458.

Rainbow Quest by Richard Ramella
Illustrated by Coni Porter
ISBN 0-88006-064-6 BK7391
7 by 9 book, cassette of 25 programs
softcover, spiral-bound, 128 pp.

331



Requires 16K Extended Color BASIC

Color my child's imagination!

Send me _____ copies of **Rainbow Quest** for the **Color Computer** (BK7391) at $24.97 per package.

(Include $2.00 per package for shipping and handling.) ☐ Payment Enclosed

☐ MasterCard ☐ VISA ☐ AMEX

Card #_____ MC Bank #_____ Exp. date_____

Name _____ Signature _____

Address _____

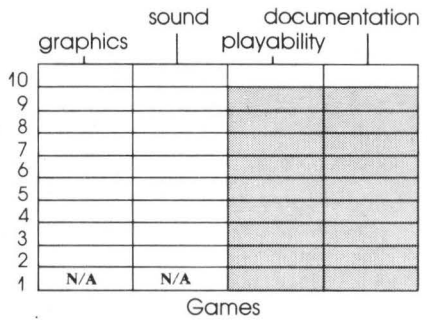
City _____ State_____ Zip_____

Wayne Green Books, Peterborough, NH 03458

343B4Q

See List of Advertisers on page 130

REVIEWS



Adventure in Wonderland
Prickly-Pear Software
9234 E. 30th St.
Tucson, AZ 85710
32K
$24.95, cassette $29.95, disk

by Beth Norman

You are in the east end of a hall. You see a three-legged glass table. On the table, you see a bottle labeled "Drink me," a scrumptious-looking cake, and a sharp key. You obediently open the bottle and sip. Suddenly, you feel all mimsy and begin to shrink. The Cheshire Cat warns that you will shrink down to nothing if you continue....

CONTENTS

edited by Mark E. Reynolds

Prickly-Pear's new Adventure in Wonderland puts you in Lewis Carroll's magical land and lets you share what Alice found there. The booklet accompanying the game claims that it is the best text adventure available for the CoCo. That may just be true. (Did anyone ever claim to have the second-best?) It is certainly one of the biggest programs around; it very nearly fills a 32K computer.

The booklet also explains that the game has a vocabulary of hundreds of words, and responds to inputs of entire sentences. Sit up straight, don't abbreviate, and use proper punctuation when speaking with the inhabitants of Wonderland. After all, as the instructions say, "Don't they have schools where you come from?"

The program is never at a loss for words. Playing it reminds me of "talking" to Shrink, Eliza, or one of the other artificial-intelligence demonstration programs.

In the adventure, you play Alice, and follow the White Rabbit through a rabbit-hole in the TV screen. A slight variation from the original tale there—curiouser and curiouser. After falling through the hole, you find yourself in a long hall and at the beginning of your quest.

The Cheshire Cat, your constant companion, babbles incessantly and can keep you from making progress. This gets annoying, especially when you are on the verge of entering the White Rabbit's cottage or finding your way out of the Tulgey Wood. But be persistent; the puss eventually shuts up. Besides, some of the things he says might prove useful.

All the characters in Wonderland are great chatting partners. They often give you hints or puzzles to help you achieve your three goals: to become a queen, to capture the Snark, and to return home. For example, here is a typically informative Wonderland sign:

> Seek it with thimbles
> Seek it with care
> Pursue it with forks and hope
> Threaten its life with
> A sharp double dare
> Charm it with smiles and soap

Adventure in Wonderland has features taken from all three of Lewis

HOT CoCo Review Policy

HOT CoCo's Reviews section is a reader service. We feel that it is our principal responsibility to give our readers a competent, objective analysis of those new products that are of reasonable interest to Color Computer users.

According to our intention to maintain objectivity, we do not permit advertisers to influence a product review. Whether a manufacturer advertises heavily in *HOT CoCo* or not at all has no bearing in our decision to review or not to review a product, or to publish a favorable or unfavorable assessment.

All reviews must be arranged through the review editor. Reviewers are encouraged not to contact the manufacturer of a product they are reviewing, unless they have a problem. It is solely the review editor's responsibility to solicit material and arrange for its review. *HOT CoCo* does not accept unsolicited reviews.

A manufacturer may send unsolicited copies of his product to the review editor. If this editor feels the product is worthy of review, he will arrange it.

During the lead time between the receipt of a review and its publication, it is possible for a manufacturer to introduce a new version of the product. In such cases, the manufacturer should inform the review editor in writing of the changes.

Manufacturers who disagree with the information in a published review may respond in writing to the review editor.

Carroll's beloved works: *Alice in Wonderland, Through the Looking Glass*, and *The Hunting of the Snark*. The game's characters are consistent with the originals. It's just plain fun to wander in the garden of talking flowers, and to meet with such old favorites as the Mock Turtle, the Gryphon, Tweedles Dee and Dum, the Duchess, the Caterpillar, and all the rest.

Speaking of wandering, Wonderland is seemingly endless—and not very logically laid out. It's helpful to draw a map, but you can't always take it too seriously.

One terrific feature: there are no deaths in Wonderland. Supposedly, there are always at least three ways out of any situation, no matter how perilous. Even after being swallowed by the Jabberwock (it happens to the best of us), you can still escape. A hint: take the Cheshire Cat's advice about whether crying ever helped anyone in Wonderland. The old cat sometimes comes through in the clutch.

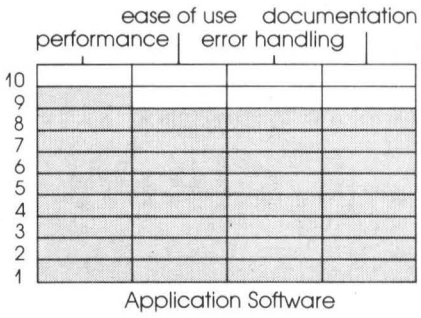
The game probably won't be half as much fun if you're not familiar with the characters. The Prickly-Pear booklet (which is loaded with those charming John Tenniel woodcuts from the original books) mentions that folks who get really stumped just might want to go to a library and read some of Mr. Carroll's works. Nicely done.

Now for a confession: I have never completed the adventure. In fact, I have only reached one of the three goals. Every time I assemble most of

the Snark-hunting tools, the Bandersnatch appears out of nowhere and takes some of them.

No matter. I've been having too much fun talking to Humpty Dumpty, the uffish sheep, and all the rest. Besides, you wouldn't want a 32K adventure to be a pushover, would you? My only regret is that there is no way to make a tape copy of a game in progress so you can pick up at the same point later.

Adventure in Wonderland is the best all-text game I've ever played on my CoCo: complex, complete, and funny. I recommend it highly. ■



Statgraf V 1.0
Sugar Software
2153 Leah Lane
Reynoldsburg, OH 43068
32K, Extended Color Basic
$24.95, cassette
$29.95, disk (same program on either medium)

by Scott L. Norman

Dennis Zaebst's Statgraf is a fine addition to the Color Computer's growing library of graphics-oriented mathematics programs. It performs linear regression analysis, a standard means of investigating possible relationships between measurable quantities, and produces graphics interpretations of the data and quantitative information tables to show the effect of your analysis.

Statgraf carries out many of the statistical computations used to see if one quantity (the dependent variable) can be meaningfully expressed in terms of another (the independent variable). It includes data-transformation routines to simplify any apparent functional dependences, and a graphing package that helps you see how the analysis is progressing.

Statgraf features several levels of menus. "Are you sure?" prompts give you a second chance before you do anything drastic. You can type in data as pairs of observations, or edit stored data files. The program can read data from tape or disk, and algebraically transform and sort it according to the independent variable.

The graphing routine can connect the data points or leave them isolated in a scatter diagram, and can display the regression line (the least-squares fit to the data) and the 95-percent confidence limits.

It is also possible to obtain both video and printed reports of several quantities of statistical interest, as well as a graphics display of residuals determined from the least-squares fit. Statgraf can plot many data sets on one graph, and save both the numerical data and the graphs themselves on tape or disk.

A word of caution: Although it is not difficult to use, Statgraf does have a few subtleties. The manual is packed with information, and you should read it carefully. What I first thought were bugs in the program turned out to be the results of my skimming through the documentation too quickly.

It was a pleasant surprise to find that my tape copy of Statgraf contained the routines for both tape and disk I/O. The program tests for a

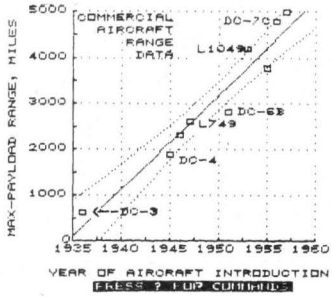


Fig. 1. A Graph Composed with Statgraf

CoCo disk controller, and if it finds it, branches to the disk routines at the appropriate points. You can also copy the tape program to disk.

Statgraf does not include the screen-print driver necessary to print out statistics and line graphs. Instead, you are advised to use one of the commercial routines written in relocatable code. The documentation supplies the details of interfacing the two programs.

To use Statgraf, enter your data. The program sorts it and computes the various statistical measures. Then it prompts you to set up the graph.

You have a choice of a circle, a triangle, a square, a small point, or any uppercase letter as a plotting character to represent your data points. You are then asked to determine the maxima and minima for each variable and set the scales on the two axes accordingly.

Statgraf divides each axis into 10 parts and labels the alternate tick marks—points not explicitly covered in the documentation. Then you label the axes of the graph (numbers and uppercase letters only).

In the labeling option, you have complete freedom to place additional text anywhere on the graph, using a special full-screen editor; there is no guesswork involved in positioning the labels. The editing commands seem strange at first, but they are all covered in a Command Reference Chart at the back of the manual.

The Statistics menu asks for estimates of the slope and intercept of a straight line to which the program fits the data. Although there is value in using your best estimate if you have prior information about the data, you can enter any two numbers. The program calculates the slope and intercept of the regression line from the actual data.

It also computes the following quantities:

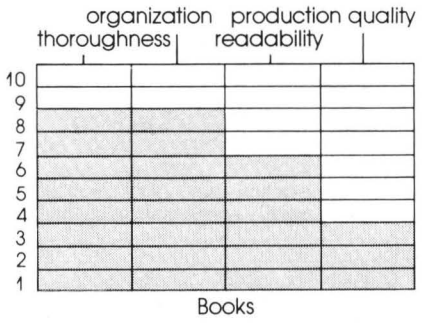
- the 95 percent and 99 percent confidence limits,
- the probability of significance for the two-sided t-test,
- the coefficient of correlation between x and y values, and
- the coefficients of determination and alienation.

If the coefficient of determination is too low, you might want to try a transformation of coordinates.

Statgraf's transformations are cumulative. It is possible to continue to work with a transformed data set, adding a constant, taking logarithms, and so on.

It is also possible to interpolate an x value from the keyboard and have the program compute the corresponding value of y.

I appreciate the freedom to tinker that this program gives you. It is modestly priced, performs well, and encourages you to experiment. That is one of the things mathematical software is supposed to do. It's not a package that you can use carelessly, but if you do exercise the proper skepticism about statistics, I highly recommend Statgraf. ■



Color Quick Reference Guide
Tandy/Radio Shack
1400 One Tandy Center
Fort Worth, TX 76102
Catalog No. 26-3194
$4.95

by Kevin W. Davidson

Complete and convenient, *Color Quick Reference Guide* is a single publication that gives a technical reference for all major Radio Shack Color Computer software. You will find Color Basic, Extended Color Basic, Disk Color Basic, and EDTASM+ material condensed from four separate manuals into one 71-page, pocket-size reference. Unfortunately, it also contains some serious mistakes.

The book was a great idea. Thanks to the format, you can hold it easily at the computer, and the index tells you exactly where to locate what you need. Each of the three Basics is separated into commands and functions. You also get a reference section with special keys, graphics codes and an ASCII code table. You find all the commands of EDTASM+ except W, and a nice section on ZBug. It also includes a list of 6809 instructions and Assembly pseudo-ops. A memory map, ROM addresses, line-printer variables, pinouts for the RS-232C, cassette and joystick ports, and a glossary complete the volume. Everything is there except a disk memory map, the RND function, and the DRAW subcommands.

> *"Everything is there except a disk memory map, the RND function, and the DRAW subcommands."*

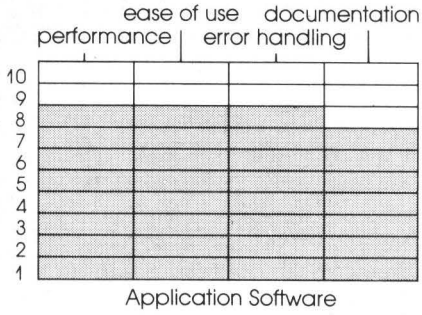
Although *Color Quick Reference Guide* is a great idea, inexcusably shoddy proofreading spoils it. In my first reading, I found 39 major mistakes. Many of the examples gave SN errors when I tried them. Extra parentheses and spurious quotes abound. The JOYSTK subscript numbers are backwards. POS device numbers are wrong, and you find functions (INSTR and MKN$) listed as commands and some Color Basic statements (CLOADM) listed as Extended Color Basic only.

Three of the eight ROM addresses on page 19 are wrong and two (GIVABF and INTCNV) are missing. The illustration for PMODEs 2 and 3 on page 27 shows the graphics elements stacked vertically rather than horizontally, as they should be.

Things improve after page 27 as a reflection of the quality source material, the EDTASM+ manual. Pages 1–27, however, represent a mindless transcription of the worst editions of the early Color Computer manuals, complete with errors. It seems that no one familiar with the Color Computer read the final draft prior to printing.

After making the numerous corrections necessary, I have found myself using *Color Quick Reference Guide* often. You might be better off to buy the Nanos Reference Card if you don't have disk or EDTASM+, but otherwise, despite its errors and problems, you should find this guide helpful. ■

↙See List of Advertisers on page 130



Humbug (MC-10 version)
Star-Kits
P.O. Box 209-H
Mt. Kisco, NY 10549
$29.95, cassette

by John S. Cullings

Sometimes it's a real pleasure to review a product of exceptional usefulness or value. Humbug, a machine-language monitor program for the MC-10, is just such a product. And, it includes a semidisassembler—a very useful addition.

Op-Code	Mnemonic	Function
08	INX	increment X
20	BRA	branch always
26	BNE	branch if not equal
39	RTS	return from subr.
5A	DECB	decrement B
7D	TST	test memory byte
8C	CPX	compare X with #
BD	JSR	jump subroutine
C6	LDAB	load B with #
CE	LDX	load X with #
E7	STAB	store B (indexed)
FE	LDX	load X from memory
FF	STX	store X to memory

Table 1. MC6803 Op-Codes and Mnemonics

Table 1 contains a list of MC6803 mnemonics and operation codes referenced in this review. Table 2 is a partial memory map of the MC-10.

A monitor program is most useful for machine-language program development. It lets you view any memory location and modify any read/write memory location. Most monitors let you display, fill, move, or execute blocks of memory, and they should also offer single-step operation of machine-language programs to simplify debugging.

Humbug contains all these monitor functions, plus many others. Features such as Find, Compare, Semidisassemble, Analyze, and Save to Tape make this program an exceptional value.

```
4481   2B       FCB   2B       + (after HUMBUG)

44DE   4B85     FDB   4B85     new AD address

4B8E   FF460B   STX   460B     save start addr
4B91   2003     BRA   +3       skip next commd

4B9F   7D45DA   TST   45DA     no operation
4BA2   C604     LDAB  #4       4 hex bytes
4BA4   FE4611   LDX   4611     memory pointer
4BA7   BD45CD   JSR   45CD     output 2 hex
4BAA   5A       DECB           done?
4BAB   26FA     BNE   -6       no
4BAD   FF4611   STX   4611     memory pointer
4BB0   BD45DA   JSR   45DA     output a space
4BB3   C604     LDAB  #4       4 hex bytes
4BB5   FE4611   LDX   4611     memory pointer
4BB8   BD45CD   JSR   45CD     output 2 hex
4BBB   5A       DECB           done?
4BBC   26FA     BNE   -6       no
4BBE   FF4611   STX   4611     memory pointer
4BC1   200B     BRA   +0B      go to ASCII dump

4BD1   7D45D6   TST   45D6     no operation
4BD4   BD45DA   JSR   45DA     output a space
4BD7   C608     LDAB  #8       8 ASCII chars.

4BFC   2098     BRA   -68      go to hex dump
```

Program Listing. Hex with ASCII Dump (Use only with the Humbug version at $4400.)

> *"In MC-10 terms, Humbug is a large program."*

However, Humbug isn't written in position-independent code (because the MC6803 microprocessor doesn't have the appropriate instructions to make it easy to do so), so you can only load and execute the program at one address. Therefore, you get three different versions of Humbug so you can execute it in three different memory areas.

In MC-10 terms, Humbug is a large program. At 2,672 bytes, it leaves only about 500 bytes for another machine-language program. This could be a real problem for a 4K MC-10.

You might need to expand to 16K to develop or modify large programs. But you can overwrite (destroy) some of the less important Humbug functions to gain a little more working memory.

Humbug is easy to use. Its help command lists all function abbrevia-

Addresses	Use
0000–001F	MC6803 registers
0080–00FF	direct page CPU memory
4000–41FF	TV screen memory
4200–434F	system memory
4350–4F8F	Basic and user memory
4F90–4FFF	stack memory (4K system)
5000–8FFF	additional 16K memory opt.
9000–BFFF	graphic and sound control
E000–FFFF	Basic ROM
FFDC–FFED	Basic I/O pointer table

Table 2. MC-10 Memory Map

tions. It has error messages and a simple pause/print option, and you can set or examine up to four breakpoints for program debugging.

Humbug comes with a well-written manual that includes a fully documented source code of the program, but the comments are chopped off after 21 characters.

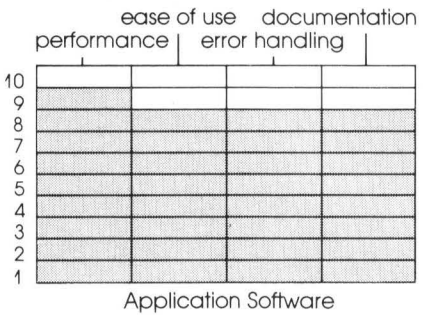
There are 13 useful machine-language subroutines that are documented and contained in a jump table at the beginning of the program. These subroutines can simplify the input and output of hexadecimal and text data. The addresses of these subroutines depend upon which of the Humbug versions you're using.

The Humbug manual is well done, but it lacks examples of many of its functions. A beginner could profit from more examples.

There are two things to remember when debugging a program. Once Humbug reaches a breakpoint, you must remove that point before continuing or single-stepping. And the register-examine (RE) command displays the A and B registers in reverse order, contrary to the example shown in the instruction manual. This makes viewing the D register more difficult because the 2 bytes are shown reversed.

Overall, Humbug is a powerful machine-language monitor. You can correct any deficiencies because Star-Kits includes the commented source code. As an example of an improvement, I've included a short patch to the ASCII and hex dump option (see Program Listing).

For viewing large machine-language programs, like the Basic ROM, I prefer to dump both hex and ASCII code together. But Humbug only lets you choose one at a time. The Program Listing combines both the AD and HD commands into one command. The source-code listing helped make this modification simple. ■



Commterm
Star-Kits
Box 209-H
Mt. Kisco, NY 10549
MC-10, modem
$19.95, cassette

by Douglas L. Swank

Commterm is a dumb-terminal program for the 16K CoCo or any MC-10. I looked at the MC-10 version, and found it a superb application for a machine with limited memory. This typical dumb-terminal program uses 1,279 bytes.

It reads the incoming copy, puts it on the screen, sends out the keyboard data, and that's all. No bells, no whistles, just load the program, connect a modem, and you're on the air. The program is in machine language and runs via the CLOADM/EXEC commands that are in every MC-10, but not mentioned in the manual.

"...the simple Commterm program is an ideal email message handler, and that is my main use for a portable terminal."

Bulletin-board operators and time-share services have pretty much standardized their data formats, and Commterm's writers have chosen the setup format to match. I tried eight hobby bulletin boards, CompuServe, The Source, Telemail, and the IBM 38XX and DEC-VAX where I work. The program worked well with all systems.

The screen display is the usual 32-by-16 characters with no word-wrap, but you can't have everything in a 4K machine. (Most BBSs and large systems have a setup section to do word-wraps according to your specified screen width.)

The only setup choices are full or half duplex. Half duplex permits communication with a simple terminal, such as another MC-10. All the large systems now have echo to the user, permitting use of full duplex.

You use the right arrow as the control key for sending control characters, and the usual shifted zero to switch to lowercase.

A terminal program is a good application for the MC-10, because at current sale prices, you could assemble a compact traveling kit at less than one-fourth the cost of a Model 100 or similar unit. The 4K MC-10 does not leave room for storage buffers or exotic features, but the simple Commterm program is an ideal email message handler, and that is my main use for a portable terminal.

If you keep your email messages to memo size, you should find the MC-10/Commterm package an effective and inexpensive link to a big system.

Although their ads say that Commterm sells for $19.95, information in the package suggests that Star-Kits is using what could be called a pass-the-hat sales approach.

Instead of spending money on duplication, shipping, and advertising, the manufacturer urges you to copy the program and pass it out to your friends. Then you can each send in what you feel the tape is worth.

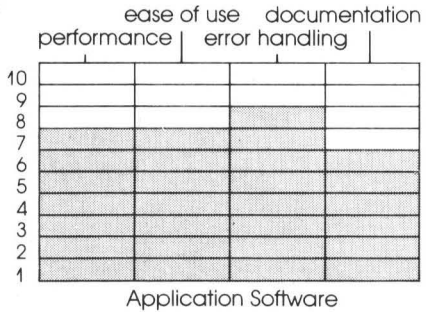
Or, you can send Star-Kits a blank cassette and a self-addressed envelope with three 20¢ stamps on it. They'll

send you the program, three pages of instructions, extra instructions, and permission to copy Commterm and give it to others.

The program and the appeal for contributions both worked for me. After using my review copy for a few days, I decided to keep it and fired off a check. (I like to encourage good writers in the hope that they will write even better stuff in the future.)

The idea is sort of like passing the hat, and from reports I've seen, it may be cheaper for users and more profitable for authors than the usual method. It certainly seems to be a good way to market small, low-priced programs.

Commterm works well, and is useful software for every traveling MC-10 owner. I find the sales scheme interesting, and hope to see more programs offered this way in the future. I recommend the MC-10 version for any email user. ■



Data Doctor
Superior Graphics Software
Box 451
Canton, NC 28716
32K, disk
$49.95

by Leigh H. French, IV

Data Doctor lets you read and modify disk files saved in the ASCII format. It is written completely in Basic, and though configured for 32K, it will run on a 16K CoCo. Despite being menu driven, Data Doctor uses existing Radio Shack DOS commands with no machine-language enhancements as utilities.

The main menu gives you 10 options: load data file; save data file to disk; retype, edit, insert, and delete data lines; print data to screen or printer; return to menu; and quit program. Since there is no directory command on this menu, you either must know the file name you want to load, or you must hit break and load your target disk to get its directory.

Once you have the file name, and you are sure it's in ASCII format, be sure to type it in correctly or the program returns you to DOS Basic. If you get a disk error, the manual tells you to type GOTO 27 to reenter the program. This gives you a second menu of error types and possible actions. This second menu lets you see the target disk's directory. With the file in the system, the rest of the commands are easy to handle, but having the documentation handy, though it has its deficiencies, is helpful.

Data Doctor is a good idea, but it is incomplete. The documentation lacks a good tutorial or a sample file on which to test the utility's functions. It also has editorial errors: The documentation tells you to enter RUN"DATADOC" to start the program. The actual file name is DATA/DR.

"Data Doctor is not a machine-language program that enhances RS DOS."

Another problem with Data Doctor is its price. It is not a machine-language program that enhances RS DOS. It does not reside with other programs; you must load it each time you want to use it. And it is limited to ASCII files, usually the alternative storage format in RS DOS because of its inefficient use of disk space. It does not stand up, price-wise, to other programs that have similar capabilities such as Telewriter and the Tool Box.

Also, this program and its documentation assume a degree of knowledge found in the hacker. Novice users might have trouble getting Data Doctor to work.

With its problems and poor documentation, Data Doctor is probably not your best choice. Save your money and wait for a better version of this program. ■

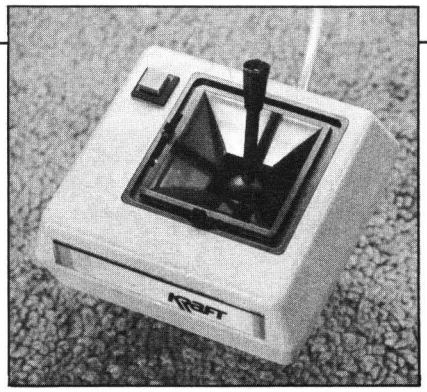
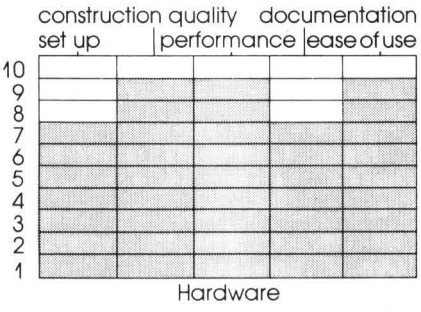


Photo 1. The Kraft Joystick



Kraft Joystick
Kraft Systems
450 W. California Ave.
Vista, CA 92083
Kraft product no. 820-003 KJS-01T
$65 each

by Richard E. Esposito

After wearing out my second set of Radio Shack joysticks, I decided to consider some of the available alternatives. The Kraft stick was appealing because, unlike most other sticks, it uses potentiometers, as opposed to leaf switches, which only indicate direction. Leaf switches cause problems with software such as Radio Shack's Project Nebula, which needs other information besides direction from the joystick.

The Kraft joystick does seem worth the money. It is sturdy, and (if you save your sales receipt) Kraft honors a one-year warranty against defects in manufacturing and materials. They use a high-quality cable that, on my machine, has helped cut down on some of the RFI.

The metal stick is unlikely to break off in normal use, and the fire button is of good quality. The potentiometers give very precise control, and as a result, I've scored considerably higher in many video games.

VIP™ Library

ANNOUNCING The VIP Library™ With a Terrific Sale!

Nelson Software Systems is now Softlaw Corporation, under new management. Our Super "Color" Library programs have also undergone a name change. All programs are the same unbeatable Super "Color" Library programs you've heard so much about, but with new VIP names. To introduce our **VIP Library** we are having a special sale on the following pages. Our low prices for such high quality can't be beat so get started today!

Official Library of Software for the TANO Dragon

(Sold for the TANO Dragon only by TANO Microcomputer Products, Corp. and its distributors)

The Library Concept

State of the Art, Quality, Integrity, Compatibility and Affordability. Five things good software must possess. Five things that epitomize the **VIP Library™**. Each program is the diamond of its class, true excellence. These programs are first in features, first in power, first in memory, and all are affordably priced. And for your convenience all disk programs can be backed up.

State Of The Art

All **Library** programs are written in machine code specifically for the Color Computer, to work without the interference of a separate operating system such as FLEX. From this comes speed and more work-space for you. Unlike other programs for the Color Computer which are said to be 64K compatible, **VIP Library™** programs are not limited to between 24 and 30K of workspace in 64K. **Library** programs have Memory Sense with BANK SWITCHING to fully use all 64K, thus giving up to 51K with a disk version and up to 53K with a tape version.

Easy To Use

Each **Library** program was carefully designed to be extremely easy to use. Built-in on-screen help tables are at your fingertips, as are menus of all kinds. Every effort is made to use logical, intuitive and easy-to-remember commands. The manuals have been thoughtfully prepared to cover every aspect of the program, and they have complete tutorials to get you going right away. We set the standard!

Lowercase Displays

State-of-the-Art graphics allow instant use of four display colors, and eight lowercase displays featuring descending lowercase letters. You can select from 51, 64 or 85 columns by 21 or 24 lines per screen, with wide or narrow characters in the 64 display. These screens provide a pleasant and relaxing way to perform your tasks, with as much text on the screen as is possible. Each program is easy to learn and a joy to use. We take pride in the stringent testing done to make these programs perform flawlessly. Every feature, every convenience, sleek, simple and elegant.

". . . PICTURE getting your instantaneous investment report over the phone, using it in your spreadsheet calculation, generating a report, and writing a memo including that report and data from your database with your word processor, and all this with VIP Library™ programs . . ."

Total Compatibility

All **Library** programs are compatible. Transfer and use of files between programs is easy and carefree. What's better, when you have learned one program the others will come easy. And every program is the best of its kind available.

The Library Programs

For your writing needs is the **VIP Writer™**, and its spelling checker, the **VIP Speller™**. For financial planning and mathematical calculations you can use the **VIP Calc™**. To manage your information and send multiple mailings there is the **VIP Database™**. For sending all these files to and from home or the office and for talking to your friends you can have the **VIP Terminal™**. Finally, to fix disks to keep all your **Library** files in good repair we offer the **VIP Disk-ZAP™**.

Mini Disk Operating System

The Disk versions each have a Mini Disk Operating System which will masterfully handle from 1 to 4 drives. It offers smooth operation for such features as the ability to read a directory, display free space on the disk, kill files, save and automatically verify files, and load, rename and append files. **Library** programs simply do not have the limitations of BASIC.

Professionalism

The **Library** comes handsomely bound in gold-embossed, padded leatherette binders to grace your work area with the professionalism it deserves. Welcome the **VIP Library™** into your home and office.

A description of each of the **Library** programs, with the special sale price, is contained in the following pages. Please indulge!

VIP Writer™

(Formerly Super "Color" Writer II)
By Tim Nelson

RATED TOPS IN RAINBOW, HOT COCO, AND COLOR COMPUTER MAGAZINE

The Official Dragon Microcomputer Word Processor†

The most powerful and easy-to-use word processor is available in the showpiece and workhorse of the **Library: The VIP Writer™**. Because of its undisputed superiority over all Color Computer word processors, it was selected by Dragon Data Ltd. of England and TANO in the U.S., to be the Official Word Processor for their line of Dragon microcomputers.

The result of two years of research, the **VIP Writer™** offers every feature you could desire from a word processor. It is the most powerful, fastest, most dependable and most versatile. With the hi-res display, workspace and compatibility features built into the **Library** the **Writer** is also the most usable.

". . . Nearly every feature and option possible to implement on the Color Computer. The design of the program is excellent; the programming is flawless . . . Features for the professional, yet it is easy enough for newcomers to master . . . Certainly one of the best word processors available for any computer . . ." October 1983 "Rainbow"

"Word processing with VIP Writer is like driving a high-performance vehicle . . . This Ferarri of a package has more features than Telewriter, Easywriter (for the IBM PC), or Applewriter." October 1983 "Hot CoCo"

The **Writer** will work with you and your printer to do things you always wanted to do. Every feature of your printer can be put to use, every character set, every graphics capability at any baud rate, EVEN PROPORTIONAL SPACING. All this with simplicity and elegance. You can even automatically print multiple copies.

Although all versions feature tape save and load, the disk version provides the Mini Disk Operating System common to the whole **Library**, plus disk file linking for continous printing.

Professional features of particular note:

- Memory-Sense with **BANK SWITCHING** to fully utilize 64K, giving not just 24 or 30K, but up to 61K of workspace with the rompak version and 50K with the disk version.
- TRUE FORMAT WINDOW allowing you to preview the printed page ON THE SCREEN BEFORE PRINTING, showing centered lines, headers, FOOTNOTES, page breaks, page numbers, & margins in line lengths of up to 240 characters. It makes HYPHENATION a snap.
- A TRUE EDITING WINDOW in all 9 display modes for those extra wide reports and graphs (up to 240 columns!).
- FREEDOM to imbed any number of PRINTER CONTROL CODES anywhere, **EVEN WITHIN JUSTIFIED TEXT**.
- Full 4-way cursor control, sophisticated edit commands, the ability to edit any BASIC program or ASCII textfile, SEVEN DELETE FUNCTIONS, LINE INSERT, LOCATE AND CHANGE, wild card locate, up to TEN SIMULTANEOUS block manipulations, word wrap around, programmable tabs, display memory used and left, non-breakable space, and headers, footers and FOOTNOTES.
- Automatic justification, automatic pagination, automatic centering, automatic flush right, underlining, superscripts, subscripts, pause print, single-sheet pause, and print comments.
- Type-ahead, typamatic key repeat and key beep for the pros, ERROR DETECTION and UNDO MISTAKE features, 3 PROGRAMMABLE functions, auto column creation, and an instant on-screen HELP TABLE.

32K (Comes with tape & disk) $59.95

†Sold as the Dragon Writer™ ONLY by Dragon Data Ltd. and its distributors.

VIP Speller™

NEW

A BRAND NEW SPELLING CHECKER!

By Bill Argyros

Spelling checkers are an invaluable aid to every writer. Habitual misspellings and typos can be found without the eyestrain, boredom and fatigue associated with endless proofreading. The **VIP Speller™** is a fast, machine-code proofreading program to correct any **VIP Library™** file. It automatically proofreads your documents against a 30,000 word stock dictionary, plus a dictionary you can create, and corrects typos or marks them for special attention. Unlike other spelling checkers, the new **VIP Speller** distinguishes between upper and lowercase letters, and it shows the misspelled word in context so you can be sure of your correction. Compatible with all CoCo word processors.

32K DISK ONLY $39.95

Lowercase displays not available with this program.

VIP™ Library

Coming Soon

VIP Calc™

(Formerly Super "Color" Calc)

TRUE VISICALC™ POWER!

By Kevin Herrboldt

- **UP TO 5 TIMES THE SCREEN DISPLAY AREA OF OTHER SPREADSHEETS!**
- **STATE OF THE ART LOWERCASE DISPLAYS**
- **MEMORY SENSE WITH BANK SWITCHING FOR UP TO 40+K in 64K!**
- **EXCLUSIVE VIDEO DISPLAY WINDOWS — EVEN UP TO 16!**
- **USER-DEFINABLE WORKSHEET — UP TO 512 COLUMNS BY 1024 ROWS**
- **WORKS WITH ANY PRINTER, EVEN LETTER QUALITY!**
- **LOCATE COMMAND TO FIND SPECIFIC NUMBERS, LABELS OR FORMULAS**
- **SORT COMMAND FOR EASY RANKING OF RESULTS**
- **ALMOST UNLIMITED PROGRAMMABLE FUNCTIONS**

VIP Calc™ is truly the finest and easily the most powerful electronic worksheet and financial modeling program available for the Color Computer. Now every Color Computer owner has access to a calculating and planning tool better than VisiCalc™, containing all its features and commands and then some, WITH USABLE DISPLAYS. Use Visicalc templates with **VIP Calc™!**

There's nothing left out of **VIP Calc™**. Every feature you've come to rely on with VisiCalc™ is there, and then some. You get up to **5 TIMES** the screen display area of other spreadsheets for the Color Computer and Memory-Sense with **BANK SWITCHING** to give not just 24, or 30, but **UP TO 61K OF WORKSPACE IN 64K!!!** This display and memory allow you the FULL SIZE, USABLE WORKSHEETS you require. You also get: User definable worksheet size, up to 512 columns by 1024 rows! * Up to **SIXTEEN VIDEO DISPLAY WINDOWS** to compare and contrast results of changes * **15 DIGIT PRECISION** * Sine, Cosine and other trigonometric functions, Averaging, Exponents, Algebraic functions, and **BASE 2, 8, 10 or 16** entry * Column and Row, Ascending and Descending **SORTS** for comparison of results * **LOCATE FORMULAS OR TITLES IN CELLS** * Easy entry, replication and block moving of frames * Global or Local column width control up to 78 characters width per cell * Create titles of up to 255 characters per cell * Limitless programmable functions * Typamatic Key Repeat * Key Beep * Typeahead * Print up to 255 column worksheet * Prints at any baud rate from 110 to 9600 * Print formats savable along with worksheet * Enter PRINTER CONTROL CODES for customized printing with letter quality or dot matrix printer * Combine spreadsheet tables with **VIP Writer™** documents to create ledgers, projections, statistical and financial reports and budgets.

Both versions feature Tape save and load, but the disk version also has the Mini Disk Operating System of the entire **Library**.

32K (Comes with tape & disk) $59.95

does not allow hi-res diplay in 32K

NEW SALE PRICES!

Check These Library Features:

- **Fully CoCo 2 Compatible**
- **Nine Display Formats: 32 by 16 51, 64, 85 by 21 or 24**
- **True Lowercase & Descenders**
- **Four Different Display Colors**
- **32 & 64K Compatible**
- **Memory Sense - Bank Switching**
- **Up to 51K Disk, 53K Tape**
- **Mini Disk Operating System**
- **Compatible With All Printers**

A SPECIAL OFFER ON THE WHOLE LIBRARY —

The entire Library, all six great disk programs, can be purchased for only **$300!**

VIP Terminal™

(Formerly Super "Color" Terminal)

RATED BEST IN JANUARY 1984 "RAINBOW"

By Dan Nelson

From your home or office you can join the communication revolution. The **VIP Terminal™** opens the world to you. You can monitor your investments with the Dow Jones Information Service, or broaden your horizons with The Source or Compuserve, bulletin boards, other computers, even the mainframe at work.

For your important communication needs you've got to go beyond software that only lets you chat. You need a smart terminal so that you can send and receive programs, messages, even other **VIP Library** files. **VIP Terminal**, the official Dragon microcomputer terminal, does much more than any other terminal and does it reliably. None can compare in features.

FEATURES: Choice of 8 hi-res lowercase diplays * Memory-Sense with BANK SWITCHING for full use of workspace * Selectively print data at baud rates from 110 to 9600 * Full 128 character ASCII keyboard * Automatic graphic mode * Word mode (word wrap) for unbroken words * Send and receive **Library** files, Machine Language & BASIC programs * Set communications baud rate from 110 to 9600, Duplex: Half/Full/Echo, Word length: 7 or 8, Parity: Odd/Even or None, Stop Bits: 1-9 * Local linefeeds to screen * Save and load ASCII files, Machine Code & BASIC programs * Lowercase masking * 10 Keystroke Multiplier (MACRO) buffers to perform repetitive pre-entry log-on tasks and send short messages * Programmable prompt or delay for send next line * Selectable character trapping * Send up to ten short messages (KSMs), each up to 255 characters long, automatically, to save money when calling long distance.

All versions allow tape load and save of files and KSMs, but the disk version also has the Mini Disk Operating System common to the **Library**.

32K (Comes with tape & disk) $49.95
16K Rompak (While they last) $49.95
(Tape does not allow hi-res displays in 16K)

Softlaw

128

9072 Lyndale Avenue So. 612/881-2777

Minneapolis, Minnesota 55420 U. S. A.

AUTHOR'S SUBMISSIONS ARE ENCOURAGED.

VIP Database™

(Formerly Super "Color" Database)

INCLUDES MAIL MERGE CAPABILITIES TOO!

By Tim Nelson

This high speed MACHINE LANGUAGE program fills all your information management needs, be they for your business or home. And it does so better than any other database program for the Color Computer, featuring machine code, lowercase screens and mailmerge capabilities. Inventory, accounts, mailing lists, family histories, you name it, the **VIP Database™** will keep track of all your data, and it will merge **VIP Writer™** files.

The **VIP Database™** features the **Library** Memory Sense with BANK SWITCHING and selectable lowercase displays for maximum utility. It will handle as many records as fit on your disk or disks. It is structured in a simple and easy to understand menu system with full prompting for easy operation. Your data is stored in records of your own design. All files are fully indexed for speed and efficiency. Full sort of records is provided for easy listing of names, figures, addresses, etc., in ascending or descending alphabetic or numeric order. Records can be searched for specific entries, using multiple search criteria. With database form merge you may also combine files, sort and print mailing lists, print "boiler plate" documents, address envelopes - the list is endless. The math package even performs arithmetic operations and updates other fields. Create files compatible with the **VIP Writer™** and **VIP Terminal™**. Unlimited print format and report generation with the ability to imbed control codes for use with all printers.

As with all other **Library** programs, the **Database** features the powerful Mini Disk Operating System.

32K DISK $59.95
64K Required for math package.

VIP Disk-ZAP™

(Formerly Super "Color" Disk-ZAP)

RAVED ABOUT IN THE APRIL 1983 "RAINBOW!"

By Tim Nelson

Your database file disk, form letter disk, or BASIC program disk goes bad. An I/O error stops loading, or even backing up of the disk. Weeks, even months of work sit on the disk, irretrievable. Now catastrophic disk errors are repairable, quickly and with confidence, using the **VIP Disk-ZAP™**. It is the ultimate repair utility for simple and quick repair of all disk errors. Designed with the non-programmer in mind, the **VIP Disk-ZAP™** will let you retrieve all types of bashed files, BASIC and Machine Code programs.

This high-speed machine code disk utility has a special dual cursor screen display to look at the data on your disk. You are able to: Verify or modify disk sectors at will * Type right onto the disk to change unwanted program names or prompts * Send sector contents to the printer * Search the entire disk for any grouping of characters * Copy sectors * Backup tracks or entire disks * Repair directory tracks and smashed disks * Full prompting to help you every step of the way * 50-plus page Operators Manual which teaches disk structure and repair.

16K DISK $39.95
Lowercase displays not available with this program.

VISA

For Orders ONLY — Call Toll Free —

MasterCard

1-800-328-2737

Order Status and Software Support call (612) 881-2777

Available at Dealers everywhere.
If your Dealer is out of stock ORDER DIRECT!

In Canada distributed by Kelly Software Distributors, LTD.

MAIL ORDERS: $3.00 U.S. Shipping ($5.00 CANADA; $10.00 OVERSEAS). Personal checks allow 3 weeks.

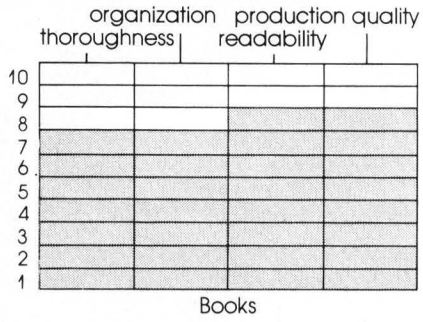
All Disk Programs are also available on 3" Diskettes for the Amdek Color AMDISK-III Micro-Floppy Disk System for an additional $3.00 each.

You can adjust the stick so that it self-centers in only the X-direction, only the Y-direction, or both, or you can make it free-floating (the only mode available with the Radio Shack sticks). You can also fine tune the center position.

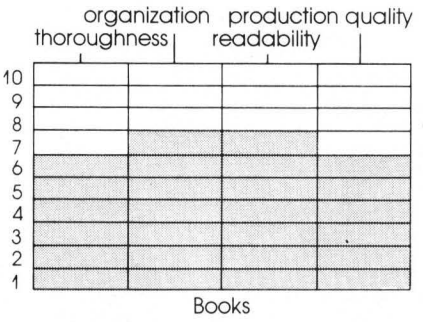
On the minus side, Kraft mounted the fire button on the top of the base, inserting a plastic plug into the button sockets on the back of the base. It would have been nice if they had supplied an additional fire button in this second socket.

The fine tuning of the center position has a tendency to require readjustment during play, and the setting of the self-centering mode is difficult at first. But if you're an arcade gamer, you'll like this stick. ■

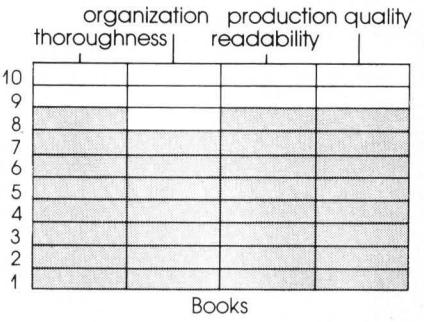
Eds. note—Since this review was written, Radio Shack began carrying the Kraft joystick for $39.95 in their 1984 catalogue. They call it the Deluxe Joystick, part no. 26-3012. It does not have a button socket on the base.



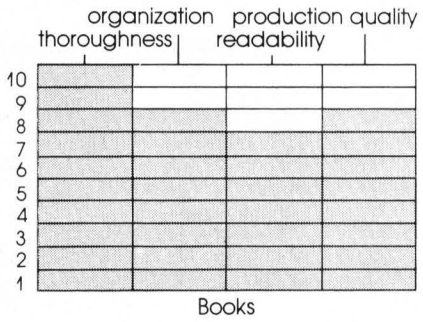
TRS-80 Color Computer Assembly Language Programming
William Barden Jr.
Tandy/Radio Shack
Fort Worth, TX 76102
Catalog No. 62-2077
$6.95, 294 pp.



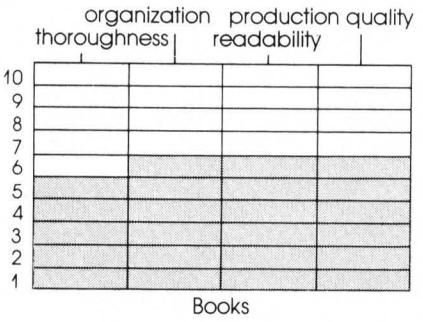
Assembly Language Graphics for the TRS-80 Color Computer
Don Inman
Reston
Reston, VA 22090
$14.95, 280 pp.



Programming the 6809
Rodnay Zaks
Sybex
Berkeley, CA 94710
$14.95, 362 pp.



6809 Assembly Language Programming
Lance Leventhal
Osborne/McGraw-Hill
Berkeley, CA 94710
$16.95, 562 pp.



The MC6809 Cookbook
Carl Warren
TAB Books Inc.
Blue Ridge Summit, PA 17214
$6.95, 162 pp.

"Leventhal's book will delight hardware and software hackers alike. If you have a question about the MC6809, look here first."

by Richard E. Esposito

These five titles represent current offerings of Assembly-language books specifically for the Color Computer or the 6809.

William Barden has aimed his book at the CoCo and deals with Radio Shack's EDTASM+ ROM pack assembler throughout. His probably outsells the other books because *TRS-80 Color Computer Assembly-Language Programming* has a low price and Radio Shack distribution.

It, like the other books in this group, is geared toward the person with some computer knowledge. Barden writes with an easy style, but, even at that, it might be difficult for beginners.

Unique to this book is its discussion of VARPTR for passing information to and from Basic when writing programs that use both Basic and machine-language USR routines.

This book is not as comprehensive nor does it present the technical material with as much detail as does Lance Leventhal's book, but considering its price and the fact that it is written specifically for the CoCo, I rate it a best buy.

Assembly-Language Graphics for the TRS-80 Color Computer is also written specifically for the CoCo, but it deals with The Micro Works' monitor, CBUG, and their editor/assembler SDS80C ROM pack.

It has a number of good sample programs that use the CoCo's color graphics and sound capabilities. It also has a chapter on using Spectral Associates' EPROM programmer.

Don Inman includes a section on using the USR function to link your machine-language programs to Basic. This book covers graphics in more detail than William Barden's book does, hence its title.

If you are using The Micro Works' software, it is certainly worth the extra

✓See List of Advertisers on page 130

cost. The Assembly-language programs in this book could prove useful to those with other software as well.

On the negative side, at $14.95 per copy, I was surprised to see dot-matrix program listings with occasional corrections in pen.

Programming the 6809 has the best introduction of this group, but it bogs down in detail. It is a good reference in that it devotes a page to each Assembly-language instruction.

This book offers more detail than do either of the previous two. Hardware hackers will like its discussions of I/O devices and I/O techniques. For the more advanced hobbyist, it has a good discussion of data structures.

6809 Assembly-Language Programming remains the 6809 "bible." The other books are not as comprehensive, nor do they approach the same level of detail.

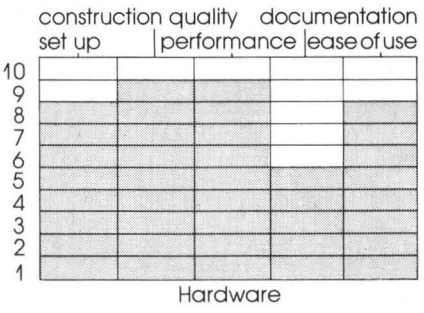
Like *Programming the 6809*, this book has a page devoted to each instruction. However, Lance Leventhal has added graphics displays of data movement as a result of an instruction and its effect on the condition-code register. It also has extensive charts and tables that are unrivaled by the others, and a listing of the instruction set by addressing mode, mnemonic op-code, and numeric op-code.

Leventhal's book will delight hardware and software hackers alike. If you have a question about the MC6809, look here first. This book is the most expensive in the group, but it is worth it. After reading any one of the other five books, you will still find this one interesting, and an invaluable reference.

The publishers of *The MC6809 Cookbook* have made an effort to keep the price low at the expense of some clarity. The detailed explanations of each instruction would be a much more effective reference tool if each started on a new page.

As texts, the other four books are better than this one. *The MC6809 Cookbook* is a reference book, but as such, it's not as worthwhile as the other two 6809 books.

Which one should you buy then? If you already have EDTASM+ or SDS80C, the first two books on the list might be best. But whichever you buy, *6809 Assembly-Language Programming* is still an excellent reference, especially if you're using FLEX or OS-9 assemblers. ■



DMP-120 Printer
Tandy/Radio Shack
1400 One Tandy Center
Fort Worth, TX 76102
Catalog No. 26-1255
$499.95

by Brian H. Alsop

The DMP-120 printer appeared to be a real buy. It had all the features I wanted, including a low price, a 132-column printout, expanded and condensed fonts, upper- and lowercase characters, dot graphics, and speed. Since it has both serial and parallel ports, you can use it in either mode with a CoCo.

"Well, the DMP-120 is a new product, and new products sometimes have bugs. I did encounter some problems, but I like the printer now and appreciate Radio Shack's cooperation in dealing with the rough spots."

Well, the DMP-120 is a new product, and new products sometimes have bugs. I did encounter some problems, but I like the printer now and appreciate Radio Shack's cooperation in dealing with the rough spots.

The Manual

The printer manual explains how to load paper and the ribbon cartridge well, but its descriptions on how to use the printer and printer graphics are only marginal. It includes typographical errors and sample problems that don't work, and it doesn't explain dot graphics very well. For example, some sample problems use CHR$ codes that don't exist.

The manual also neglects to explain how the dots of the print head are numbered (see Fig. 1). The discussion on control characters gives only a single sentence to the Color Computer: "If you have a Color Computer, read all LPRINTs as PRINT#-2,."

Hardware

The electronics consist of a microprocessor, ROM, parallel interface, serial interface, and print-head drivers. Unfortunately, the CoCo and the DMP-120 are somewhat incompatible. In two instances, the printer drops characters because it has problems communicating with the CoCo.

Radio Shack calls this a timing problem and says that it also occurs with the Scripsit ROM pack. This problem appears to be connected with the serial interface. The Color Computer probably doesn't receive the handshake signal fast enough at times to tell it to stop sending characters.

I phoned Tandy, and was told to install an updated ROM kit. The new ROM and a POKE 151,255 to add a one-second delay after a carriage return before printing the next character did the trick.

However, now the printer occasionally stops in the center of a line and waits—even though the computer is defaulted to 132 characters per line. Even with this glitch, though, the DMP-120 has worked satisfactorily ever since.

If you have this problem, Radio Shack will apparently fix it free of charge, regardless of warranty status. Tell them you have a timing problem and it will ring a bell. Be sure to try the POKE 151,255 command first.

Another annoyance is that the serial-port jack and DIP option (serial/parallel, baud rate, carriage return option and test/normal) selector switches are located behind the machine and are both completely obscured by the paper being fed. The DMP-200 has a similar design.

I was never able to get the printer to friction-feed paper that worked in my old Teletype and normal typewriter. The DMP-120 advertises friction feed

as an option, although the manual doesn't specify what weight paper will work. The platen appears to be much too slick to ever work in the friction-feed mode. I did not try heavier Radio Shack roll paper, however.

Other than these complaints, though, I found the printer to be well constructed and well designed. The self-test feature and buffered bidirectional printing are real plusses.

- 1
- 2
- 4
- 8
- 16
- 32
- 64

Fig. 1. Print-Head Dot Identification
To print dots 1, 4, and 64, use PRINT#-2,CHR$(1+4+64).

Software

One reason I bought a dot-matrix printer was to print out a graphics screen. I fired up my old Radio Shack Screen Dump program and it didn't work. It spit out a few dots and died.

Radio Shack now admits that Screen Dump won't work with the DMP-120, but they say a fix is on the way.

To solve this problem, I wrote a short Basic program to print out a PMODE4 screen (see the Program Listing). I've provided two scales.

Figure 2 gives a sample printout of a flight-simulator program on scale = 1. The y-direction is expanded about 10 percent relative to the screen and somewhat exaggerates the CoCo's stretching of this direction. You can change line 30 of the program by a PRINT#-2,CHR$(27)CHR$(19) and get it to over-correct slightly in the opposite direction. As is, scale = 1 gives a picture that is 2.5 by 2.5 inches.

Figure 3 gives a 6-by-7-inch sample of the scale = 2 option. Here, the y-direction and x-direction are very faithful reproductions of the screen. Both options are slow: It takes 10 minutes to print a full screen in scale = 1, and 20 minutes in scale = 2. Being so slow, it doesn't overheat the print head.

The manual gives conflicting guidance on this point. In one place it says to give the printer a 1-minute rest after 10 minutes of graphics printing. In another place, it says to give it a rest

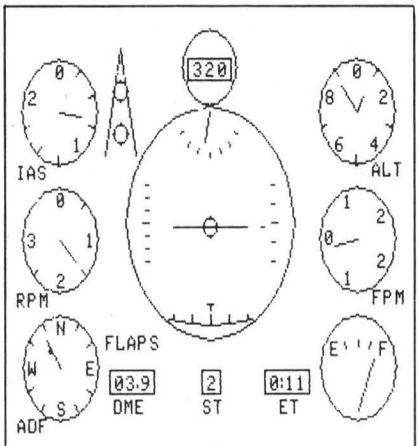


Fig. 2. Flight-Simulator at Scale = 1

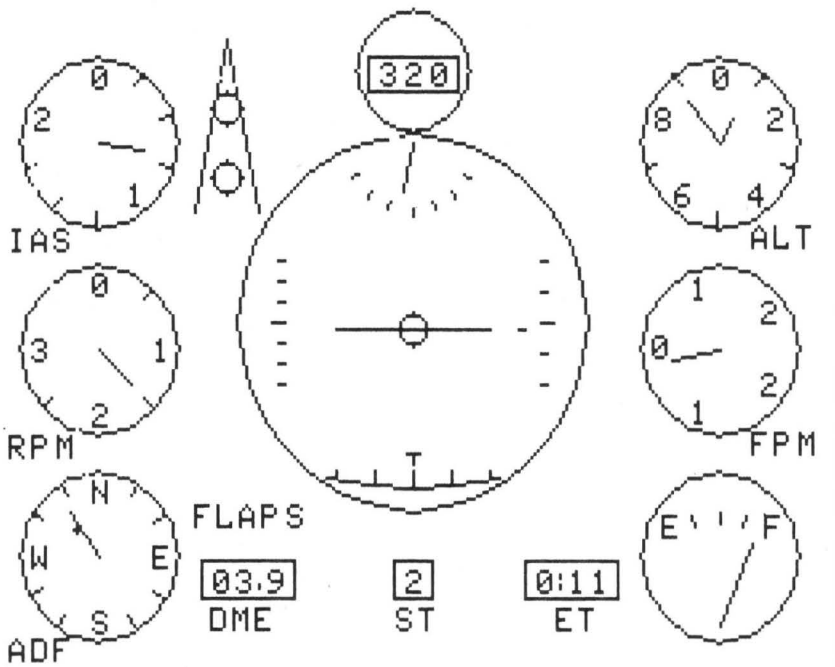


Fig. 3. Flight-Simulator at Scale = 2

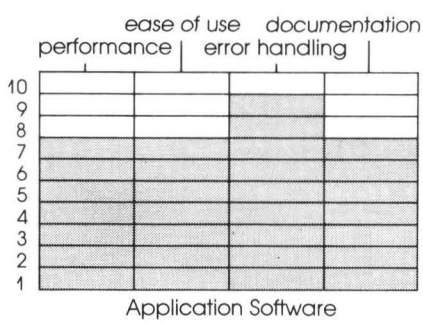
after printing one-sixth of a page.

The graphics program I've included takes 10 minutes to print out 25 percent of a page. A machine-language code could do it many times faster. Thus, a programmer will have to count rows and pause after each one-sixth of a page. This evidently means it will take about six minutes to dump a single screen. This seems a bit long.

Now that the printer is working reliably, I have only one complaint in that I can't use friction-fed roll paper. I like its low cost and the convenience of not having it sit on the table. All in all, though, I consider the DMP-120 worth the money. ■

```
10 POKE 151,255
20 PRINT#-2,CHR$(27)CHR$(20)
30 PRINT#-2,CHR$(18)
40 INPUT"FROM X1 TO X2 (0-255)";
X1,X2
50 INPUT"FROM Y1 TO Y2 (0-192)";
Y1,Y2
60 INPUT"SCALE 1OR2";A$
70 IF A$="2"GOTO 170
80 CLS:SCREEN 1,0:TIMER=0
90 REM LOOP TO GET 7 ROWS
100 FORJ=Y1 TOY2 STEP 7
110 FORI=X1 TOX2:N=PPOINT(I,J)+P
POINT(I,J+1)*2+PPOINT(I,J+2)*4+P
POINT(I,J+3)*8+PPOINT(I,J+4)*16+
PPOINT(I,J+5)*32+PPOINT(I,J+6)*6
4+128:PRINT#-2,CHR$(N);:N=0:NEXT
120 PRINT#-2
130 PRINT@25,J
140 NEXT J
150 PRINTTIMER/3600
160 END
170 FOR I=X1 TO X2 STEP 7
180 FOR J=Y2 TO Y1 STEP-1:N=PPOI
NT(I,J)*3+PPOINT(I+1,J)*12+PPOIN
T(I+2,J)*48+PPOINT(I+3,J)*64+128
:PRINT#-2,CHR$(N);CHR$(N);CHR$(N
);:N=0:NEXT:PRINT#-2
190 FOR J=Y2 TO Y1 STEP -1:N=PPO
INT(I+3,J)+PPOINT(I+4,J)*6+PPOIN
T(I+5,J)*24+PPOINT(I+6,J)*96+128
:PRINT#-2,CHR$(N);CHR$(N);CHR$(N
);:N=0:NEXT:PRINT#-2
200 PRINT I
210 NEXT I
220 PRINT TIMER/3600
230 END
```

Program Listing. Screen Print Program



Time & Money
The Computer House
Box 1051
DuBois, PA 15801
16K, Extended Color Basic
$19.95, cassette
$24.95, disk

by Robert P. Bussell

Time & Money provides a convenient means to perform the following common financial calculations:

- Compound Interest
- Sinking Funds
- Loans (ordinary annuity)
- Lease (annuity due)

You select each type from a main menu. Then you'll see a format that contains a line for each of the factors to be entered. The cursor indicates the first of these.

You can enter a value for this factor or use the up- and down-arrow keys to move to another factor. You can enter a factor as the number to be used or as a simple expression, like 4*12. The program recognizes all math operators.

"The program locks out all keys, including break, that you don't need to enter numbers or perform command functions."

After entering a value, press the enter key to put the value into the calculation, and press the up/down arrows to position the cursor at the next factor. When you have entered all known factors, position the cursor at the factor to be calculated and press the @ key to perform the calculation.

The program locks out all keys, including break, that you don't need to enter numbers or perform command functions. This makes it difficult to enter incorrect data.

Here is a sample calculation format:

Loans (ordinary annuity)

Initial Value	$	7,772.41
Interest (Apr)	%	10.50
Pmts. Per Year		12.00
No. of Payments		48.00
Payment Amount	$	199.00
Ending Balance	$	−0.01
Total Payments	$	9,552.00
Total Interest	$	1,779.58
Chnge in Princ	$	−7,772.42

In this example I want to know how much I can afford to pay for a new

CHARTER OFFER
SAVE $2.00 OFF THE BASIC RATE!

RUN

THESE COULD BE THE KEYS TO YOUR FUTURE

Unlock *all* the potential of your Commodore 64 and VIC-20* with **RUN**.

Explore . . . Experiment . . . Enjoy . . . Beginner and expert alike will be taken beyond the manual to the limits of their abilities. Enter your own game programs. Construct a simple hardware add-on. Broaden your scope with unique applications . . . And . . . get a 13th issue FREE!

Enjoy key features like these:

- Games for fun & strategy.
- Programming tips help you learn short cuts.
- Candid reviews help you make money-saving decisions.
- Programs to add to your library.
- Instructions & tutorials to increase your skills.
- Hardware & software modifications help your machine work smart.
- Unique applications broaden your scope.

Here's a system-specific magazine written with you in mind. Written by and for the reader to give time-saving, money-saving hints. You'll get instructions and tutorials to increase your skills, and candid reviews to help you make the right decisions. Most of all though, you'll have fun.

Commodore 64 and VIC-20 owners are one of the largest groups of computerists today. Enjoy the benefits of this with your own magazine. Be in control like never before. Order **RUN** today and get a 13th issue free with your prepaid order (check or credit card) of only $17.97. Send in the coupon or call toll free *1-800-258-5473.* In N.H. call 1-924-9471.

*Send me a subscription to **RUN** for only $17.97 per year. I understand that with payment enclosed or credit card order I will receive a FREE issue making a total of 13 issues for $17.97. Save $2.00 off the basic rate!*

☐ *CHECK/MO* ☐ *MC* ☐ *AE* ☐ *VISA* ☐ *BILL ME*

card # _____ exp. date _____

signature _____

name _____

address _____

city _____ state _____ zip _____

Canada & Mexico $20.97; Foreign Surface $37.97, 1 year only. US funds drawn on US bank. Foreign airmail, please inquire. Please allow 6 to 8 weeks for delivery.

RUN • Box 954 • Farmingdale, NY 11737

343F4

*Commodore 64 and VIC-20 are registered trademarks of Commodore Business Machines, Inc.

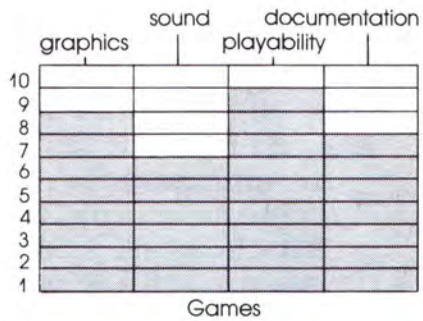
car, given the interest rate of 10.5 percent for a period of four years and my monthly payment of $199.

If, after running this program, I want to know what the effect on my payment is if I spend $10,000, I move the cursor to the Initial Value and change it to $10,000. Then I place the cursor at the Payment Amount and press the @ key. The screen displays my new monthly payment.

A handy feature of this program is its ability to save the results of one calculation as an entry value for another type of calculation selected from the menu.

I use this program frequently. I wish, however, they had included a feature to let you save the tape version to disk.

The 16-page documentation pamphlet explains how to use the program and gives a typical, real-world situation as an example of how to use each calculation. Time & Money is well written and executes without a flaw. ■



Pooyan
DataSoft
9421 Winnetka Ave.
Chatsworth, CA 91311
32K, joysticks optional
$29.95, cassette and disk together

by Peter Paplaskas
***HOT CoCo* staff**

If you thought the Three Little Pigs did away with the Big Bad Wolf, think again—now he's back with all his brothers in Pooyan, a considerably altered version of the old story.

In this machine-language game, you use a joystick or the up/down-arrow keys to raise or lower a piglet (pooyan means piglet in Chinese, so we're told) and fend off an aerial assault by wolves.

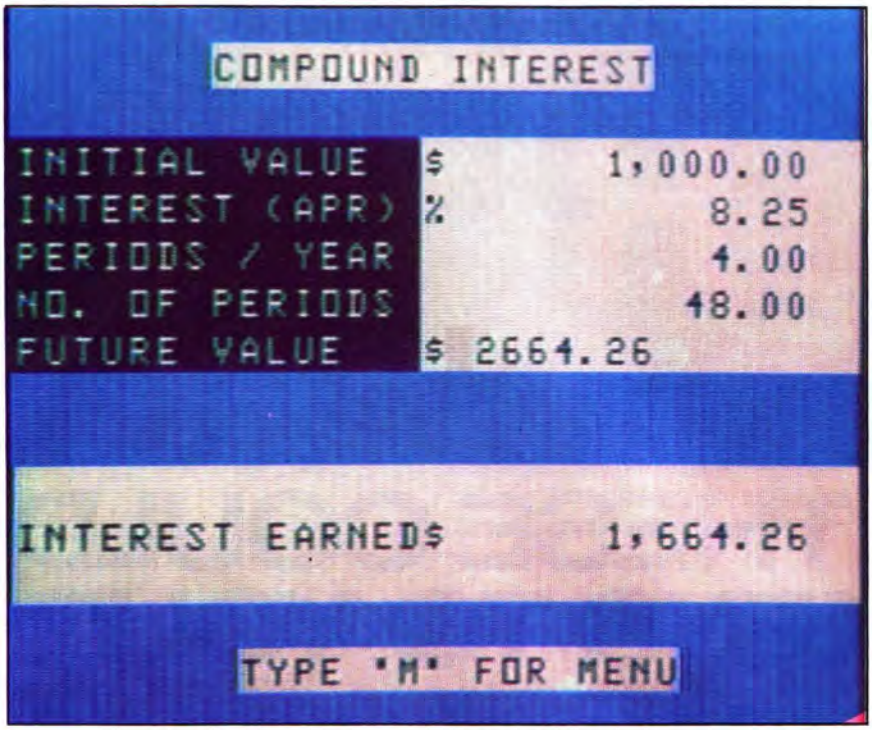


Photo 2. Screen Display from Time and Money

In the first screen, 32 wolves come five at a time to the top of the screen and float to the ground via balloon. The piglet moves up and down in a basket, firing arrows at the balloons and dodging the acorns that the wolves throw.

Once in a while the pig gets a secret weapon in the form of a piece of meat that appears at the top of the screen. If he can get up and grab it and fire it into a cluster of wolves, they will let go of their balloons to catch it. Ah, the wages of greed.

Any wolves that make it safely to the ground run immediately across to a ladder and climb up to one of four platforms from which they can push the piglet out of his basket.

In the second screen, 40 wolves come five at a time to the bottom of the screen, grab two or three balloons, and attempt to float to the top. It takes several hits here to stop each wolf, and if seven of them reach the top of the screen, they line up to push a rock down on the hapless pig.

This is an excellent, fast-action game. The packaging is very impressive and contains a tape and a disk version of the game. DataSoft does this to make it easy when cassette users upgrade their systems to disk. I think that's a worthy idea. ■

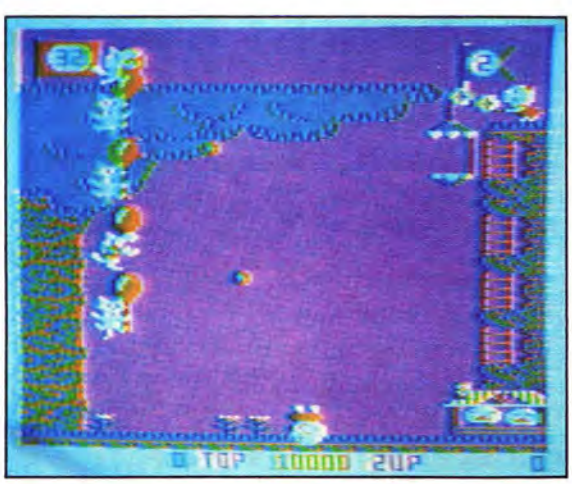


Photo 3. Pooyan

✓See List of Advertisers on page 130

BY PHILIP MCLAUGHLIN

WHAT'S DISK?

Know your sectors from your granules, and even learn to emulate FLEX utilities on Color Basic.

One of the best things about the magnetic disk system as opposed to a cassette tape system is that it lets you store and access larger amounts of programs or data in a shorter period of time. Because it is so easy to save programs, you can be confident when tinkering with and improving them. If the new version malfunctions, the old version is still on the disk.

If you know how a disk stores information, you can use your system more efficiently. Here's how to get the most out of your disk-based CoCo.

DSKI$ and DSKO$

Those of you who are used to other operating systems might get frustrated with the omission of file dating, until you realize it is compensated by two special commands. DSKI$ and DSKO$ (pronounced Disky and Disco) are unusual input/output commands that function similarly to the high-priced zapper utilities.

DISKO$, the output function of the pair, is so powerful it is actually dangerous to the information on the disk because it disregards the directory system that normally keeps files from being overwritten. It should be used only by those who possess both a clear understanding of what they want to do with it and a good back-up disk. DSKO$ is the only Basic command word that can clobber the directory.

The other command, DSKI$, is perfectly safe because it only inputs. But it is no less powerful. To DSKI$ there is no such thing as a protected disk. It allows you to position the drive head over, and read the information off, any part of the surface. With these commands the Color Computer disk system can be made to do just about anything any other system does and some things those others can't. But using these commands requires a little knowledge of how information is physically arranged on the disk. Tracks, granules, and sectors are, in descending order of size, the divisions of the surface recognized by the system.

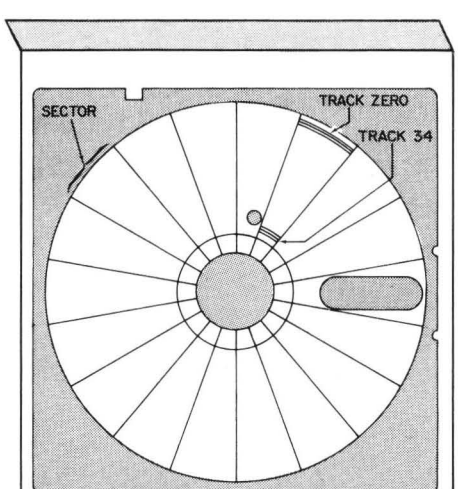


Fig. 1. Radio Shack CoCo disks feature 18 sectors with 35 tracks each. The directory is stored on track 17.

Tracks and Sectors

The track is an obvious physical division similar to the grooves of a phonograph record. In this case there are 35 concentric rings of information. Each track contains 4,608 bytes or roughly three typewritten pages of storage. Because the inner tracks are shorter, the information on them is squeezed together, but the programmer doesn't need to know this. Track changing is the only operation that causes the magnetic head to move. The other divisions are accessed by the spinning of the disk.

A sector is a pie-slice shaped section. Here there are 18 such sections. The portion of track that falls in each sector is 256 bytes long. This is the standard chunk of data written or read by the head at once and is the portion addressed by DSKI$ and DSKO$.

For example, DSKI$ 0,7,11,A$,B$ would read from drive 0 the portion of track 7 which is in sector 11. It is a peculiarity of these commands that they divide the 256 bytes of information they

System Requirements

16K RAM
Disk Color Basic
Printer (optional)

handle into two string variables, which are both 128 bytes long. In this example, A$ would contain the first half of the sector and B$ the last.

The Directory

Now you have the tools to get information back and forth, but to make use of them you need to know how the directory works.

The directory is stored on track 17. This location makes sense because it is right in the middle so the head can swing to the first or last track in equal time. The actual directory entries take up sectors 3 to 11 in this track. There are eight on each of these nine sectors, making room for 72 entries. Each entry is 32 bytes long. The file name takes up the first 8 bytes of each entry. The routine below steps through each of the nine sectors and each of the eight entries in each sector and puts the file names into the variable NA$ in line 150.

```
100 FOR SE=3 TO 11
110 DSKI$ 0,17,SE,A$,B$
120 A$=A$+LEFT$(B$,114)
200 FOR DE=0 TO 7
210 PT=DE*32
250 NA$=MID$(A$,PT+1,8)
260 PRINT NA$
.
.
.
800 NEXT DE
900 NEXT SE
```

The outer portion of this double loop puts the 256 bytes of each sector into A$ and B$, which are then joined together as A$ in line 120. Because a Basic string variable can't hold all 256 bytes, the very last of B$ is truncated. The inner loop sequence is a pointer PT (line 210) that is used in a MID$ function to get the 8-character filename in line 250. (MID$ takes eight characters of A$ starting at the first byte after the pointer.)

The routine is a general-purpose skeleton. Changes can be made around line 250 that allow it to do different things. For instance, after the 8 bytes of the file name are 3 bytes for the extension. This might be obtained by adding EX=MID$(A$,PT+9,3) to line 250. It would be easy to add conditional statements to make the routine print only files with a certain extension such as: IF EX$="NEW" THEN PRINT NA$, EX$, or to kill all files of a certain extension, KK$=NA$+"/"+EX$:IF EX$="OLD" THEN KILL KK$, or do other manipulations.

Users of the highly rated FLEX system might recognize a similarity to the CAT,--- (catalog by extension) or X---(delete conditional on extension) utilities in the suggested modifications. You can approximate most of the other FLEX commands. One of the very good features of Disk Extended Color Basic is that the operating system is contained in Basic instead of the other way around. This allows sophisticated file handling by Basic programs without ever having to go into a system mode or load a DOS.

"One of the very good features of Disk Extended Color Basic is that the operating system is contained in Basic instead of the other way around."

The next 3 bytes in the directory entries (11, 12, and 13) are the file type, the ASCII flag, and a pointer to the beginning of the file in the disk storage area. Here is the entry format:

Byte	Function
0–7	File Name
8–10	Extension
11	File Type
12	ASCII Format Flag
13	Points to Start Granule
14–15	Ending Data
16–31	Not Used

The last half of each entry (bytes 16–31) is blank. The owner's manual states that this space is reserved for future use. Some people use this area for expanded directory area, putting date-time or other information in with DSKO$.

```
9 '***initialization
10 CLEAR1100:CLS6:PRINT@75,"'DIR
ECT'";
15 PRINT@164,"EXPANDED DISK DIRE
CTORY";:PRINT@288," SHOWS PART O
F THE FIRST SECTOR     OF ALL FI
LES ON THE DISK."
20 PRINT@384,"    INPUT 1 FOR HA
RD COPY.":PRINT@494,"";:INPUTZZ:
IFZZ=1THENPR=-2 ELSEPR=0
30 IFPR=-2THENLL=26ELSELL=18'def
ault expansion lengths
32 CLS6:PRINT@166,"DIRECTORY EXP
ANSION";:PRINT@204,"WILL BE";:PR
INT@237,LL;:PRINT@266,"SPACES LO
NG";
34 PRINT@416,"    INPUT 1 TO CHA
NGE THIS.":PRINT@494,"";:INPUTZZ
:ON ZZ GOSUB 1000
40 IFPR=0THEN70
50 CLS6:PRINT"PRINTER READY?":PR
INT#PR,"":PRINT#PR,"":PRINT#PR,"
":INPUT"DISK NAME OR IDENTIFICAT
ION";ID$
60 PRINT#PR,ID$:PRINT#PR,"":PRIN
T#PR,STRING$(LL+8,"*")
70 '***phase 1 get name of file
(type,start gran)from dir track
80 TR=17'set track to 17 (direct
ory track)
90 '***sector loop (outer)
100 FORSE=3TO11
110 DSKI$ 0,TR,SE,A$,B$
120 A$=A$+LEFT$(B$,112)'concaten
ate secter strings
190 '***directory loop (inner)
200 FORDE=0TO7
210 PT=DE*32
240 '***get filename
250 NF$=MID$(A$,PT+1,8):EX$=MID$
(A$,PT+9,3)
260 '***get file type byte11
270 TY=ASC(MID$(A$,PT+12,1))
280 '***get asc flag byte 12
290 AF=ASC(MID$(A$,PT+13,1))
300 '***get gran# byte 13
310 GR=ASC(MID$(A$,PT+14,1))
320 '***flag dead files
330 IFLEFT$(NF$,1)=CHR$(0)THENMI
D$(NF$,1,1)=" "
340 '***end loop if file name is
 end character
350 IFLEFT$(NF$,1)=CHR$(255)THEN
905
360 '***print out file name
370 PRINT#PR,NF$"/"EX$" ";
380 IFTY=2THEN600'dont print mac
hine code
390 '***phase 2 get line 0 from
file tracks
400 '***convert gran# to track a
nd sector
410 FT=GR/2:IFFT-INT(FT)=0THENFS
=1ELSEFS=10
420 FT=INT(FT):IFGR>33THENFT=FT+
1
430 '***read start of file
440 DSKI$ 0,FT,FS,C$,D$
450 C$=C$+LEFT$(D$,114)'concaten
ate
460 CC=0 'reset character counte
r
470 IFTY=0THENFF=0ELSEFF=1'reset
 basic flip-flag if needed
480 '***filter crazy characters
490 FORZZ=1TO240:Z$=MID$(C$,ZZ,1
)'peel off characters
500 IFASC(Z$)>128 ORZZ>8THEN FF=
1'line number flip-flag
510 IFZZ<5 ANDTY=0 THENFF=0
520 IFAF=255THENFF=1
530 IFASC(Z$)<32 OR ASC(Z$)>122
THEN590'divert if its not alphan
umeric
540 IFFF=0THEN590'divert if basi
c token
550 CC=CC+1'count characters
560 IFCC>LL THEN600'if too many
on line
570 '***print first of file
580 PRINT#PR,Z$;
590 NEXTZZ
600 PRINT#PR,""'carriage return
790 '***next entry
800 NEXTDE
890 '***next sector
900 NEXTSE
905 PRINT#PR,FREE(0)" GRANULES L
EFT ON DISK
910 '***another disk?
920 PRINT"INPUT 1 TO QUIT-":PRIN
T"OTHERWISE PUT IN ANOTHER DISK"
;:INPUTZZ
930 IFZZ=1THENENDELSE40
1000 CLS6:INPUT"LENGTH  OF EXPAN
SION";LL:GOTO 32
```

Program Listing. Direct

However, there are some advantages to having this information more directly attached to the file so it will never get lost if the file is renamed or copied. Then the routine can be set up and modified without having to worry about accidental zaps.

One solution that works well with Basic programs is to use the first line (line 0) of the program itself for identification. With DSKI$, you can get this directly off the disk without having to load or list the program. The place to look for the start of the saved program is specified by byte 13 of each directory entry. Byte 13 will be a number from 0 to 67. To use it, it is necessary to understand one more unit of disk space allocation.

Granules

The granule is a more artificial division than the track or the sector, and it lies between them in physical size. It is half of a track, which is equal to nine sectors, or 2,304 bytes. This is the minimum unit of storage normally allocated by the filing system. Adding just 1 byte to a program or file, which is exactly one granule long, causes the space reserved to double. Hence, a disk containing many short programs might have a lot of dead space.

The number in byte 13 of each directory entry is the identification of one of 68 possible granules (possible because 34 tracks are used for data and each track has two granules). But DSKO$ and DSKI$ need to know a track number and sector number, not a granule number.

Converting

All granules start on either sector 1 or sector 10. Track 0, sector 1, is the start of granule 0. Sector 10 of this track starts granule 1, and the first sector of the next track (track 1) is the beginning of granule 2. The conversion formula is that the track number equals half the granule number. The sector is 1 if the granule number is even, or 10 if the granule number is odd. If the granule is more than 33, the formula in the last two sentences applies, except that the track number is greater by one.

Though this sounds convoluted, it is simple, except that the directory in track 17 is not counted as granules. This throws the sequence off for the higher tracks. The conversion is done in lines 410 and 420 of the Program Listing, Direct.

About Direct

This utility looks directly at the first part of programs on the disk in two phases. Phase 1 is the general-purpose, directory reader loop shown and explained above. Phase 2 calculates the beginning track and sector from the granule number in byte 13 of each entry.

DSKI$ is used here to get the first sector of the program or file into the variables C$ and D$ (line 440). D$ is strung together with C$, and then a portion of it is printed out, either to the screen or the printer (line 580). Then it loops to the next entry. The program is no more complicated than that. The additional lines are mainly garbage filters because some files contain data, such as codes for carriage returns and backspaces, that might upset a printer.

Since screen space is limited, another trap prevents printing the line number of the first Basic line. Direct also reads the first sector of other types of files. Machine-language files usually look like gibberish, so a flag in line 380 blanks them out.

On the other hand, it is often informative to see ASCII and data files, so they are displayed even though they look strange. You can sometimes reconstruct long lost or forgotten data formats this way. Because of the peculiar way direct-access files are stored, a portion near the last of the file is seen.

Seeing Ghosts

Direct displays the ghosts of dead files. Killing a file merely places a zero in the first byte of the file name in the directory entry, allowing it to be written over later. If you accidentally kill a crucial file, it is possible to resurrect it by inserting a valid character in place of the zero, provided you haven't written over the old storage area. Very careful use of DSKO$ accomplishes this. Direct shows dead files just like live files except it displays a blank space for the first letter of the name.

The program prints an expanded directory containing the beginning of all the Basic programs on the disk to the screen, or it makes a printout. The default expansion length is set so each entry only takes up one line on the screen or on a 40-column printer. This provides a minimum of 18 more characters in addition to the eight-character file name and the three-character extension.

You have the option to change the length so you can view up to 256 characters of the files, or more with some changes to the program. The default values for expansion lengths are in line 30 and you can change them to fit different printers. With no DSKO$, it is perfectly safe to tinker with the routine.

If the first 18 characters of the first line of a Basic program are now part of the directory, it is helpful to adopt some kind of standard order for them. Otherwise, the directory will look confusing. I include in my first program line (line 0) an apostrophe or REM for remark, possibly followed by the date and the particular distinctive feature of that version of the program.

It seems best to put the name of the Basic program after all the other information because it is generally the same or similar to the file name. Fig. 2 is a sample expanded directory.

At the end of the routine is a prompt that asks you to exit or insert another disk. This way it takes only a few minutes to get a printout of the contents of a small library of disks. Once generated, you can tack the hardcopy to the wall or put it where it is visible at a glance. It sure beats having to go through a stack of disks loading and listing the programs to see what's in them. ■

```
DISK# 15        MAY 21
DIRECT          /BAS MAY2 FOR HOT COCO MAGAZINE
DIRECT          /OLD DEC 82 OLD VERSION OF UTIL
WORDPLOW        /BAS ARTICLE WRITER APR19
FLOWCHART       /UTL DRAWS FLOW CHART SUN MAY1
BALMER          /BAS BALMER SPECTRUM MAY 19 4PM
370DOWN         /IBM DOWNLOAD SYS370 CORE DUMP
HEMOCRIT        /BAS BLOOD CELL COUNT UTILITY
FASDRAW         /BAS RAPID 3D DRAWING AID MAY12
WHO?            /BAS VOICE RECOGNITION EXPERIME
SPIDER          /UTIL SUPER FILE MAINT. UTIL. OC.
CBASFORT        /IBM EXT. BASIC TO FORTRAN TRAN
```

Fig. 2. Sample Directory

Address correspondence to Philip McLaughlin, 510 Fort Worth Drive, Denton, TX 76201.

'REAL TALKER'

HARDWARE Voice Synthesizer

NEW from COLORWARE.. only...$59.95

THINKING OF BUYING A COCO VOICE SYNTHESIZER?

READ THIS....

Making your computer talk couldn't be any easier! 'Real Talker' is a full featured, ready to use, HARDWARE voice synthesizer system in a cartridge pak. It uses the Votrax SC-01 phoneme synthesizer chip to produce a clear, crisp voice.

FREE TEXT-TO-SPEECH

Included free with 'Real Talker' is Colorware's remarkable Text-to-Speech program. This is a truly powerful machine language utility. What it does is automatically convert plain English to speech. And it has an unlimited vocabulary! For example, use it in the direct mode: Type in a sentence or a paragraph, even mix in numbers, dollar signs, etc., then press enter. The text is spoken. At the same time a phoneme string is generated which can be saved to cassette or disk, modified or used in a Basic program.

We originally planned to sell this major piece of programming for about $40.00 but decided it was so useful that no 'Real Talker' user should be without it. Besides, it really shows off the capability of 'Real Talker'.

Also included with 'Real Talker' is our unique Phoneme Editor program. It allows you to explore and create artificial speech at the phoneme level. Phenomes are the fundimental sounds or building blocks of word pronunciation. There are 64 different phenomes, as well as 4 inflection levels at your disposal. Creating and modifying speech at the phenome level is both fascinating and educational. The Phenome Editor may also be used to customize the pronunciation of speech produced by the Text-to-Speech program.

You don't have to use any of our utility programs though. If you write your own Basic Programs, you will find the pocket sized Votrax Dictionary (included free) is all you need to make your own Basic programs talk. This dictionary gives you quick access to the phenome sequences used to create approximately 1400 of the most used words in the English language.

How about compatibility? 'Real Talker' is compatible with any 16K, 32K, 64K, Extended or non-extended Color Computer. It works with any cassette or disk based system, with or without the Radio Shack Multi-slot expander. No other synthesizer under $100 can make this claim. Most other CoCo voice synthesizers require an expensive Multi-slot expander in order to work with the disk system. 'Real Talker' requires only an inexpensive Y-adapter. This is an important consideration if you plan on adding a disk or have one already.

'Real Talker' comes completely assembled, tested and ready to use. It is powered by the CoCo and talks through your T.V. speaker so there is nothing else to add. Price includes Text-to-Speech and other programs on cassette (may be transferred to disk), User Manual and Votrax Dictionary. ONLY $59.95

'Y-BRANCHING CABLE' For disk systems. This 40-pin, 3 connector cable allows 'Real Talker' to be used with any disk system $29.95

YOU DECIDE....

Order yours today on our Toll-Free Order Line. If you are not delighted with your 'Real Talker' system, simply return it within 30 days for a prompt, courteous refund.

COLORWARE

COLORWARE INC.
78-03H Jamaica Ave.
Woodhaven, NY 11421
(212) 647-2864

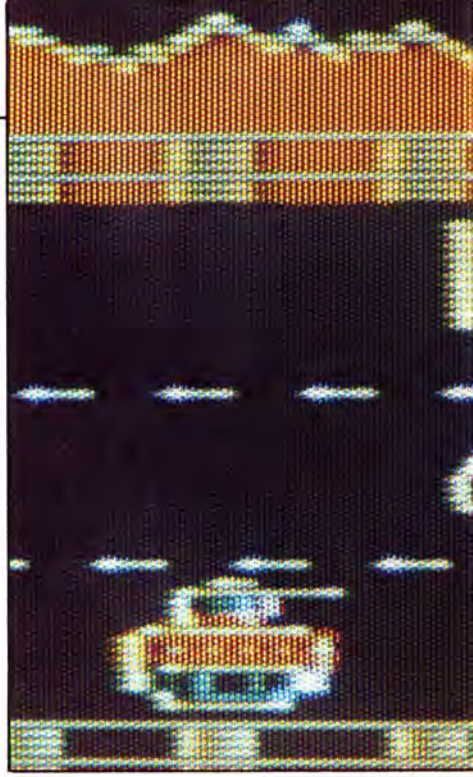
VISA MasterCard

★ ★ ★ ORDERING INFORMATION ★ ★ ★

ADD $2.00 PER ORDER FOR SHIPPING & HANDLING.
C.O.D.'S: ADD $3.00 EXTRA.
SHIPPING & HANDLING FOR CANADA IS $4.00
WE ACCEPT VISA, MASTER CARD, M.O.'S, CHECKS.
N.Y. RESIDENTS MUST ADD SALES TAX.

BY NICK BRADBURY

POSSUM RUN



instant CoCo

You are a desperate possum trapped on a busy, three-lane highway, patrolled by cars, milk trucks, and tanks. Can you survive long enough to get home?

Possum Run is a high-resolution arcade-style game for the CoCo, loosely based on the game Frogger. It is fast because it uses machine-language routines. The hi-res graphics are detailed and a good example of artifacting colors in PMODE 4.

Make sure that you save several copies before running the program because a mistake in typing in the machine-language routines can result in a crash. If your computer does not work in the high-speed mode, then change line 4 to read: 4 PK = 65494. This fixes the speed for the entire program.

The first thing you see when you run Possum Run is a red or orange-red screen. If the screen is a bluish color, press reset and run the program until the screen is red. After a short pause, you are asked to pick your difficulty level.

Level 1 is next to impossible, while level 20 is a snap. A good starting level is 15. You begin game play at the chosen level. Use the right joystick to move your possum along the highway lanes and up and down each lane. The vehicles periodically move along the highway.

Poor old possum headin' home—the car misses, the milk truck too! Watch out for that tank! Splat!

If you collide, your possum becomes a graphic splat and you lose him. Because of the machine-language routines, the screen is not limited to only one or two vehicles moving at the same time. Play at level 1 and you will see what I mean. There are usually six or seven vehicles per lane to make the game more challenging. At the top of the screen is a timer. If you survive long enough for the timer to run out, you are awarded a 1,000-point bonus and you continue at a level two notches higher.

System Requirements

16K RAM
Extended Color Basic
Joystick

Program Listing. Possum Run

```
0 CLS0:CLEAR200,16000:PCLEAR4:DI
MA1(16),A2(16),A3(14),A4(14),BU(
8),G1(8)
4 PK=65495
8 POKEPK,0
10 PMODE3,1:PCLS3:PMODE4,1:SCREE
N1,1
15 GOSUB5000
20 GOSUB2000
25 CLS7:PRINT@203,"POSSUM RUN";:
PRINT@264,"BY NICK BRADBURY";:SC
REEN0,0:POKE65494,0:PLAYP2$:POKE
PK,0
30 GOSUB1000
40 RN=40:SC=0:NB=4
80 GOSUB8000
82 CLS:POKE65494,0:SCREEN0,0:PRI
NT"ENTER DIFFICULTY LEVEL:":INPU
T"(1=DIFFICULT,20=EASY)";CS:IFCS
<1ORCS>20THEN82ELSECS=CS+3
88 POKEPK,0
90 X1=2:Y1=101:X2=2:Y2=101:X3=21
0:Y3=82:X4=210:Y4=118
92 ZX=100:ZY=103:GOSUB500
94 GOSUB8200
96 PLAY"V30L255T255"
98 PMODE4,1:SCREEN1,1
100 J0=JOYSTK(0):J1=JOYSTK(1)
104 IFJ0<16THEN300ELSEIFJ0>47THE
N325
108 IFJ1<16THEN350ELSEIFJ1>47THE
N375
110 U=USR0(0):U=USR1(0):U=USR2(0
):GOSUB505:U=USR3(0):U=USR4(0):U
=USR5(0):P1=PPOINT(ZX+2,ZY+2):P2
=PPOINT(ZX+20,ZY+6):GOSUB500:IFP
1<>0THEN600ELSEIFP2<>0THEN600
120 SC=SC+1:IFSC>CS THENSC=0ELSE
130
122 IFRND(2)=1THENGOSUB400
124 IFRND(2)=1THENGOSUB420
126 IFRND(2)=1THENGOSUB425
130 TI=TI+1:IFTI>8THENTI=0:GOSUB
8100:S1=S1+5:PLAYP1$
199 GOTO100
200 POKE65494,0:PLAY"V30L255T255
":FORI=1TO5:PLAY"O"+STR$(I):PLAY
"EADCF":NEXTI:TP=0
205 PMODE4,1:COLOR1,0:LINE(0,101
)-(255,115),PRESET,BF:LINE(0,82)
-(255,98),PRESET,BF:LINE(0,118)-
(255,133),PRESET,BF:U=USR2(0):GO
SUB500
210 '
215 PLAY"O3V30L5T5EGAV5AV30GEDV5
DV30BCAV5AV30GECV5CV30AV5AV30O4B
AGFEV5EV30DV5DV30CV5CV30"
220 TI=0:S1=S1+1000:CLS:SCREEN0,
0:PRINT"YOU HAVE SCORED A 1000 P
T BONUS!":PRINT:PRINT"YOU NOW HA
VE A SCORE OF"S1:PRINT"  CONGRA
TULATIONS!"
```

Photo —Frank Cordelle

How It Works

I made the detailed graphics in Possum Run by first drawing the outline on graph paper. I transferred this into a DRAW statement. The windows, racing stripes, and other such frills were done by trial and error PSET, LINE, and PAINT commands. I saved all finished figures in a GET array.

Each machine-language routine scrolls a separate part of the screen. The mountain-scrolling routine moves the mountains to the left and wraps them around to the right. Two separate routines scroll the medians, and another moves the guard rail. This gives the effect of the possum moving toward its destination. The three other routines move the areas containing the cars, tanks, and milk trucks. Using this method, the overall speed remains the same if there are two or 200 cars on the screen.

32K users might want to relocate the machine-language routines (lines 5000–5999) into upper RAM. This frees another 16K that can be used to add things such as on-screen scoring that doesn't fit into 16K. Possum Run runs unchanged on 32K systems, but, to my knowledge, it does not run on disk systems without relocating the machine-language routines. ■

Address correspondence to Nick Bradbury, 10500 Sandpiper Lane, Knoxville, TN 37922.

```
230 GOSUB8000
235 CS=CS-2:IFCS<4THENCS=4
240 PRINT:PRINT"YOU WILL NOW ADV
ANCE TO A HIGHERDIFFICULTY LEVEL
 OF"CS-3"."
245 PRINT" GOOD-LUCK!!!"
250 FORI=1TO2400:NEXTI
260 POKEPK,0
290 GOTO610
300 IFZX<50THEN108ELSEP1=PPOINT(
ZX-9,ZY+4):IFP1<>0THENGOSUB505:Z
X=ZX-8:GOTO600ELSEGOSUB505:ZX=ZX
-8:GOSUB500
320 PLAYP4$:GOTO108
325 IFZX>190THEN108ELSEP1=PPOINT
(ZX+29,ZY+4):IFP1<>0THENGOSUB505
:ZX=ZX+8:GOTO600ELSEGOSUB505:ZX=
ZX+8:GOSUB500
345 PLAYP4$:GOTO108
350 IFZY<90THEN110ELSEP1=PPOINT(
ZX+14,ZY-14):IFP1<>0THENGOSUB505
:ZY=ZY-18:GOTO600ELSEGOSUB505:ZY
=ZY-18:GOSUB500
370 PLAYP4$:GOTO110
375 IFZY>110THEN110ELSEP1=PPOINT
(ZX+13,ZY+24):IFP1<>0THENGOSUB50
5:ZY=ZY+18:GOTO600ELSEGOSUB505:Z
Y=ZY+18:GOSUB500
390 PLAYP4$:GOTO110
400 IFRND(2)=1THEN405ELSEPUT(X1,
Y1)-(X1+42,Y1+13),A1,PSET
402 RETURN
405 PUT(X2,Y2)-(X2+42,Y2+13),A2,
PSET
408 RETURN
410 LINE(X1,Y1)-(X1+42,Y1+13),PR
ESET,BF
412 RETURN
415 LINE(X2,Y2)-(X2+42,Y2+13),PR
ESET,BF
418 RETURN
420 PUT(X3,Y3)-(X3+32,Y3+13),A3,
OR
422 RETURN
425 PUT(X4,Y4)-(X4+30,Y4+16),A4,
OR
427 RETURN
430 PUT(X5,Y5)-(X5+20,Y5+10),G1,
PSET
432 RETURN
500 PUT(ZX,ZY)-(ZX+24,ZY+10),BU,
OR
502 RETURN
505 PMODE4,1:COLOR1,0:LINE(ZX,ZY
)-(ZX+24,ZY+10),PRESET,BF
508 RETURN
600 POKE65494,0:PLAY"O3V30L255T2
55":X5=ZX:Y5=ZY:GOSUB430:FORI=12
TO1STEP-1:PLAYSTR$(I):PLAYSTR$(I
):NEXTI
605 NB=NB-1:IFNB<0THEN9000
610 PMODE4,1:COLOR1,0:LINE(0,101
)-(255,115),PRESET,BF:LINE(0,82)
-(255,98),PRESET,BF:LINE(0,118)-
(255,133),PRESET,BF:POKEPK,0
690 GOTO90
990 POKE65494,0:POKE359,126
995 SOUND200,2
999 GOTO999
1000 PMODE4,1:PCLS0:SCREEN0,1
1010 DW$="R2E1R3D1R4F1D1L2U1L1G1
L3H1L1G1L1U2"
1020 PMODE3,1:DRAW"C4S12BM128,96
;XDW$;"
1022 PAINT(146,99),3,4:COLOR2,1:
LINE(126,98)-(158,98),PSET
1024 DRAW"C1S12BM128,96;XDW$;"
1026 PMODE4,1:COLOR1,0:DRAW"C1S1
2BM128,96;XDW$;"
1028 LINE(140,94)-(144,96),PRESE
T,BF
1030 CIRCLE(135,101),3,1
1032 CIRCLE(152,101),3,1
1034 PMODE3,1:COLOR3,1:LINE(156,
99)-(161,101),PSET,BF:LINE(153,9
7)-(158,99),PSET,BF
1036 COLOR2,1:LINE(126,98)-(158,
98),PSET
1038 PMODE4,1:DRAW"S12C1BM128,96
;XDW$;"
1040 GET(122,92)-(164,105),A1,G
1045 PMODE4,1:PCLS0
1050 PMODE3,1:DRAW"C4S12BM128,96
;XDW$;":PAINT(146,99),2,4:COLOR3
,1:LINE(128,98)-(158,98),PSET
1052 DRAW"C1S12BM128,96;XDW$;"
1054 PMODE4,1:COLOR1,0:DRAW"C1S1
2BM128,96;XDW$;"
1056 LINE(140,94)-(144,96),PRESE
T,BF
1058 CIRCLE(135,101),3,1:CIRCLE(
152,101),3,1
```

Listing continued

Listing continued

```
1060 GET(122,92)-(164,105),A2,G
1065 PMODE4,1:PCLS0
1070 DW$="E1R2F1ND1R1D1L1L1L3U1"
1072 DRAW"C1S8BM128,96;XDW$;"
1074 PAINT(132,96),1,1
1076 PMODE3,1:COLOR2,1:LINE(129,
99)-(129,101),PSET:LINE(129,101)
-(131,101),PSET
1078 LINE(134,99)-(134,101),PSET
:LINE(134,101)-(136,101),PSET
1080 COLOR3,1:LINE(138,96)-(140,
98),PSET,B
1082 PSET(142,98,3)
1084 LINE(126,96)-(124,94),PSET
1090 GET(120,92)-(144,102),BU,G
1095 PMODE4,1:PCLS0
1100 DW$="E1R1E1R2ND3U2R8D5L1U1L
3D1L1U1L3D1L2U1L3D1"
1110 PMODE3,1:DRAW"C4S8BM128,96;
XDW$;"
1112 PAINT(142,92),4,4
1114 PAINT(134,94),3,4
1116 PMODE4,1:COLOR1,0:CIRCLE(14
1,98),2,1
1118 CIRCLE(149,98),2,1
1120 CIRCLE(131,98),2,1
1122 DW$="R1D1F1D2L3U2E1U1"
1124 DRAW"C2S8BM146,89;XDW$;"
1130 GET(124,88)-(156,101),A3,G
1135 PMODE4,1:PCLS0
1140 DW$="U2E1R3H1U1NL4R5D2R2F1D
2L1H1L7G1NL1F1R7E1"
1150 PMODE3,1:COLOR4,1:DRAW"C4S8
BM128,96;XDW$;"
1152 PAINT(140,92),2,4
1154 LINE(136,85)-(140,85),PSET
1156 LINE(136,90)-(144,90),PSET
1158 FORX=130TO148STEP4:PRESET(X
,92):NEXTX
1160 PAINT(138,96),3,4
1162 FORX=134TO148STEP5:LINE(X,9
5)-(X+1,96),PRESET,B:NEXTX
1164 COLOR3,1:LINE(139,87)-(142,
88),PSET,B
1170 GET(122,84)-(152,100),A4,G
1175 PMODE4,1:PCLS0
1180 DW$="R1F1E1R1F1E1R1D1G1F1D1
L1H1G1L1H1G1L1U1E1H1U1"
1182 DRAW"C1S8BM128,96;XDW$;"
1184 PAINT(132,100),1,1
1186 PMODE3,1:CIRCLE(134,100),4,
3
1188 PAINT(135,100),3,3
1190 PMODE4,1:COLOR1,0:GET(124,9
5)-(144,105),G1,G
1500 PMODE4,1:PCLS0:PMODE3,1
1505 PMODE1,3:PCLS2:PMODE4,1
1550 PMODE3,1:COLOR1,0:LINE(0,96
)-(255,140),PSET,BF
1600 FORX=0TO255STEP23:LINE(X,13
5)-(X+8,140),PRESET,BF:NEXTX
1605 PMODE3,1:X=0:Y=58:COLOR2,1
1610 CV=256:FORX=0TO128STEP2:LIN
E(X,79)-(X,Y),PSET:PSET(X,Y,4):P
SET(X,Y+1,4):LINE(CV,79)-(CV,Y),
PSET:PSET(CV,Y,4):PSET(CV,Y+1,4)
:CV=CV-2:Y=Y-2+RND(3):IFY<54THEN
Y=56ELSEIFY>76THENY=74
1612 NEXTX
1625 COLOR1,0:FORX=0TO255STEP23:
LINE(X,74)-(X+8,79),PRESET,BF:NE
XTX
1630 PMODE4,1:COLOR1,0
1635 LINE(0,138)-(255,138),PSET:
LINE(0,134)-(255,134),PSET:LINE(
0,73)-(255,73),PSET:LINE(0,77)-(
255,77),PSET
1640 FORI=2TO250STEP16:LINE(I,99
)-(I+8,99),PSET:NEXTI
1700 U=USR2(0)
1999 RETURN
2000 P1$="O3V30L255T255C2"
2010 P2$="O3V30L5T5GFEDDEFGGFEDD
EFGFV5FV30DV5DV30CV5CV30L3T3FV5L
8FV30"
2020 P3$="O3V30L10T10DP7D"
2025 P4$="O3V30L255T255FO2AO3A"
2200 TM$="BM+3,0U6L3R6BR2BD6;BM+
5,-6L4R2D6L2R4BR3;BM+1,0U6R1F2D1
U1E2R1D6BR3;BM+5,0L4U6R4BD3BL2L1
BD3BR6;"
2999 RETURN
5000 '
5020 ML$="8E0C41108E0C40A680A7A0
8C101F26F78E0FFF108E101FA684A7A4
3088E031A8E08C0C5F26F139"
5025 TT=16300:DEFUSR0=TT:GOSUB55
00
5030 ML$="8E16A1108E16A0A680A7A0
8C17DF26F78E17BF108E17DFA684A7A4
3088E031A8E08C167F26F139"
5035 TT=16200:DEFUSR1=TT:GOSUB55
00
5060 ML$="8E1260A684A71FA7890240
30018C128026F18E125FA684A78820C6
00E78439"
5065 TT=16100:DEFUSR2=TT:GOSUB55
00
5100 ML$="8E143EA684A701301F8C12
8026F58E14208600A7843088E08C1280
26F639"
5105 TT=16000:DEFUSR3=TT:GOSUB55
00
5110 ML$="8E1021A684A71F30018C11
FF26F58E11FF8600A7843088E08C101F
26F639"
5115 TT=16150:DEFUSR4=TT:GOSUB55
00
5120 ML$="8E14E1A684A71F30018C16
BF26F58E16BF8600A7843088E08C14DF
26F639"
5125 TT=16050:DEFUSR5=TT:GOSUB55
00
5490 RETURN
5500 FORI=1TOLEN(ML$)STEP2:AA$=M
ID$(ML$,I,2):AA=VAL("&H"+AA$):PO
KETT+VV,AA:VV=VV+1:NEXTI:VV=0
5505 RETURN
6000 X=100:Y=100:Z=2
6005 PMODE4,1:SCREEN1,1:PMODE3,1
6010 PSET(X,Y,Z)
6012 IFPEEK(341)=247THENY=Y-1
6014 IFPEEK(342)=247THENY=Y+1
6016 IFPEEK(343)=247THENX=X-2
6018 IFPEEK(344)=247THENX=X+2
6020 GOTO6010
8000 PMODE4,1:COLOR1,0:DRAW"C1S4
BM10,14;XTM$;"
8005 TM=50
8008 SX=50:SY=8
8010 PMODE3,1:COLOR3,1:FORT=1TOT
M:LINE(SX,SY)-(SX,SY+6),PSET:SX=
SX+4:NEXTT
8020 PMODE4,1:COLOR1,0
8099 RETURN
8100 SX=SX-4:PMODE4,1:COLOR1,0:L
INE(SX,SY)-(SX,SY+6),PRESET
8110 TM=TM-1:IFTM<1THEN200
8190 RETURN
8200 PMODE4,1:COLOR1,0:CX=74:CY=
20:LINE(CX,CY)-(CX+120,CY+12),PR
ESET,BF
8205 IFNB<=0THEN8220
8210 FORI=1TONB:PUT(CX,CY)-(CX+2
4,CY+10),BU,OR:CX=CX+28:NEXTI
8220 '
8299 RETURN
9000 CLS:POKE65494,0:PL$="V30EV1
0EV30FEFEV10EV30FGFEV10EV30DCBGV
10GV30FEDCV10CV30P5;O+;CV10C"
9005 PLAY"O1V30L6T6;XPL$;"
9010 SCREEN0,0:PRINT"YOU HAVE SQ
UASHED YOUR LAST      POSSUM. TOU
GH-LUCK,EH?"
9015 FORI=1TO800:NEXTI:PLAY"O3V3
0L7T7;XPL$;":PRINT:PRINT"YOUR FI
NAL SCORE WAS"S1"PTS.":FORI=1TO7
00:NEXTI
9020 PRINT:INPUT"PLAY AGAIN (Y/N
)";I$:IFLEFT$(I$,1)="Y"THENRUN E
LSEIFLEFT$(I$,1)="N"THEN9030ELSE
9020
9030 PLAY"V30L8T8":PRINT"GAME OV
ER":FORO=1TO5:PLAY"O"+STR$(O):FO
RI=1TO12:PLAYSTR$(I):NEXTI:NEXTO
9040 PLAY"O5V30L4T4;12;P1":PLAY"
O1L6T6CV5CV30L2T2CV5C"
9045 END
```

END

↙See List of Advertisers on page 130

BY DENNIS ELFERT

GIVE YOUR DISK DRIVE A ONE-TRACK MIND

I/O error blues from a crashed disk? Speed your back-up copy procedure with this CoCo utility.

Anyone who owns a Radio Shack disk system knows the phrase I/O error all too well—another crashed disk. But what can you do about it? Most people have back-ups of all their disks, so they reformat the blown disk and recopy from the back-up. For those of us who have only one drive, however, this is a slow process. In a normal Back-up procedure for a one-drive system it takes six changes of the two disks to completely copy a disk. There is a faster way to do this.

To solve the problem you need to know how the Radio Shack disk operating system stores things on a disk. The disk is divided into 35 tracks numbered 0–34, and each of these tracks is divided into 18 sectors. The DOS uses track 17 as its directory track, and on it stores things like the names and types of programs (Basic, binary, data), how long the program is, and where it is stored on the disk.

Approximately 95 percent of all I/O errors come from a blown track 17. You can tell if you have a crashed track

System Requirements

16K RAM
Disk Basic
One Disk Drive

Program Listing. Directory Disk

```
50 '**  MENU  **
55 CLEAR3500:DIM NI$(64),TI$(34)
,EL(34):CLS3:PRINT@64,"":PRINT:P
RINT
60 PRINT@69,"DIRECTORY DISK PROG
RAM":PRINT@108,"VER.1.1":PRINT@1
37,"DENNIS ELFERT"
65 PRINT@192,"1. SAVE DIRECTORY
TO D.D.":PRINT@224,"2. DISPLAY C
ATALOG ON D.D":PRINT@256,"3. REP
AIR DAMAGED DIRECTORY":PRINT@288
,"4.
 ERASE DIRECTORY ON D.D":PRINT@3
20,"5. DISPLAY A DIRECTORY":PRIN
T@352,"6. END"
70 PRINT@416,"(D.D. REFERS TO DI
RECTORY DISK) ";
75 PRINT@486,"INPUT 1,2,3.4,5, O
R 6";
80 A$=INKEY$:IF A$=""THEN80
85 A=VAL(A$):IF A<=0 OR A>6 THEN
 55
87 IF A=6 THEN CLS:END
90 CLS:IF A>1 AND A<6 THENPRINT@
224,"INSERT DIRECTORY DISK & <EN
TER>":ELSE IFA=1 THENPRINT@225,"
INSERT DISK TO LOAD DIR. TRACK":
PRIN
T@263,"AND PRESS <ENTER>"
95 A$=INKEY$:IF A$="Q"THEN 55 EL
SE IF A$<>CHR$(13) THEN 95 ELSE
ON A GOTO 105,205,305,405,505,60
0
100 '** SAVE DIRECTORY **
105 CLEAR 4000
110 T=0:X=0:Y=0:DIM D$(10),E$(10
),F$(35)
115 FORX=2 TO 11
120 DSKI$ 0,17,X,D$(Y),E$(Y)
125 Y=Y+1
130 NEXTX
135 CLS:PRINT@224,"INSERT DIRECT
ORY DISK & <ENTER>"
140 A$=INKEY$:IF A$=CHR$(13)THEN
145 ELSE IF A$="Q"THEN 55 ELSE 1
40
145 GOSUB 1000
150 FORX=1TO2:FORY=1TO17
155 F$(Y*X)=MID$(B$(X),Z*15+2,1)
:F=ASC(F$(Y*X))
157 T=T+1
160 IF F=255 THEN 170
165 Z=Z+1:NEXTY:Z=0:NEXTX:CLS:PR
INT@235,"DISK FULL":FORX=1 TO 65
0:NEXTx:GOTO55
170 CLS:PRINT@234,"NAME OF DISK"
:PRINT@266,"";:INPUTNM$:IF LEN(N
M$)>12 THEN 170
175 IF LEN(NM$)=12 THEN 185
180 NM$=NM$+" ":GOTO175
185 MID$(B$(X),Z*15+2,1)="O":MID
$(B$(X),Z*15+3,12)=NM$
190 GOSUB 2000:X=0:Y=0:FORX=2TO1
1
195 DSKO$ 0,T,X,D$(Y),E$(Y):Y=Y+
1:NEXTX:GOTO55
200 '** DISPLAY CATALOG**
205 CLS3:PRINT@4,"DIRECTORY DISK
 CATALOG";
210 GOSUB 3000
220 PRINT@416,"<ENTER> TO CONTIN
UE":PRINT@448,"<Q> TO QUIT"
225 A$=INKEY$:IF A$=CHR$(13) THE
N 230 ELSE IF A$="Q"THEN 55 ELSE
225
230 CLS3:PRINT@4,"DIRECTORY DISK
 CATALOG";
235 GOSUB3500
240 PRINT@416,"<ENTER> TO SEE AG
AIN"
245 PRINT@448,"<Q> TO QUIT"
250 A$=INKEY$:IF A$=CHR$(13)THEN
 205 ELSE IF A$="Q"THEN 55 ELSE
250
300 '**REPAIR DAMAGED DIRECTORY
305 CLS3:PRINT@5,"DIRECTORY REPA
IR MODE";
310 GOSUB 3000:PRINT@422,"<ENTER
> TO CONTINUE";:PRINT@454,"<S> T
O SELECT DIR. ";:PRINT@486,"
<Q> TO QUIT    ";
315 A$=INKEY$:IFA$=CHR$(13)THEN
GOTO320ELSE IF A$="S"THEN 330 EL
SE IF A$="Q"THEN55 ELSE 315
320 CLS3:PRINT@5,"DIRECTORY REPA
IR MODE";:GOSUB3500:PRINT@422,"<
ENTER> TO SEE AGAIN";:PRINT@454,
"<S> TO SELECT DIR.  ";:PRINT@48
6,"
   <Q> TO QUIT     ";
325 A$=INKEY$:IFA$=CHR$(13) THEN
 305 ELSE IFA$="S"THEN330 ELSE I
F A$="Q"THEN55ELSE325
330 PRINT@416,"ENTER NUMBER OF D
IRECTORY":PRINT@448,"":PRINT@490
,"           ";:PRINT@442,"";:I
NPUT T
335 X=0:Y=1:FORX=2TO11
340 DSKI$ 0,T,X,D$(Y),E$(Y)
345 Y=Y+1:NEXTX:X=0:Y=0
350 CLS:PRINT@225,"INSERT DAMAGE
D DISK & <ENTER>"
355 A$=INKEY$:IFA$<>CHR$(13)THEN
355
360 FOR X=2 TO11:Y=Y+1
365 DSKO$0,17,X,D$(Y),E$(Y)
370 NEXTX
375 GOTO 55
400 '**KILL DIRECTORY ON D.D.**
405 CLS3:PRINT@5,"DELETE DIRECTO
RY MODE";
410 GOSUB3000:PRINT@416,"<ENTER>
TO CONTINUE":PRINT@448,"<K> TO K
```

Listing continued

17 if you type in DIR and it returns an I/O error. This is due to the fact that track 17 is constantly being accessed. Almost all disk commands use track 17 in one way or another.

The Back-up procedure recopies all 35 tracks, but only one track is bad, so you recopy 34 tracks for nothing.

Instead of making a complete back-up of all 35 tracks to another disk, the Program Listing copies only track 17. Now, instead of needing one whole disk to back up another, you can have one disk backing up 34 other disks. With this program you can store up to 34 different track 17s on one disk, called the Directory Disk, since it stores only the directory tracks of other disks. In this program 34 out of the 35 tracks hold directories while the remaining track, called the catalog, is for the program's own directory.

Once you have a bug-free program, run it. You will be greeted with the main menu of six options, each explained in detail below.

● Save Directory to D.D. (Directory Disk): Choose this when you want to make a back-up of one of your disk's directories. It tells you to insert the disk you want to back up. Once you have inserted the disk and pressed the enter key, the program copies the contents of track 17 into arrays D$ and E$. Insert your Directory Disk. The program checks to see if there are any open tracks, and if so it prompts you for the name of the disk (up to 12 letters). It then saves the track to the Directory Disk. Make sure that no two entries to the Directory Disk have the same name, since this causes problems in other features of the program.

● Display Catalog on D.D.: This displays the names of the directories on your Directory Disk.

● Repair Damaged Directory: This is used when you get an I/O error on a disk's track 17. After inserting your Directory Disk, the program displays the catalog of directories. Find the name of the blown disk and enter its number. The program then loads into memory the directory track of the blown disk. Insert the damaged disk and it copies the directory track in memory on track 17 of the blown disk. You should now have a good disk.

● Erase Directory on D.D.: This mode is similar to Disk Basic's KILL command, and it erases unwanted directories on your Directory Disk to make that track available for storing others.

● Display a Directory: This prompts you to insert the Directory Disk that displays the list of saved directory tracks. From here you can request a display.

● End: Returns control to Basic. In all modes you can return to the main menu at any time by pressing Q.

I set the program up on a 32K machine, but it runs on a 16K machine if you first enter PMODE0:POKE 25,14 POKE 3584,0: NEW <ENTER>. This frees the 6K grabbed by Basic for graphics. ■

Address correspondence to Dennis Elfert, 315 Richard Drive, Houma, LA 70360.

"With this program you can store up to 34 different track 17s on one disk, called the Directory Disk, since it stores only directory tracks of other disks."

```
ILL",:PRINT"<Q> TO QUIT"
415 A$=INKEY$:IF A$=CHR$(13)THEN
 GOTO 420 ELSE IF A$="K"THEN GOT
O 430 ELSE IF A$="Q" THEN 55 ELS
E 415
420 CLS3:PRINT@5,"DELETE DIRECTO
RY MODE";:GOSUB3500:PRINT@416,"<
ENTER> TO SEE AGAIN":PRINT@448,"
<K> TO KILL",:PRINT"<Q> TO QUIT
"
425 A$=INKEY$:IF A$=CHR$(13) THE
N GOTO 405 ELSE IF A$="K" THEN G
OTO 430 ELSE IF A$="Q" THEN GOTO
 55 ELSE 425
430 PRINT@416,"":PRINT@448,"":PR
INT@416,"NUMBER OF FILE TO BE KI
LLED";:INPUT K:PRINT@451,"CONFIR
MATION(YES OR NO)";:INPUT A$:IF
A$="
YES" THEN GOTO 440 ELSE 55
440 IF K<1 OR K>34 THEN 55 ELSEG
OSUB1000:Z=0:T2$=TI$(K):FORY=1TO
34:IFY>17THENX=2 ELSE X=1
445 IF MID$(B$(X),Z*15+3,12)=T2$
 THEN GOTO 460
450 Z=Z+1:IF Y=17 GOSUB 452
451 NEXTY:CLS:PRINT@264, "DIRECT
ORY NOT FOUND":FOR X=1 TO 1000:N
EXT:GOTO55
452 Z=0:RETURN
460 MID$(B$(X),Z*15+2,1)=CHR$(25
5):MID$(B$(X),Z*15+3,12)="
    "
470 GOSUB 2000:GOTO55
500 '** DISPLAY A DIRECTORY FROM
 D.D.**
505 CLS3:PRINT@5,"DIRECTORY DISP
LAY MODE";
510 GOSUB3000:PRINT@416,"<ENTER>
 TO CONTINUE":PRINT@448,"<S> TO
SELECT",:PRINT"<Q> TO QUIT"
515 A$=INKEY$:IF A$=CHR$(13) THE
N 520 ELSE IF A$="S" THEN 530 EL
SE IF A$="Q" THEN 55 ELSE 515
520 CLS3:PRINT@6,"DIRECTORY DISP
LAYS";:GOSUB3500:PRINT@416,"<ENT
ER>TO CONTINUE":PRINT@448,"<S> T
O SELECT",:PRINT"<Q> TO QUIT"
525 A$=INKEY$:IF A$=CHR$(13) THE
N 505 ELSE IF A$="S" THEN 530 EL
SE IF A$="Q"THEN 55 ELSE 525
530 PRINT@448,"":PRINT @416,"ENT
ER NUMBER OF DIRECTORY";:INPUT V
:PRINT@448,""
532 IF V<0 OR V>34 THEN 55
535 FOR X=3 TO 11
540 DSKI$ 0,V,X,D$,E$
545 F$=D$+LEFT$(E$,127)
550 FORZ=0TO7:NM$=MID$(F$,Z*32+1
,8):EXT$=MID$(F$,Z*32+9,3)
560 IF ASC(NM$)=0 THEN 575 ELSE
IF ASC(NM$)=255 THEN 585
565 Q=Q+1:NI$(Q)=NM$+"/"+EXT$
575 NEXT Z
585 NEXT X
587 R=1
595 CLS3:PRINT@5,"DIRECTORY NUMB
ER";V;:Y=64
600 FOR X=R TO Q
605 PRINT@Y,NI$(X);
610 IF X/24=INT(X/24) THEN 630
615 Y=Y+16:NEXTX
630 PRINT@484,"<ENTER> TO CONTIN
UE";
635 A$=INKEY$:IF A$=CHR$(13) THE
N 640 ELSE 635
640 IF X>Q THEN 55 ELSE R=X+1:GO
TO 595
1000 DSKI$ 0,0,1,A$(1),A$(2):DSK
I$ 0,0,2,A$(3),A$(4)
1010 A$(2)=LEFT$(A$(2),127):A$(4
)=LEFT$(A$(4),127)
1020 B$(1)=A$(1)+A$(2):B$(2)=A$(
3)+A$(4)
1030 A$(1)="":A$(2)="":A$(3)="":
A$(4)=""
1040 RETURN
2000 A$(1)=LEFT$(B$(1),128):A$(2
)=RIGHT$(B$(1),127)+"0":A$(3)=LE
FT$(B$(2),128):A$(4)=RIGHT$(B$(2
),127)+"0"
2010 B$(1)="":B$(2)=""
2020 DSKO$ 0,0,1,A$(1),A$(2):DSK
O$ 0,0,2,A$(3),A$(4)
2030 RETURN
3000 '**DISPLAY DIRECTORY**
3005 GOSUB 1000
3010 Z=0:FORY=1 TO34
3015 IF Y<=17 THEN X=1 ELSE X=2
3020 TI$(Y)=MID$(B$(X),Z*15+3,12
):EL(Y)=ASC(MID$(B$(X),Z*15+2,1)
)
3025 Z=Z+1:IFY=17 THEN Z=0
3030 NEXT Y
3035 D=64:FORX=1TO10:IF EL(X)=25
5 THEN TI$(X)="unused track"
3037 X$=RIGHT$(STR$(X),2)
3040 PRINT@D,X$;:PRINT@D+3,TI$(X
);
3045 D=D+32:NEXTX
3050 D=80:FORX=11TO20:IF EL(X)=2
55 THEN TI$(X)="unused track"
3053 X$=RIGHT$(STR$(X),2)
3055 PRINT@D,X$;:PRINT@D+3,TI$(X
);
3060 D=D+32:NEXTX
3065 RETURN
3500 D=64:FORX=21TO27:IF EL(X)=2
55 THEN TI$(X)="unused track"
3503 X$=RIGHT$(STR$(X),2)
3510 PRINT@D,X$;:PRINT@D+3,TI$(X
);
3520 D=D+32:NEXTX
3530 D=80:FORX=28TO34:IFEL(X)=25
5 THEN TI$(X)="unused track"
3533 X$=RIGHT$(STR$(X),2)
3540 PRINT@D,X$;:PRINT@D+3,TI$(X
);
3550 D=D+32:NEXTX
3560 RETURN
```

END

BY JOHN R. KELTY

YOU LIGHT UP MY MULTI-PAK

The RS Multi-Pak Interface is a boon to hardware hackers. Here's why, and a nifty LED add-on project.

The Radio Shack Multi-Pak Interface (MPI) is the best peripheral to come along for the CoCo since disk drives. It is a great convenience to novice users, but more importantly to me, it allows for some interesting interfacing and programming projects. I've taken a look at what's inside this wonderful device, and what follows are my findings plus instructions on how to add on/off and slot-selection LEDs to the MPI.

Strange POKEs

The software POKEs used for selecting a slot are given in the owner's manual with an addenda sheet as:

POKE 65407,0	Slot 1
POKE 65407,17	Slot 2
POKE 65407,34	Slot 3
POKE 65407,51	Slot 4

Why do they look so odd? The numbers 0, 17, 34, and 51 written in binary or in hex are enlightening:

Decimal	Hex	Binary
0	0	00000000
17	11	00010001
34	22	00100010
51	33	00110011

See the pattern? There are four expansion slots and two separate counters. Unlike other expansion devices the CTS and SCS signals are individually switched in the MPI with a decoder circuit.

CTS is the cartridge-select signal used to select the address space $C000 through $FFEF ($C000, decimal 49152, is where most ROM packs begin). SCS is the spare-select signal selecting the address space $FF40 through $FF5F. Some internal investigation and testing with homemade peripherals showed that the CTS and SCS signals can be independently selected, and that an LED display of these selections would be a desirable feature on the MPI.

The simplest way to program these lines is by POKEing either address

POKE VALUE into 65407 or 65439	CTS	SCS SLOT NUMBER	* BOTH
* 0	1	1	1
1	1	2	
2	1	3	
3	1	4	
16	2	1	
* 17	2	2	2
18	2	3	
19	2	4	
32	3	1	
33	3	2	
* 34	3	3	3
35	3	4	
48	4	1	
49	4	2	
50	4	3	
* 51	4	4	4

* Indicates values given in the Radio Shack Owner's Manual for POKEs. Manual shows address 65439 ($FF9F) with addenda sheet change to 65407 ($FF7F).

Note that you can add 4, 8, 64, or 128 to any of these POKEs and still get the same result.

Example: POKE 65407,255 would select both CTS and SCS for slot number 4, the same as using POKE 65407,51.

Table 1. Slot Select Lines (CTS and SCS) Minimum Decimal POKE Values

Photo 1. The MPI complete with on/off and slot-selection LEDs installed.

Photo 2. Here the internal LED wiring is displayed.

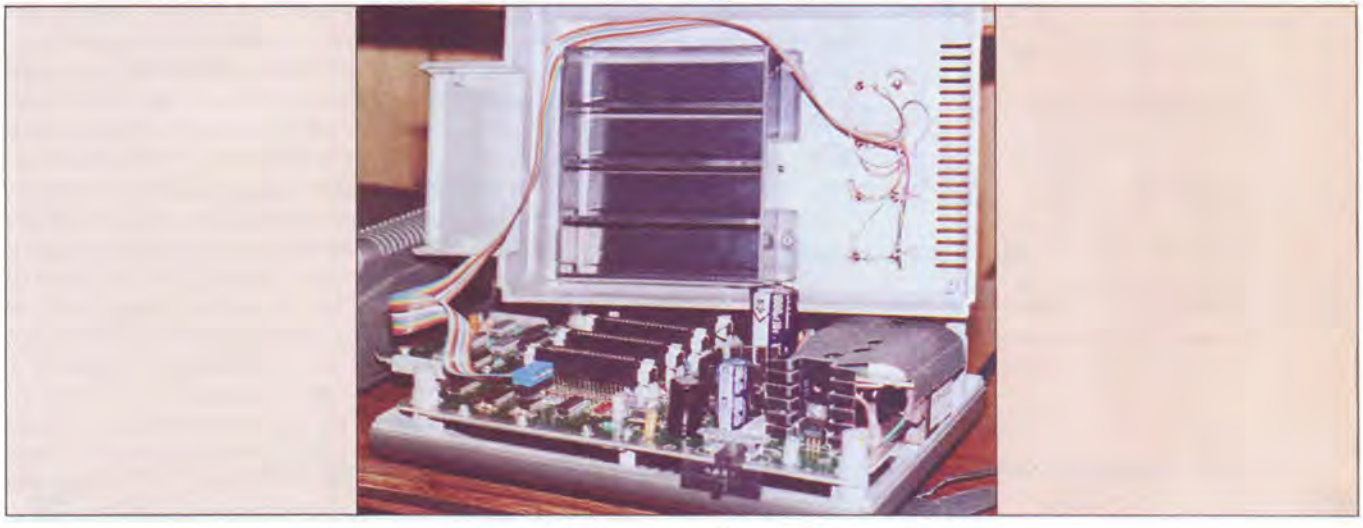


Photo 3. Left. Interior view of completed project.

65407 or 65439 as shown in Table 1. You can POKE other values, but these are redundant since the bits for 4, 8, 64, and 128 do nothing. For example, POKE 65407,48 selects the CTS of slot 4 and the SCS of slot 1. You could get the same result with POKE 65407,52, POKE 65407,56, POKE 65407,112, or POKE 65407,176. You could also POKE these numbers into 65439 for the same result.

If you had the disk controller in slot 4 (as the owner's manual suggests) and another hardware device that used the SCS line in slot 1, you could use the Disk Basic ROM with your hardware device and switch back and forth from one hardware device to another. It works, and it is a great tool for hackers who want to save a couple of chips per hardware project that are typically needed for address decoding. Now you must add some indicator LEDs to let you know which slot-select lines are enabled.

The LED Project

When you open the MPI you void the warranty by breaking the seal over one of the four Phillips-head screws used to secure the lid. Inside is a near carbon copy of the CoCo power supply: +5V dc with pass-transistor, +12V dc regulator with heat-sink, and the −12V dc regulator without a heat-sink.

There is plenty of power here. There is also a socketed PAL (programmable array logic, 14L4CN, 20-pin DIP) device that is probably used to memory-map the MPI I/O to the CoCo expansion bus.

These interesting devices are quickly changing standard logic design practices. This one chip obviously makes the I/O implementation easy, even if it requires changes in memory address (such as 65407 or 65439). A service manual should describe the function of the PAL in more detail. There are of course buffers (74LS367s and LS245) with some other logic ICs, and a nice shield on the bottom side of the board, but you do not need to remove any of this to perform this add-on modification.

To indicate the CTS and SCS status of each slot you need to duplicate the action of the 74LS139 dual 2-to-4 line decoders/multiplexers circuit within the MPI (see Fig. 1). This is IC U13 and is located next to slot 1 (J1).

Place a separate 74LS139 on top of U13 (piggyback) with pins 1, 4, 5, 6, 7, 9, 10, 11, 12, 15, and 16 bent outward. You only need to solder pins 2, 3, 8, 13, and 14 of the add-on LS139 to U13. Pins 16 and 8 are the +5V dc and ground pins, respectively.

Use a short piece of small wire to connect pin 16 of the piggyback LS139 (not of U13) to the positive lead of the 100-microfarad capacitor C30 found between the J1 and J2 connectors. The +5V dc supply at pin 16 of U13 is from the CoCo and would not indicate whether the MPI power is on or off.

Be careful when soldering this wire to

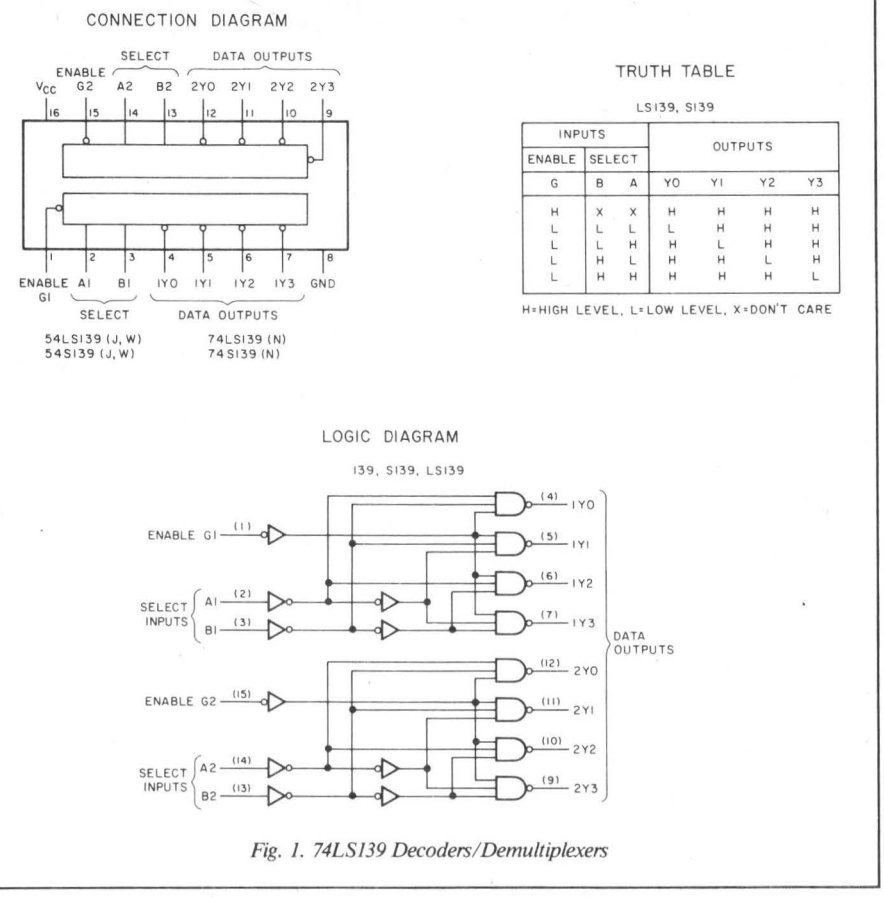


Fig. 1. 74LS139 Decoders/Demultiplexers

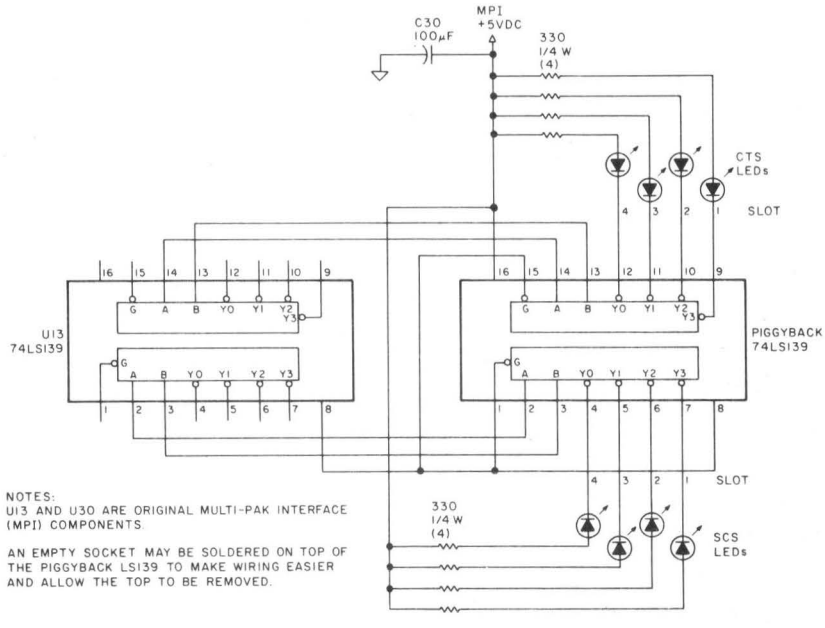


Fig. 2. Schematic for LED Add-on

```
10 'REMOVE ALL PAKS FROM MULTI-PAK INTERFACE
20 A=65407:'OR USE A=65439
30 FOR V=0 TO 255
40 POKE A,V
50 FOR D=1 TO 10:NEXTD:'DELAY
60 NEXT V
```

Program Listing. POKE Combinations to Test LEDs

the capacitor lead, since all around is ground and could easily short out the MPI +5V dc supply. Pins 2 and 3 of the LS139 ICs are used to select the slot for the SCS line, and pins 13 and 14 are used to select the slot for the CTS line.

Connect pins 1 and 15 on the piggyback IC to pin 8 (ground). This permanently enables the selection outputs used to power the LEDs (see Fig. 2).

To keep the case top and base easy to separate, I used a 16-pin DIP connector with about 18 inches of flat ribbon cable to reach from the LS139 to where the LEDs are on the case top. You can unplug the LEDs if desired or set the top next to the base in case you have to repair the MPI.

Solder an empty 16-pin DIP socket on top of the new LS139 (all 16 pins). Strictly a job for confident hackers, this requires a neat and careful assembly of the piggyback arrangement to avoid any troubles. If one of the LEDs does not light, this area is the first to search for questionable solder connections.

I drilled holes in the top of the case just large enough for the LEDs that I had on hand. Be sure to use eight LEDs that are alike. Usually the long lead of the LED is the anode (connect this to +5V dc). I positioned the LEDs so that a +5V dc bus could be made in a straight line between the two rows of four LEDs and used Super Glue to hold them in place. I recommend a drill press for a neat job, but LED sockets will probably cover up any accidents.

Finish the wiring as per the schematic, soldering the resistors directly to the flat cable and LED cathodes (short leads). I don't recommend a board since it takes up too much space. There is a larger filter capacitor next to slot 3 and the power transformer is next to that. I had no problems placing the LEDs over the transformer area directly across from the slot numbers, but there is no room over the large capacitor for anything. Be sure to stay clear of this area. I used transfer labeling to mark which column of LEDs are for CTS and SCS.

To test your lamps, plug in the MPI to the CoCo. (Make sure all power is off.) Turn on the MPI and then the CoCo. The LEDs are powered by the MPI, so they act as the MPI power-on indicator. Due to the LS139 IC, there will always be one LED lit (and only one) in each column when the power is on. The LEDs next to the slot chosen by the front selector switch should both be on.

No matter what position the front slide switch is in, slot 4 is constantly

These parts may be purchased from Digi-Key Corporation, Highway 32 South, P.O. Box 677, Thief River Falls, MN 56701. 1-800-346-5144.

Single-ended, 2-foot DIP jumper cable assembly with flat ribbon R114-24-ND	$ 2.46
Wide-angle LED T-1 3/4 NSL6053	$ 2.20 /10
16-pin DIP socket (TIN) C8916	$ .16
Dual 2/4 decoder IC 74LS139N	$ .66
1/4-Watt carbon resistors 330Q	$ .50 /10
Total	$ 5.98

Table 2. Parts List

selected if the CoCo power is off. This is why the owner's manual suggests using slot 4 for the disk drive. With the CoCo and the Multi-Pak on, move the selector switch from 1 through 4 and verify that the LEDs switch on in pairs by slots 1 through 4 respectively. If they do not light, check your wiring, especially the soldering near the LS139. If they do not light in the correct order, check the wiring out of the flat ribbon to the LED resistors.

Next, POKE the values listed in Table 1 to see if all combinations work. If they do not, check for errors and verify that the number was correctly POKEd by PEEKing the same location. The Program Listing shows you all combinations (including the redundant POKEs) and makes an interesting display.

Do not try any of these tests with cartridges in the MPI since they may or may not become activated. Once you know the lights work, you can confidently insert your disk controller or other packs or accessories (turn all power off and wait for the LEDs to go off), and explore some real expansion capabilities. With the help of these LED slot-selection (and power-on) indicators, the Radio Shack Multi-Pak Interface opens a new world of computer interfacing and programming to CoCo users. ■

Address correspondence to John R. Kelty, 1440 N. 61st, Lincoln, NE 68505.

See List of Advertisers on page 130

BY STEPHEN TITCHENAL

BUILD A BETTER ERROR TRAP...

Make your computer work harder for you. Edit more efficiently with this machine-language routine.

You have just finished typing in additions to your program; a new routine adds that important bell or whistle to make it outstanding. You run it, feeling it's a job well done, but the computer doesn't feel the same way. All it prints out is "FC ERROR IN 211."

After hours of debugging Basic programs, I decided to make the computer work a little harder and give me more information about the error. The Assembly-language routine in Program Listing 1 indicates where within a Basic line an error occurred. Then it enters the Extended Basic Edit routine automatically. You no longer need to type EDIT 211.

You need Extended Basic to use this routine, and Disk Basic works without modification. The code is position independent and can be located in any protected RAM location.

The computer's Basic interpreter uses a subroutine stored in RAM locations $9F to $AA to get characters from the Basic line, one at a time. When the interpreter discovers an error, locations $A6 and $A7 point to the last character in the Basic line that was read by the interpreter.

Basic lines are stored in tokenized form. All Basic commands are stored as 1- or 2-byte numbers called tokens. This saves space and makes program execution faster, but complicates the line reading for the operator.

In order for the error location to be useful it must point to the location in a Basic line in detokenized form. Fortunately, the routine beginning at $B7CB spells out the commands represented by tokens and puts the detokenized line in the buffer pointed to by the Y register. A zero represents the end of a Basic line. By putting a zero in the location where the error occurred, the detokenization routine stops at the error location. Lines 350–450 save the location pointer, put the original character back in the line, and continue detokenization.

Lines 490–580 print the line number and detokenized line just as Edit would, and then enter the Edit routine at $855C.

Your only problem now is how to get the Basic interpreter to jump to the new Error routine after it completes its own error-handling routine. The interpreter jumps to RAM locations $15E to $1A8 during various operations. The jump before Basic, which outputs a character to the screen (CHROUT $A282), is important since your new Edit routine should be run after the interpreter prints —ERROR IN—.

This jump can be used to check whether that statement is being output to the screen. ERROR IN is stored in locations $ABE1 to $ABEB of the Basic ROM. The CHROUT routine expects this location to be in the X register for output.

There is one bug in this method of adding the new Error routine. ERROR and IN are called be separate lines of the ROM Basic Error routine. In order to add the new routine after IN, only IN can be checked. When a Basic program is stopped by the break key, the address of IN is also used. You are put in the edit mode at the location in the program line last executed before you pressed break.

The CHROUT routine is checked by this jump before it has completed the screen output. By changing the return address used at the end of this ROM routine, the new Error routine will not be entered until CHROUT is finished. These return addresses are stored in a stack pointed to by the S register. While in the jump table they are three deep and, since each address is 2 bytes, inserting the new Error routine start address at

System Requirements

16K RAM
Extended or Disk Color Basic
Editor/Assembler (optional)

6,S executes it at the proper time.

The subroutine in lines 110–200 inserts the new jump into the jump table. This routine allows a jump already in the table at location $167 to be performed immediately following the error-handling routine. This takes more code but maintains compatibility with other machine-language routines using this jump. The subroutine even works with non-Extended Basic, but is of no value because of the Extended Basic calls in the main program.

Program Listing 2 is a Basic program to load the machine-language routine without an editor/assembler. Lines 20 and 25 check the maximum memory in your computer and automatically clear the last 1K for the routine.

Debugging should be much easier. Not only do you know where the error is within the line, you are in the Edit routine with the cursor at the error. Use the Q command to exit the Edit mode if you do not want to make changes at this time. Q is the only Edit command that preserves the variable table if you need access to it for debugging purposes. ■

Address correspondence to Stephen Titchenal, 937 Montford, Cleveland Heights, OH 44121.

"After hours of debugging Basic programs, I decided to make the computer work a little harder... The Assembly-language routine in Program Listing 1 indicates where within a Basic line an error occurred. Then it enters the Extended Basic Edit routine automatically. You no longer need to type Edit 211."

```
                        00010 ************************************
                        00020 *COPYRIGHT 1983 BY STEPHEN TITCHENAL
                        00030 *         937 MONTFORD ROAD
                        00040 * CLEVELAND HEIGHTS, OHIO 44121
                        00050 ************************************
7F00                    00100          ORG $7F00
                        00105 *****RESET JUMP BEFORE CHARACTER OUTPUT - CHROUT ($A282
)
7F00 8E   0167          00110          LDX #$167       LOCATION OF RAM JUMP FROM CHROU
T
7F03 A6   80            00120          LDA ,X+         GET PRESENT JUMP RETURN
7F05 A7   8D 0020       00130          STA EJUMP,PCR   AND STORE IT AT
7F09 10AE 84            00140          LDY ,X          THE END OF THE
7F0C 10AF 8D 0019       00150          STY 1+EJUMP,PCR    NEW JUMP ROUTINE
7F11 31   8D 0008       00160          LEAY ERCHCK,PCR    GET ADDRESS OF NEW JUMP
7F15 10AF 84            00170          STY ,X          AND STORE IT IN JUMP TABLE
7F18 86   7E            00180          LDA #$7E        6809 JMP
7F1A A7   1F            00190          STA -1,X        STORE IT IN JUMP TABLE
7F1C 39                 00200          RTS
                        00205 ****CHECK IF "ERROR IN" LOCATION IS BEING PRINTED****
7F1D 8C   ABEA          00210 ERCHCK   CMPX #$ABEA     PRINTING "ERROR IN"?
7F20 26   07            00220          BNE EJUMP       NO - DO NORMAL JUMP
7F22 31   8D 0006       00230          LEAY ERROR,PCR  GET ADDRESS OF NEW ERROR ROUTIN
E
7F26 10AF 66            00240          STY 6,S         STORE AS LAST RETURN ADDRESS
7F29 39                 00250 EJUMP    RTS             SAVE 3 BYTES FOR HOOK RETURN AD
DRESS
7F2A 39                 00260          RTS
7F2B 39                 00270          RTS
                        00274 ************************************
                        00275 *****NEW ERROR HANDLING ROUTINE*****
                        00276 ************************************
7F2C BD   B958          00280 ERROR    JSR $B958       CLEAR TO END OF LINE ON SCREEN
7F2F DC   68            00290          LDD $68         GET CURRENT LINE #
7F31 DD   2B            00300          STD $2B         STORE IT
7F33 BD   BDCC          00310          JSR $BDCC       PRINT CURRENT LINE #
7F36 BD   B9AC          00320          JSR $B9AC       PRINT SPACE
7F39 BD   AD01          00330          JSR $AD01       PUT LOCATION OF CURRENT LINE #
IN REGISTER X
7F3C 25   47            00340          BCS ERREND      DO ERROR IF CURRENT LINE # NOT
CORRECT
7F3E 108E 02DD          00350          LDY #$2DD       LOCATION OF KEYBOARD BUFFER
7F42 A6   9F 00A6       00360          LDA [$A6]       LOCATION AT END OF ERROR IN BAS
IC LINE
7F46 34   02            00370          PSHS A          SAVE THAT CHARACTER
7F48 6F   9F 00A6       00380          CLR [$A6]       SET TO 0 TO END SUBROUTINE AT E
RROR
7F4C 30   04            00390          LEAX 4,X        GET START OF BASIC  LINE LISTIN
G
7F4E BD   B7CB          00400          JSR $B7CB       DETOKENIZE UP TO LINE ERROR (SE
T TO 0)
7F51 35   02            00410          PULS A          GET CHARACTER REPLACED BY 0
7F53 A7   9F 00A6       00420          STA [$A6]       PUT IT BACK
7F57 30   1F            00430          LEAX -1,X       RESET FOR DETOKENIZING REST OF
LINE
7F59 34   20            00440          PSHS Y          SAVE LOCATION OF ERROR IN BUFFE
R
7F5B BD   B7CB          00450          JSR $B7CB       DETOKENIZE REST OF LINE
7F5E 1F   20            00460          TFR Y,D         PUT END OF BUFFER IN D FOR COMP
UTATION
7F60 83   02DE          00470          SUBD #$2DE      GET LENGTH OF BUFFER
7F63 DD   D7            00480          STD $D7         SAVE FOR EDIT ROUTINE
7F65 8E   02DD          00490          LDX #$2DD       START OF BUFFER
7F68 BD   85B4          00500          JSR $85B4       PRINT BUFFER
7F6B BD   B958          00510          JSR $B958       CLEAR TO END OF LINE
7F6E BD   BDCA          00520          JSR $BDCA       PRINT CURRENT LINE #
7F71 BD   B9AC          00530          JSR $B9AC       PRINT SPACE
7F74 8E   02DD          00540          LDX #$2DD       BUFFER LOCATION
7F77 35   06            00550          PULS D          GET LOCATION OF ERROR IN  BUFFE
R
7F79 83   02DD          00560          SUBD #$2DD      COMPUTE DISTANCE FROM BUFFER ST
ART
7F7C BD   85B6          00570          JSR $85B6       PRINT BUFFER TO END OF ERROR
7F7F BD   855C          00580          JSR $855C       DO EDIT ROUTINE
7F82 7E   AC73          00590          JMP $AC73       GO TO BASIC COMMAND LOOP
7F85 7E   AED2          00600 ERREND   JMP $AED2       DO UL ERROR
          0000          00610          END
00000 TOTAL ERRORS

EJUMP   7F29
ERCHCK  7F1D
ERREND  7F85
ERROR   7F2C
```

Program Listing 1. Errorbas

```
10 IF PEEK(&H80C0)=18 THEN GOTO2
0
15 PRINT"PROGRAM WILL NOT WORK W
ITHOUT EXTENDED BASIC": END
20 CLEAR200,PEEK(&H74)*256
25 X=PEEK(&H74)*256
30 FOR Y=X TO X+133
40 READ Z$: Z=VAL("&H"+Z$)
50 POKE Y,Z
60 C=C+Z
70 NEXTY
80 IF C<>15312 THEN PRINT"DATA E
NTRY ERROR": END
90 EXECX
100 PRINT"PROGRAM SUCCESSFULLY L
OADED TO"
110 PRINT"HEX LOCATIONS: "HEX$(X
)" TO "HEX$(X+133)
200 DATA 8E,01,67,A6,80,A7,8D,00
,20,10,AE,84,10,AF,8D,00,19
210 DATA 31,8D,00,08,10,AF,84,86
,7E,A7,1F,39,8C,AB,EA,26,07
220 DATA 31,8D,00,06,10,AF,66,39
,39,39,BD,B9,58,DC,68,DD,2B,BD,B
D,CC
230 DATA BD,B9,AC,BD,AD,01,25,47
,10,8E,02,DD,A6,9F,00,A6,34,02
240 DATA 6F,9F,00,A6,30,04,BD,B7
,CB,35,02,A7,9F,00,A6,30,1F,34,2
0
250 DATA BD,B7,CB,1F,20,83,02,DE
,DD,D7,8E,02,DD,BD,85,B4,BD,B9,5
8
260 DATA BD,BD,CA,BD,B9,AC,8E,02
,DD,35,06,83,02,DD,BD,85,B6,BD,8
5,5C
270 DATA 7E,AC,73,7E,AE,D2
```

Program Listing 2. Error Edit Routine

BY MARTIN TICHBORNE

WORDSEARCH

No more scavenging the magazine racks hunting for word puzzles; now you can design them yourself with your CoCo. Computer Wordsearch is similar to an ordinary wordsearch puzzle, with the words placed in a grid vertically, horizontally, backwards, forwards, and diagonally (see Fig. 1). Filling the remaining spaces in the grid are random letters that hide the words and complete the wordsearch.

Choose a theme and challenge yourself. Now you can create your own Wordsearch puzzles.

Choose a theme and type words related to the theme into the computer. The words must be entered one at a time. See the words inserted, or challenge yourself and wait until the grid is filled to search.

The Basic version, Program Listing 1, is slow. There is a long delay between the time you enter your word and when the computer inserts it into the grid. Entering the last few words of your puzzle may take as long as 10 minutes or more for each word.

You stop the computer while it is searching by pressing the slash (/) key. When you finish your program type X and the computer completes the wordsearch.

The machine-language version, Program Listing 2, written with Tandy's EDTASM + cartridge, is much faster than Basic. Aside from the speed, this version has several advantages. Choose any matrix size from 1 by 1 to 31 by 16, the words are inserted in reverse lettering, and the computer prompts you to turn on the printer.

(Ed. note: Because of the length of the Assembly version of Wordsearch, we've omitted the op codes. You must assemble this program with Radio Shack's EDTASM + . Anyone wanting the Assembly listing with op codes can send a self-addressed, stamped envelope

EUROPEAN COUNTRIES

```
E A I N A B L A V G U K A I R A G L U B
E I G Y J I J U I C E R W A L E S L Z I
N E M A N P V D N A L T O C S F Z M C H
G D Z W V T N A Q A S Z A S R R Y V J I
L X U R B A M U L U L W I W U A M X Z F
A V M O L O X F N S F P N E S N X B C M
N R D N A L L O H T O K A D S C X T U M
D Y I G E T K Z N R S G M E I E I I R I
P F L Q R V W A X I G F U N A G G R I T
W D N A L E C I X A A I R Y S L I E A C
V V X F T Z E R Z T P P L R E W H L Y W
D U I Y U I V C E D B N S B R Q Z A U D
H U N G A R Y F E D P O L A N D G N V J
N C Z E C H O S L O V A K I A B G D I C
```

ENGLAND	SPAIN	FRANCE	BELGIUM	SCOTLAND
RUSSIA	YUGOSLAVIA	RUMANIA	FINLAND	AUSTRIA
ICELAND	WALES	CZECHOSLOVAKIA	BULGARIA	HUNGARY
POLAND	IRELAND	SWEDEN	HOLLAND	ALBANIA
ITALY	NORWAY	GREECE		

Fig. 1. Sample Wordsearch Puzzle

System Requirements

16K RAM
Extended Color Basic
Editor/Assembler
Printer

to *HOT CoCo*, Pine St., Peterborough, NH 03458.)

As this program needs more than one page of display, the origin is moved from 600 to 2000. To assemble in memory, follow this 10-step procedure (these instructions include the computer prompts # and * which need not be typed):

1. Insert EDTASM+ cartridge
2. Enter *Z
3. Enter #W
4. Enter #FF/ 600 2000
5. Enter #GC006
6. Ready the cassette at the start of the source-code version.
7. Enter *L
8. Enter Ready Cassette
9. Tape reads in.
10. +*A/IM/AO

This will assemble the program to memory. It can be saved to tape using the following steps:

1. Enter *Z
2. Ready the cassette
3. Enter #P WS2 1000 3000 1000

When saved, you can reload from Basic by rewinding the tape and using CLOADM:EXEC or you can run it straight away by:

1. Enter #G 1000

Wordsearch puzzles on your CoCo are fun when you design them yourself. It's a challenge to invent a theme and choose the related words, and then to search for them. ■

Write Martin Tichborne c/o HOT CoCo, *Pine St., Peterborough, NH 03458.*

Program Listing 1. Wordsearch, Basic Version

```
10 REM WORD SEARCH PROGRAM
20 CLEAR3000
30 DIMF$(280)
40 DIM G$(80)
50 RR=1
60 CLS0
70 POKE 65494,0
80 PRINT@230,"WORD SEARCH";
90 PRINT@262,"COPYRIGHT (C)";
100 PRINT@294,"BY M.J. TICHBORNE
";
110 GOSUB1640
120 Z=0:Q=1
130 FOR I=1 TO 280
140 F$(I)="."
150 NEXT I
160 CLS:PRINT"INSTRUCTIONS";:INP
UT IN$:IF LEFT$(IN$,1)="N" THEN2
30
170 IF LEFT$(IN$,1)<>"Y" THEN 16
0
180 CLS:PRINT"THIS PROGRAM MAKES
 RANDOM WORD  SEARCHES.":PRINT"Y
OU SELECT THE WORDS AND I WILL I
NSERT THEN FOR YOU.":PRINT"WHEN
YOU ARE FINISHED TYPE 'X'  AND I
 WILL FILL THE GAPS IN"
190 PRINT:PRINT"IF I TAKE TOO LO
NG FINDING A    SPACE FOR YOUR W
ORD THEN HIT '/'AND I WILL GIVE
YOU THE CHANCE  TO TYPE A NEW WO
RD IN":PRINT
200 PRINT"H A P P Y  S E A R C H
 I N G !"
210 PRINT"HIT -ENTER- TO CONTINU
E";:INPUT G:CLS
220 CLS
230 PRINT"IF YOU DO NOT WISH TO
SEE THE   WORDS BEING INSERTED T
HEN TYPE  <D><ENTER> BUT IF YOU
DO JUST   HIT <ENTER>.";:INPUT R
$
240 POKE 65495,0:REM--SPEED IT U
P
250 J=6:CLS
260 FOR I=1 TO 280
270 PRINT@J,F$(I)
280 J=J+1
290 IF(INT(J/32)*32-J+32)<7 THEN
 J=J+12
300 NEXT I
310 IF Z>100 THEN 1260
320 PRINT@448,STRING$(31," ")
330 PRINT@449,"WORD";Q;:INPUT A$
340 IF A$<>"X"THEN G$(Q)=A$:Q=Q+
1
350 IF A$="X"THEN 1200
360 C=RND(8)
370 POKE 65494,0
380 IF INKEY$="/" THEN 1610
390 POKE 65495,0
400 Z=0
410 GOTO 450
420 REM
430 IF Z<8 THEN 1170
440 GOTO 360
450 D=LEN(A$)
460 GOTO 990
470 JJ=0:ON C GOSUB 670,700,740,
780,800,820,840,860
480 IF JJ=1 THEN 420
490 C$=A$
500 KK=0
510 FOR J=1 TO D
520 IF KK=1 THEN 560
530 IF F$(B+((J-1)*E))="." THEN
550
540 IF F$(B+((J-1)*E))<>LEFT$(C$
,1) THEN 1620
550 C$=RIGHT$(C$,D-J)
560 NEXT J
570 IF KK=1 THEN 420
580 C$=A$
590 FOR J=1 TO D
600 K=(B+((J-1)*E))
610 F$(K)=LEFT$(C$,1)
620 C$=RIGHT$(C$,D-J)
630 IF R$="D" THEN 650
640 POKE INT((K-1)/20)*12+1029+K
,ASC(F$(K))
650 NEXT J
660 GOTO 320
670 E=-20
680 GOSUB 880
690 RETURN
700 E=-19
710 GOSUB 880
720 GOSUB 950
730 RETURN
740 E=1
750 GOSUB 880
760 GOSUB 970
770 RETURN
780 E=21
790 GOTO 710
800 E=20
810 GOTO 680
820 E=19
830 GOTO 710
840 E=-1
850 GOTO 750
860 E=-21
870 GOTO 710
880 S=INT((B-1)/20)
890 T=INT(((D-1)*E+B-1)/20)
900 IF T>13 THEN 1630
910 IF T<0 THEN 1630
920 IF S>13 THEN 1630
930 IF S<0 THEN 1630
940 RETURN
950 IF ABS(S-T)<>D-1 THEN 1630
960 RETURN
970 IF ABS(S-T)<>0 THEN 1630
980 RETURN
990 H=RND(D)
1000 C$=RIGHT$(LEFT$(A$,H),1)
1010 FOR J=1 TO 50
1020 N=RND(280)
1030 IF F$(N)=C$ THEN 1060
1040 NEXT J
1050 B=N:GOTO 470
1060 ON C GOTO 1070,1100,1110,11
20,1130,1140,1150,1160
1070 P=20
1080 B=P*(H-1)+N
1090 GOTO 470
1100 P=19:GOTO 1080
1110 P=-1:GOTO 1080
1120 P=-21:GOTO 1080
1130 P=-20:GOTO 1080
1140 P=-19:GOTO 1080
1150 P=1:GOTO 1080
1160 P=21:GOTO 1080
1170 C=RND(8)
1180 Z=Z+1
1190 GOTO 1060
1200 FOR I=1 TO 280
1210 IF F$(I)<>"." THEN 1230
1220 F$(I)=CHR$(ASC("A")+RND(26)
-1)
1230 NEXT I
1240 Z=1000
1250 GOTO 250
1260 POKE 65494,0
1270 PRINT@448,STRING$(31," ")
1280 PRINT@448,"HIT ENTER WHEN P
RINTER READY";:INPUT A$
1290 PRINT#-2,CHR$(30)
1300 PRINT@448,STRING$(31," ")
1310 PRINT@448,"TYPE HEADING";:I
NPUT B$
1320 PRINT@448,"BIG(B)OR SMALL(S
) LETTERS";:INPUT A$
1330 PRINT#-2,TAB(30);B$
1340 PRINT#-2
1350 PRINT#-2
1360 IF A$="B" THEN PRINT#-2,CHR
$(31):GOTO 1390
1370 IF A$="S" THEN PRINT#-2,CHR
$(30):GOTO 1390
1380 GOTO 1320
1390 POKE 65495,0
1400 I=1
1410 IF A$="B" THEN C$=" "ELSE C
$="  "
1420 IF A$="B" THEN D$=""ELSE D$
=STRING$(13," ")
1430 FOR K=1 TO 20
1440 D$=D$+F$(I)+C$:I=I+1
1450 NEXT K
1460 POKE 65494,0
1470 PRINT#-2,D$
1480 PRINT#-2
1490 IF I<280 THEN 1410
1500 PRINT#-2,CHR$(30)
1510 IF HH=1 THEN 1660
1520 FOR I=1 TO Q STEP 5
1530 PRINT#-2,G$(I);TAB(15);G$(I
+1);TAB(30);G$(I+2);TAB(45);G$(I
+3);TAB(60);G$(I+4)
1540 PRINT#-2,X$
1550 NEXT I
1560 PRINT#-2,CHR$(30)
1570 GOSUB 1640
1580 PRINT@448,"NEXT COPY(C):NEW
 SEARCH(N)";
1590 INPUT A$:IF A$="C" THEN 127
0 ELSE IF A$="N" THEN RUN
1600 END
1610 Q=Q-1:G$(Q)="":GOTO 320
1620 KK=1:GOTO 560
1630 JJ=1:RETURN
1640 PLAY"V31;T6;L8;E;L8;C#;L4;O
2;A;A;L8;A;B;O3;L8;C#;D;L4;E;E;E
;C#;"
1650 RETURN
1660 FOR I=1 TO Q-1 STEP 5
1670 GOTO 1530
```

END

Program Listing 2. Wordsearch, Assembly Version

```
00100  ORG $1000
00110 WRDSER JMP START
00120 *
00130 *
00140 *GENERATES A RANDOM IN D
00150 *
00160 RAND LDA A1
00170  ADDA #7
00180  LDB B1
00190  MUL
00200  STA A1
00210  ADDB A1
00220  SUBB D1
00230  STB A1
00240  LDA D1
00250  ADDA #$1
00260  STA D1
00270  LDB B1
00280  MUL
00290  ADDB #$A7
00300  ADDB C1
00310  STB B1
00320  LDA C1
00330  ADDA A1
00340  STA C1
00350  LDA B1
00360  ANDA #1
00370  LDB A1
00380  RTS
00390 A1 FDB $00
00400 B1 FDB $00
00410 C1 FDB $00
00420 D1 FDB $00
00430 *
00440 PAGE2 EQU $600 (600)
00450 PAGE2E EQU $800 (800)
00460 CHANGE EQU $FFC7 (FFC7)
00470 SIZE EQU 512 (512)
00480 *
00490 *
00500 *CLEAR VIDEO PAGE
00510 *
00520 CLEAR1 LDY #$400
00530  LDA #$20
00540 CLR1 STA ,Y+ CLEAR PAGE
00550  CMPY #$600
00560  BNE CLR1
00570  LDY #$400
00580  RTS
00590 *
00600 *
00610 START JSR INTRO INTRODUCTION TEXT
00620  LDA #$FE
00630  STA $6F
00640 ST1 JSR INPUT INPUT PARAMETERS
00650  CLRA
00660  STA LIST INIT LIST
00670  JSR MATRIX DRAW MATRIX
00675  JSR POLKEY
00680 ST2 JSR WORD INPUT WORD
00690  JSR LOCATE LOCATE WORD IN MATRIX
00700  BRA ST2
00710 *
00720 *
00730 *INTRODUCTION TEXT
00740 *
00741 INTRO JSR CLEAR1
00742  LDX TXTCP
00744  LDY #$400
00746  JSR STRING
00748  JSR POLKEY
00750  LDX TEXT1 X=SOURCE TEXT ADD.
00760  LDY #$400 Y=DEST SCREEN ADD.
00770  JSR STRING O/PTO SCREEN
00775  JSR POLKEY
00780  RTS
00790 TEXT1 FDB TEXT10
00800 *
00810 *
00820 *STRING S/R
00830 *
00840 STRING LDA ,X+
00850  CMPA #$2A
00860  BEQ STR1 END OF STRING = *
00870  CMPA #$41
00880  BLT STR0 <A
00890  CMPA #$5A
00900  BGT STR0 >Z
00910  SUBA #$40
00920 STR0 STA ,Y+
00930  CMPY #$600
00940  BLT STRING
00950  LDY #$400
00960 STR1 RTS
00970 *
00980 *
00990 *INPUT OF VARIABLES
01000 *
01010 INPUT LDX TEXT2
01012  JSR CLEAR1
01014  LDY #$400
01020  JSR STRING HIT ENTER TEXT
01030 IN1 JSR POLKEY POLE KEYBOARD
01040  CMPA #$4E =N
01050  BNE IN2A
01052  CLRA
01054 IN2A STA SWT
01060 IN2 JSR CLEAR1 CLEAR PAGE 1
01070  LDY #$460
01080  LDX TEXT3
01090  JSR STRING PRINT DEFINE COLUMNS
01100  JSR NEWLIN
01110  JSR CLRLIN
01120  JSR NUMBIN READ IN NUMBER
01130  LDD NUMB
01140  CMPD #0
01150  BLS IN2 NOT ZERO
01160  CMPD #31
01170  BGT IN2 GREATER THAN 31
01180  STD COL COLUMN #
01190  JSR NEWLINE GO TO NEW LINE
01200 IN3 LDX TEXT4
01210  JSR STRING PRINT DEFINE ROWS
01220  JSR NEWLIN
01230  JSR CLRLIN
01240  JSR NUMBIN READ IN NUMBER
01250  LDD NUMB
01260  CMPD #0
01270  BLS IN4 NOT 0
01280  CMPD #16
01290  BLS IN5 LESS THAN 16
01300 IN4 JSR CLRLIN
01310  BRA IN3
01320 IN5 STD ROW
01330  RTS
01340 POLCAT EQU $A000
01350 *
01360 *
01370 *
01372 TXTCP FDB TXTCP1
01374 TXTCP1 FCC /         COPYRIGHT 1983                  FRANK TICHBORNE
   /  '
01375  FCC /                                /
01376  FCC        /   BRISTOL BS14OBX                     U.K. /
01378  FCC            /                                  0272 832904*/
01380 TEXT10 FCC /         WORDSEARCH              THIS PROGRAM WILL CONSTRUCT
   A RANDOM MATRIX OF YOUR WORDS./
01390  FCC       /  YOU TYPE THE WORDS IN, AND       I WILL PUT THEM INTO THE
   PUZZLE IN RANDOM DIRECTIONS/
01400  FCC   /       AND POSITIONS. WHEN YOU HAVE      COMPLETED YOUR ENTRIES THEN
   TYPE "X" AND I WILL PRINT/
01410  FCC /         OUT THE COMPLETE WORDSEARCH       INCLUDING RANDOM LETTERS FOR
   THE BLANKS. IF I TAKE TOO /
01420  FCC  ?        LONG TRYING TO FIND A PLACE       THEN YOU CAN ABANDON THAT
   WORD BY TYPING "/" .  <MORE>    *?
01430 COL FDB $0000
01440 ROW FDB $0000
01450 NUMB FDB $0000
01460 TEXT2 FDB TEXT20
01470 TEXT3 FDB TEXT30
01480 TEXT4 FDB TEXT40
01490 TEXT20 FCC / IF YOU DON'T WANT TO SEE YOUR   WORDSEARCH GENERATED THEN
 /
01492  FCC        / HIT <N> OTHERWISE HIT ANY KEY*/
01500 TEXT30 FCC /HOW MANY COLUMNS DO YOU WANT ?    1 TO 31*/
01510 TEXT40 FCC /HOW MANY ROWS DO YOU WANT ?       1 TO 16*/
01520 *INPUT NUMBER
01530 *
01540 NUMBIN LDD #0
01550  STD NUMB NUMBER=0
01560 NM1 JSR POLKEY
01570  CMPA #$0D
01580  BEQ NM3 =ENTER
01590  STA ,Y+
01600  CMPY #$600
01610  BNE NM4
01620  LDY #$400
01630 NM4 CMPA #$30
01640  BLT NM2 <0
01650  CMPA #$39
01660  BGT NM2 >9
01670  SUBA #$30 MAKE NUMBER
01680  STA TEMP
01690  BSR M10 *10
01700  ADDB TEMP
01710  ADCA #$0
01720  STD NUMB
01730  BRA NM1 NEXT NUMBER
01740 M10 LDA NUMB
01750  LDB #$0A =10
01760  MUL  A*10
01770  STB NUMB
01780  LDB #$0A
01790  LDA NUMB+1
01800  MUL
01810  STB NUMB+1
01820  ADDA NUMB
01830  STA NUMB
01840 NM3 RTS
01850 NM2 LDX TEXT5
01860  JSR NEWLIN
01870  JSR STRING
01880  JSR NEWLIN
01890  BRA NUMBIN
01900 TEXT5 FDB TEXT50
01910 TEXT50 FCC /NOT A NUMBER! TRY AGAIN.*/
01920 *
01930 *
01940 *NEW LINE
```

Listing 2 continued

Listing 2 continued

```
01950 *
01960 NEWLIN STY TEMP
01970  STD TEMP1
01980  LDD TEMP
01990  ADDD #$0020
02000  CMPD #$600
02010  BLT NW1
02020  LDD #$400 RESET PAGE
02030 NW1 ANDB #$E0
02040  STD TEMP
02050  LDY TEMP
02060  LDD TEMP1
02070  RTS
02080 TEMP FDB $0000
02090 *
02100 *
02110 BACKLI STY TEMP
02120  STD TEMP1
02130  LDD TEMP
02140  SUBD #$20
02150  CMPD #$400
02160  BGE NW1
02170  LDD #$5E0
02180  BRA NW1
02190 TEMP1 FDB $0000
02200 *
02210 *
02220 *
02230 *
02240 *CLEAR CURRENT LINE
02250 *
02260 CLRLIN JSR NEWLIN
02270  JSR BACKLI
02280  STD TEMP
02290  LDA #0
02300  LDB #$20
02310 CLN1 STB ,Y+
02320  ADDA #01
02330  CMPA #$20
02340  BNE CLN1
02350  LDD TEMP
02360  JSR BACKLI
02370  RTS
02380 *
02390 *
02400 *MATRIX 600-800
02410 *
02420 MATRIX STY TEMP
02430  STA CHANGE SET TO 2ND PAGE
02440  LDD BLIST
02450  STD ALIST INITIALISE OUTPUT LIST
02460  LDA #$20
02470  SUBA COL+1
02480  LSRA
02490  STA COL SAVE START COL.
02500  LDA #$10
02510  SUBA ROW+1
02520  LSRA
02530  STA ROW SAVE START ROW
02540  LDY #PAGE2
02550  LDA #$20
02560 MT1 STA ,Y+ CLEAR PAGE 2
02570  CMPY #PAGE2E
02580  BLT MT1
02590  LDB ROW
02600  MUL  32*ROW DIF.
02610  ADDB COL
02620  ADDD #PAGE2 START OF MATRIX
02630  EXG Y,D
02640  LDA ROW+1
02650  STA TEMP1
02660 MT3 LDA COL+1
02670  STA TEMP1+1
02680  LDB #$2E PUT DOTS IN MATRIX
02690 MT4 STB ,Y+
02700  DEC TEMP1+1
02710  BNE MT4 END OF ROW?
02720  LDD #$20
02730  SUBB COL+1
02740  STD TEMP2 NEXT ROW ADDRESS
02750  EXG Y,D
02760  ADDD TEMP2
02770  EXG Y,D
02780  DEC TEMP1
02790  BNE MT3 LAST ROW?
02820 MT2 LDY TEMP
02830  RTS
02840 TEMP2 FDB $0000
02850 *
02860 *
02870 *INPUT THE WORD
02880 *
02890 WORD LDX TEXT6
02900  LDD #$5E9
02910  STD TEMP1
02920 WORD1 JSR CLEAR1 CLEAR PAGE
02940  STA $FFC6 SET TO PAGE 1
02950  LDY #$5E0
02960  JSR STRING
02970 WD0 LDA #$AF
02980  STA ,Y
02990 WD1 JSR POLKEY
03000  CMPA #$0D
03010  BEQ WD2 ENTER
03020  CMPA #$15
03030  BEQ WD5 LEFT ARROW
```

Listing 2 continued

✓See List of Advertisers on page 130

Listing 2 continued

```
03040  CMPA #$08
03050  BNE WD3 DITTO
03060 WD5 LDA #$20
03070  STA ,Y
03080 WD4 EXG D,Y
03090  SUBD #01
03100  CMPD TEMP1
03110  BLE WD8
03120 WD4A EXG D,Y
03130  BRA WD0 CURSOR BACK
03140 WD3 STA ,Y+
03150  CMPY #$600
03160  BNE WD0
03170  BRA WD4
03172 WD2 LDA SWT
03174  BEQ WD2A
03180  STA CHANGE SET TO P 2
03190 WD2A CMPY #$5EB
03200  BNE WD6
03210  LDA $5EA
03220  CMPA #$58 =X
03230  LBEQ RANDOM
03240 WD6 RTS
03250 WD8 ADDD #1
03260  BRA WD4A
03270 *
03280 *
03290 *
03295 SWT FDB $0000
03300 TEXT6 FDB TEXT60
03310 TEXT60 FCC /TYPE WORD */
03320 *
03330 *
03340 *POLES FOR KEY AND UPDATES RANDOM WORD
03360 POLKEY JSR [POLCAT]
03370  BEQ PK1
03371  *
03380  RTS
03382 PK1 JSR RAND
03402  BRA POLKEY
03410 *
03420 *
03430 *CHECKS IF POSITION IS IN BOX, ENTRY D=0 TO 512
03440 *EXIT Y=BOX ADDRESS, Z=1 IF IN BOX
03450 *
03460 BOXCHK CMPD #0
03470  BLT BX0
03480  CMPD #SIZE (NORMALLY=512)
03490  BLT BX1
03500 BX0 JSR RAND
03510  LDA #0
03520 BX2 RTS  EXIT, NOT IN BOX.
03530 BX1 ADDD #PAGE2
03540  EXG D,Y
03550  LDA ,Y
03560  CMPA #$20 IS IT BLANK?
03570  BEQ BX2 YES
03580  RTS EXIT, IN BOX.
03600 *
03610 *LOCATE WORD INTO MATRIX
03620 *
03630 LOCATE LDA $5EA
03640  CMPA #$AF =CURSOR
03650  LBEQ ST2
03660  JSR RAND
03670  CLRA
03680  ANDB #$1F
03690  ADDD #$5EA
03700  CMPD #$600
03710  BGE LOCATE PAST END OF PAGE
03720  EXG D,X
03730  LDA ,X
03740  CMPA #$20
03750  BEQ LOCATE
03760  CMPA #$AF
03770  BEQ LOCATE =CURSOR
03780 *RAND LETTER IN A
03790 *ITS ADD. IN X
03800  LDB #$FF
03820  STB X2 STORE LOOP COUNT
03830 LC4 JSR RAND
03840  JSR BOXCHK
03850  BEQ LC4 NOT IN BOX?
03860  LDA ,Y
03870  CMPA ,X IS THIS LETTER IN BOX?
03880  BEQ LC5 FOUND ONE
03890  DEC X2
03900  BNE LC4 NOT END OF LOOP
03910 LC4B LDA ,Y
03920  CMPA #$2E DOT?
03930  BEQ LC5 OK
03940 LC4A JSR [POLCAT]
03950  NOP
03960  CMPA #$2F /
03977770  LBEQ ST2
03980  CMPA #$0D
03990  BEQ LOCATE
04000 LC4X JSR RAND PICK ANOTHER
04010  JSR BOXCHK
04020  BEQ LC4A NOT IN BOX
04030  BRA LC4B CHECK IT
04040 *
04050 *NOW CHOOSE A DIRECTION
04060 LC5 LDA #$8
04070  STA X2 LOOP COUNT=8
04080  STX TEMP
04090  STY TEMP1
```

Listing 2 continued

WANTED!

Young men and women seeking adventure, excitement and thrill-a-minute action. No experience necessary—just you and your Color Computer. See below:

FOR THE 32K THRILLSEEKER

ZAXXON, Disk or Cass.	DataSoft	$29.95
PROTECTOR II, Cass.	Synapse	$29.95
DESERT PATROL, Cass.	Arcade Anim.	$21.95
ICEMASTER, Cass.	Arcade Anim.	$21.95
FOODWAR, Cass.	Arcade Anim.	$22.95
WACKY FOOD, Cass.	Arcade Anim.	$19.95
CASHMAN, Cass.	Comp. Shack	$24.95
CHOPPER STRIKE, Cass.	Comp. Shack	$24.95

LOTS OF PLAY FOR 16K

MOONSHUTTLE, Cass.	DataSoft	$26.95
SHAMUS, Cass.	Synapse	$29.95
FROG TREK, Cass.	Oelrich	$14.95
3-D TIC-TAC-TOE, Cass.	Oelrich	$16.95

Interested applicants send check or m/o to:

OELRICH PUBLICATIONS
4040 N. NASHVILLE
CHICAGO, IL 60634

Credit card orders, call: 800-621-0105
(In Illinois: 312-545-9286)

✓288

CP/M

FOR YOUR COLOR COMPUTER

- Now have access to the largest library of programs available
- CP/M is the recognized leader in professional and personal software
- Adds the power of a 4 MHz, Z80A
- CP/M 2.2 included
- No modifications are necessary. Simply plug into the cartridge port and plug Radio Shack's disk controller into it.
- Maintains full Radio Shack compatibility
- Requires 64K memory and one or more disks with controller

$250.00

WAYNE TECHNOLOGY

P.O. BOX 5196 • ANAHEIM, CA 92804-1196
(714) 772-5757

✓170

Radio Shack™ Tandy / Radio Shack Corp. / CPM™ Digital Research

YORK 10™ CASSETTES

COMPUTER GRADE DATA TRAC BLANK CASSETTES

C-05, C-06, C-10, C-12, C-20, C-24, C-32

From the leading supplier of Computer Cassettes, new, longer length C-12's (6 minutes per side) provide the extra few feet needed for some 16K programs.

- BASF-LHD (DPS) world standard tape. Premium 5 screw shell with leader.
- Internationally acclaimed. Thousands of repeat users.
- Error Free • Money back guarantee.

✓156

500 C-12's or C-10's — 38¢ each
(w/labels, add 4¢ • Shipping $17./500
500 Boxes 13¢ ea. • Shipping $10./500
(does not apply to 500 quantity offer)

TRACTOR FEED • DIE-CUT BLANK CASSETTE LABELS
WHITE: $3.00/100 $20.00/1000

NEW **COLORED LABELS** • Pastels - Red, Blue, Green, Yellow, Lavender
$4.00/100 $30.00/1000

CASSETTE STORAGE CADDY
Holds 12 cassettes w/o boxes
Includes edge labels and index card
$2.95

FREE
1 CADDY WITH EVERY 4 DOZ. CASSETTES PURCHASED

VISA **Call: 213/700-0330** for IMMEDIATE SHIPMENT on Credit Card Orders MasterCard

ORDER NOW . . . MAIL TO . . . **YORK 10™** 9525 Vassar Ave. #HC Chatsworth, CA 91311

ORDER FORM

#HC

ITEM	1 DOZEN	2 DOZEN	TOTAL
C-05	☐ 7.00	☐ 13.00	
C-06	☐ 7.00	☐ 13.00	
C-10	☐ 7.50	☐ 14.00	
C-12	☐ 7.50	☐ 14.00	
C-20	☐ 8.75	☐ 16.50	
C-24	☐ 9.00	☐ 17.00	
C-32	☐ 11.00	☐ 21.00	
Hard Box	☐ 2.50	☐ 4.00	
White Labels	☐ 3.00/100	☐ 20.00/1000	
Colored Labels Color	☐ 4.00/100	☐ 30.00/1000	
DESCRIPTION	PRICE	QUANTITY	
Storage Caddy	2.95		
		SUB TOTAL	
Calif. residents add sales tax			
Shipping/handling (any quantity — using prices above)			3.50
Outside 48 Continental States — Additional $1 per caddy; per doz. cassettes or boxes.			
		TOTAL	

Each cassette includes two YORK 10 labels only. Boxes are sold separately. We prefer to ship by UPS as being the fastest and safest. If you need shipment by Parcel Post, check here ☐

NOTE: Additional charges outside 48 Continental States. Shipments to AK, HI, and USA possessions go by Priority Mail. Canada & Mexico—Airmail. All others— Sea Mail.

Ask about our DUPLICATING SERVICE

CHECK OR M.O. ENCLOSED ☐ Charge to Credit Card: VISA ☐ MASTERCARD ☐
☐ PLEASE SEND QUANTITY DISCOUNTS

Card No. _____ Exp. _____

Name _____

Address _____

City _____ State/Zip _____

Signature _____ Phone _____

Computer make & model _____ Disk? (y/n) ____

✓See List of Advertisers on page 130

Listing 2 continued

```
04100 LC5D JSR RAND
04110  ANDB #$07 8 DIRECTIONS!
04120  CMPB #3
04130  BLT LC6 B=0,1,2 (-33,-32,-31)
04140  BNE LC5A
04150  LDB #34 B=3 (+1)
04160  BRA LC6
04170 LC5A CMPB #7
04180  BNE LC5B
04190  LDB #32 B=7 (-1)
04200  BRA LC6
04210 LC5B ADDB #60 B=4,5,6 (31,32,33)
04220 *
04230 LC6 CLRA
04240  SUBD #33
04250  STD X3 SAVE OFFSET
04260 LC6B LDA ,-X
04270  CMPA #$20
04280  BEQ LC7 REACHED BEGINING
04290  EXG D,Y
04300  SUBD X3 ADD OFFSET
04310  SUBD #PAGE2
04320  JSR BOXCHK
04330  BEQ LC6A NOT IN BOX
04340  LDA ,Y GET CHAR
04350  CMPA #$2E DOT?
04360  BEQ LC6B OK
04370  CMPA ,X
04380  BEQ LC6B OK
04390 LC6A LDX TEMP
04400  LDY TEMP1
04410  DEC X2
04420  BNE LC5D RESET & CHANGE DIR. '
04430  BRA LC4A CHOOSE ANOTHER LETTER
04440 *
04450 *
04460 LC7 EXG D,Y
04470  SUBD X3
04480  EXG D,Y
04490  LDA ,X+
04500 LC7A LDA ,X+
04510  STA X4
04520  CMPA #$AF
04530  BEQ LC8 END FOUND (CURSOR)
04540  EXG D,Y
04550  ADDD X3 ADD OFFSET
04560  SUBD #PAGE2
04570  JSR BOXCHK
04580  BEQ LC6A NOT IN BOX
04590  LDA ,Y GET CHAR
04600  CMPA #$2E DOT?
04610  BEQ LC7A OK
04620  CMPA X4 CHARS. EQUAL?
04630  BEQ LC7A OK
04640  BRA LC6A NOT OK, CHOOSE ANOTHER DIR.
04650 X3 FDB $0000
04660 *
04670 *
04680 LC8 LDA ,-X
04690  CLRA
04700  STA X2
04710 LC8A LDA ,-X
04720  CMPA #$20
04730  BEQ LC9 FINISHED
04740  STY [ALIST]
04750  INC X2
04760  INC X2
04765  SUBA #$40
04770  STA ,Y
04780  EXG D,Y
04790  SUBD X3
04800  EXG D,Y
04810  BRA LC8A STORE CHARS IN MATRIX
04812 LC9 LDA SWT
04814  BEQ LC9A
04816  JSR POLKEY
04818 LC9A LDA ,X+
04820 LC9C EXG D,Y
04822  ADDD X3
04824  EXG D,Y
04826  LDA ,X+
04828  CMPA #$AF CURSOR?
04830  BEQ LC9B
04832  STA ,Y
04834  BRA LC9C
04836 *
04840 LC9B LDY ALIST
04850  LDA ,Y
04860  ANDA #1
04870  ADDA X2
04880  STA ,Y++
04890  LDA X3+1
04900  STA ,Y+
04910  STY ALIST
04920  CLRA
04930  STA ,Y
04940  CMPY CLIST
04950  BGE RANDOM
04960  RTS
04970 X2 FDB $00
04980 X4 FDB $00
04990 RANDOM LDY #PAGE2
05000  STA CHANGE CHANGE TO PAGE 2
05010 RN0 LDA ,Y+
05020  CMPA #$2E =DOT?
05030  BNE RN2 NO
05040 RN1 JSR RAND
05050  ANDB #$1F
05060  CMPB #25 A TO Z
05070  BGT RN1 NO
05080  ADDB #$41 MAKE A TO Z
05090  STB ,-Y
05100  LDB ,Y+
05110 RN2 CMPY #PAGE2E
05120  BNE RN0 NOT DONE
05122  JSR POLKEY
05124  STA $FFC6
05130  JMP PRINT
05140 *
05150 *
05160 *
05170 BLIST FDB LIST
05180 ALIST FDB LIST (CHANGES)
05190 CLIST FDB LIST+900
05200 LIST RMB 901
05210 *
05220 *
05230 *PRINT ROUTINE
05240 *
05250 PRINT LDY #$5E0
05260  JSR CLRLIN
05270  LDX TEXT8
05280  JSR STRING
05290 PR1 JSR POLKEY
05300  CMPA #$0D
05310  BEQ PR1A
05320  STA ,Y
05330  BRA PR1
05340 PR1A LDA ,Y
05350  CMPA #$50
05360  BEQ PR2
05370  CMPA #$44
05380  LBEQ ST1
05390  CMPA #$45
05400  BNE PR1
05410  SWI
05450 PR2 JSR CLRLIN
05451  LDX TXT1A
05453  JSR STRING PRINTER READY?
05455  LDA #10
05457  JSR [CHROUT]
05459  JSR CLRLIN
05460  JSR TITLE
05465  STA $FFC6 P1
05470 PR2C JSR CLRLIN
05480  LDX TEXT9
05490  JSR STRING
05500 PR2A JSR POLKEY
05510  CMPA #$0D ENTER
05520  BEQ PR2B
05530  STA ,Y
05540  BRA PR2A
05550 PR2B LDA ,Y
05560  CMPA #$4C
05570  BNE PR4
05580  LDA #4
05590  STA X2 # OF SPACES
05600  LDA #31
05610  BRA PR5
05620 PR4 CMPA #$53
05630  BNE PR2C
05640  LDA #8
05650  STA X2 # OF SPACES
05660  LDA #30
05740 PR5 JSR [CHROUT]
05750  JSR CLRLIN
05755  STA CHANGE SET P2
05760  JSR PRIMAT
05770 *
05780 *
05790 *PRINT LIST
05800 *
05810  LDX BLIST ADD OF LIST
05820 PNT0 LDA #13 CR
05830  JSR [CHROUT]
05840  LDA #80
05850  STA TEMP1+1 COL COUNT
05860 PNT1 LDA #16
05870  STA TEMP1
05880  LDB ,X
05890  BEQ PNTL1 FINISH
05900  LDA ,X
05910  ASRA 1
05920  ANDA #$1F
05930  STA TEMP+1 STORE # OF LETTERS
05940  LDD ,X++
05950  ANDA #1
05960  ADDD #PAGE2
05970  EXG D,Y SOURCE ADD.
05980  LDB ,X+
05990  SEX
06000  STD TEMP2
06010 PNT2 LDA ,Y GET CHAR
06020  JSR [CHROUT]
06030  DEC TEMP+1
06040  BEQ PNT3 END OF WORD
06050  EXG Y,D
06060  ADDD TEMP2 ADD DIR.
06070  EXG Y,D
06080  DEC TEMP1+1
06090  BEQ PNT0 END OF LINE
06100  DEC TEMP1
```

Listing 2 continued

Listing 2 continued

```
06110  BEQ PNT1 END OF 16 CHARS
06120  BRA PNT2 MIDDLE OF WORD
06130 PNT3 DEC TEMP1+1
06140  BEQ PNT0
06150  DEC TEMP1
06160  BEQ PNT1
06170  LDA #$20
06180  JSR [CHROUT]
06190  BRA PNT3
06200 PNTL1 LDA #13
06210  JSR [CHROUT]
06220  LDX TXTC
06230  LDY #$5E0
06235  STA $FFC6 SET P1
06240  JSR STRING MORE?
06250 MR1 JSR POLKEY
06260  CMPA #$0D
06270  BEQ MR2
06280  STA ,Y
06290  BRA MR1
06300 MR2 LDA ,Y
06310  CMPA #$52
06320  LBEQ ST1
06330  CMPA #$41
06340  LBEQ PR2
06350  CMPA #$45
06360  BNE MR1
06370  SWI
06380 *
06390 *
06400 * TITLE
06410 TITLE LDX TXT1B
06420  LDD #$5E6
06430  STD TEMP1
06440  JSR WORD1
06450  LDA #31
06460  JSR [CHROUT] LARGE
06470  EXG Y,D
06480  SUBD #$5E7
06490  STB X3
06500  LDB #40
06510  SUBB X3
06520  LSRB
06530  LDY #$5E7
06540 TT0 LDA #13
06550 TT3 JSR [CHROUT]
06560  LDA #$20
06570  DECB
06580  BNE TT3
06590 TT4 LDA ,Y+
06600  CMPA #$AF
06610  BEQ TT1
06620  JSR [CHROUT]
06630  BRA TT4
06640 TT1 LDA #13
06650  JSR [CHROUT]
06660  RTS
06670 *
06680 TXT1B FDB TXT1B0
06690 TXT1B0 FCC /TITLE? */
06700 *
06710 *
06720 *PRINT MATRIX
06730 PRIMAT LDX #PAGE2
06740  LDD #$0A02
06750 PMT1 JSR [CHROUT] O/P CR*N
06760  DECB
06770  BNE PMT1
06780 PMT2 LDA #10
06790  JSR [CHROUT]
06800  LDB X2
06810  STB TEMP # OF SPACES
06820  ANDB #8
06830  BEQ PMT3A
06840  LDA #10
06850  JSR [CHROUT]
06860 PMT3A LDA #32
06870  STA TEMP+1
06880  LDA #$20
06890 PMT3 JSR [CHROUT]
06900  DEC TEMP
06910  BNE PMT3
06920 PMT4 LDA ,X+
06930  JSR [CHROUT]
06940  LDA X2
06950  ANDA #8
06960  BEQ PMT5
06970  LDA #$20
06980  JSR [CHROUT]
06990 PMT5 DEC TEMP+1
07000  BNE PMT4
07010  CMPX #PAGE2E
07020  BLT PMT2
07030  LDA #30
07040  JSR [CHROUT] SMALL CHARS
07050  LDA #10
07060  JSR [CHROUT]
07070  RTS
07080 TEXT8 FDB TEXT80
07090 TEXT80 FCC /DELETE <D>, END <E>, PRINT <P>.*/
07100 TEXT9 FDB TEXT90
07110 TEXT90 FCC /CHARACTERS LARGE <L>, SMALL <S>*/
07120 CHROUT EQU $A002
07130 TXT1A FDB TXT1AA
07140 TXT1AA FCC /TURN PRINTER ON*/
07150 TXTC FDB TXTC1
07160 TXTC1 FCC /ANOTHER<A> RESTART<R> END<E> */     END
```

↙See List of Advertisers on page 130

BY W.H. BARBER

PUT PIZAZZ IN YOUR TEXT DISPLAYS

Put an end to boring and static text messages with this program that adds sound, scrolling and more.

Do you have problems presenting text messages? Here are three techniques to change them. Almost any program can use them in some form. When written as subroutines, they require very little memory beyond that required for more conventional PRINT statements. However, the action they provide is worth the few bytes of memory.

Program Listing 1 demonstrates all three techniques for presenting text messages. The first scrolls each line of the message from the right border onto the screen. The next prints the message, letter by letter, across the screen. The third scrolls lines onto the bottom of the screen. It then shifts the entire video display up one line to make room for another line scrolled onto the now vacant screen bottom.

All three techniques use a Sound routine to enhance the effect. It provides a distinctive teletype sound every time a new letter appears on the screen. The sound also provides a slight delay to improve the viewing of each technique.

The program can be divided into five distinct sections. One occupies lines 400–420, and is the DATA statement section containing text messages to be printed. Another section, lines 10–65, controls the demonstration. The three remaining sections do the actual printing. Two of them are subroutines numbered in the 100s and 200s. The last section is numbered in the 300s.

The first two techniques are called from the control section as subroutines by line 30. Prior to calling these subroutines, line 25 establishes a new value for K that is used in the subroutines to control the print location.

The value for K can be any whole number from 0 to 511. This value represents a unique screen location starting at zero in the upper-leftmost position of the screen. They increase in value as one moves across the screen. Each screen row has 32 of these locations. There are 16 rows on the screen, hence 512 total screen locations.

Type and run the Listing below for a better understanding of this numbering system. The short program moves a black rectangle across the screen. As it moves, it displays the numeric value of the screen location occupied. To slow the sequence, increase the duration of the sound call in line 60.

```
10 FOR N=0 TO 510
20 CLS
30 PRINT @ N, CHR$(128);
40 IF N < 320 THEN PRINT @ 320 ELSE
   PRINT @ 32
50 PRINT "SCREEN LOC = = = >" N;
60 SOUND 1,3
70 NEXT
80 END
```

The subroutines used to print the text only print one line each time they are called. The control section of the program determines the number of lines printed. Lines 20 and 35 accomplish this. By changing the value of K in line 20, the start location can be changed and the ending location is controlled by line 35. The number of lines printed is the difference between the value of K in line 20 and line 35.

Before looking at the two subroutines in detail, look at lines 400–420. Both subroutines start by reading a line of text from these statements. When it reads a DATA statement, the Color Computer ignores any blank spaces following a comma (or the keyword DATA). It continues to ignore spaces until it encounters a nonblank space. This character is considered the first character of the next data element.

To center text on the screen, take care in writing a DATA statement. Positioning cannot be done with blank spaces, but other characters suffice. For example, examine the first entry in DATA statement 410. The period immediately following the word DATA functions as a positioning character. Remove it and the word this is printed flush with the left screen border rather than indented three spaces as intended.

Both subroutines use the LEFT$ function to select the proper character(s) to print. In the first technique, line 120 is used to print a substring of N$ that is 31–N characters long. Since the value of N starts at 31, and is reduced by one each time line 110 executes, the substring appears to grow in length by one

System Requirements

4K RAM
Color or Micro Color (MC-10) Basic

```
 9 '***LOADER PROGRAM TO DEMO***
10 F=1
15 CLS
20 K=32
25 K=K+32
30 ON F GOSUB100,200
35 IF K<448 GOTO 25
40 RESTORE
45 F=F+1
50 IF F=2 GOTO 15
55 RESTORE
60 CLS
65 GOTO300
70 '
98 '*************************
99 '***SCROLL IN FROM LEFT*****
100 READ N$
105 IF N$="" THEN RETURN
110 FOR N=31 TO 0 STEP -1
120 PRINT@K+N,LEFT$(N$,31-N);
130 SOUND198,1
140 NEXT
150 RETURN
155 '*************************
160 '
198 '*************************
199 '**ONE CHARACTER AT A TIME*
200 READ N$
210 FOR N=1 TO LEN(N$)
220 PRINT@K,LEFT$(N$,N)
230 SOUND198,1
240 NEXT
250 RETURN
255 '*************************
260 '
298 '*************************
299 '**SCROLL IN AND THEN UP***
300 K=480
310 GOSUB100
320 PRINT@510
330 B=B+1
340 IF B>12 THEN END
350 GOTO 310
355 '*************************
360 '
398 '*************************
399 '**SENTENCES FOR PRINTING**
400 DATA.........news flash.,.,
410 DATA.  THIS IS A COLOR COMPU
TER ,.DEMONSTRATING THREE WAYS T
O ,.DISPLAY DATA ON THE TV SCREE
N. ,
420 DATA.  TRY YOUR OWN EXPERIME
NTS ,.WITH THESE THREE SUBROUTIN
ES. , ,.  I HOPE YOU LIKE THE EF
FECT! ,., ,
```

Program Listing. Demonstration Techniques

letter each time through the loop.

The print location is also changed during each loop by the equation K + N in line 120. The combination of these two actions produces the scrolling effect of the first technique.

The second subroutine also uses a substring of N$, but this time the substring is always one letter long. The character in the substring is determined by the LEFT$ function in line 220, and is the character that is N characters from the leftmost character of N$. As N increases each time through the loop, the sentence appears to print on the screen, one letter at a time, left to right.

The third display uses the same scrolling system as the first technique and is seen in the subroutine call of line 310. Prior to calling the subroutine, line 300 causes printing to occur on the bottom of the screen (row beginning with screen location 480).

After the sentence has completely scrolled onto the screen, line 320 is executed, printing to the lower right screen location (i.e., 510), and causing a vertical scrolling of the entire video screen. This action occurs as a result of Color Basic's automatic screen-scroll function, which is very fast. The result is a clearing of the bottom screen row, that is now ready for another line of text.

The third technique is not called as a subroutine, so it must provide its own control. This is done by lines 330 and 340 that are used to limit printing to 12 lines of text. This is sufficient for the example presented, but by changing the values that are accepted, you can select a different number of lines. ■

Address correspondence to W.H. Barber, 978 Cherry St., Winnetka, IL 60093.

✓See List of Advertisers on page 130

BY MICHAEL V. KRAFT

CoCo World Control

Here's a low-cost, versatile design that interfaces your Color Computer to the world around you. It allows for a cartridge port, buffers the address and data buses, and decodes 16+ addresses for select signals. This was designed originally to interface a scratch-built PROM-burner, thus the layout of S1 and S2, but you can use it for many purposes.

The circuit is connected to the computer via the cartridge port using 40-wire ribbon cable and connectors available from Radio Shack. I used aluminum duct tape on the ribbon cable to cut down on the RFI. The board is a prototype board available from Radio Shack, and the sockets are all wire-wrap.

The numbers on the left of the schematic are connector numbers from the computer. All lines marked Vcc are connected to the +5V power supply needed for the interface. The line marked +5 (9) comes from the computer and is used only to power the ROM pack. All grounds are tied together.

Check the tips in this hardware feature for a Color Computer interface with a multitude of uses.

Address lines 0–15 are buffered by Z1, Z2, and Z3. These are 74LS367 tri-state buffers with their enable pins (1 and 15) tied to ground so they are always enabled. AB0–AB13 are wired to the ROM pack socket as shown. AB0–AB3 also go to Z9, a 74154, one of 16 decoders. Z3 also buffers the read/write signal and the spare device-select signal (more on this later).

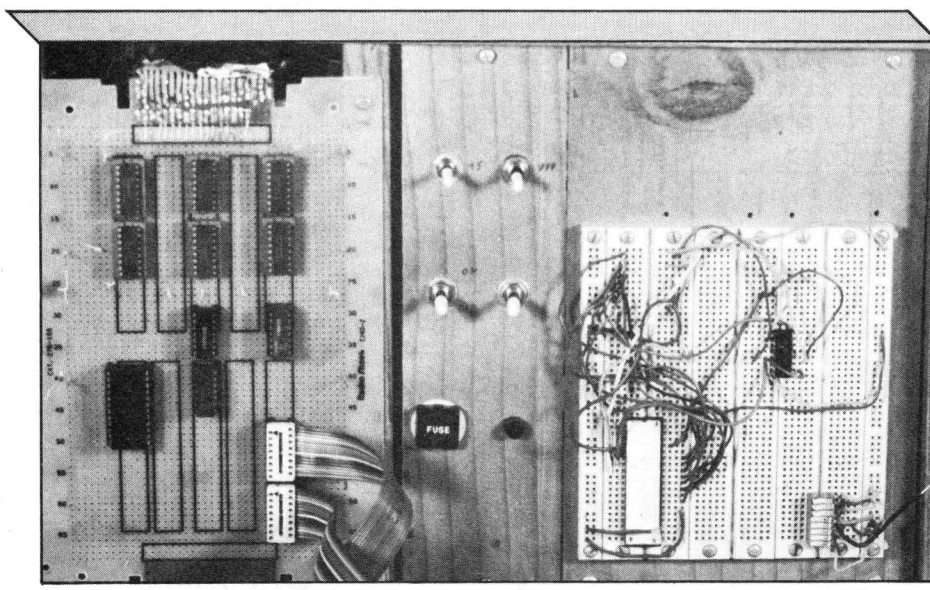
The data bus comes in on lines 10–15 (Z4) and 16 and 17 (Z5). These are labeled DW (data write) as the flow of data is from left to right when Z4 and a portion of Z5 are enabled. The lines marked DR0–DR7 are data read signals and the flow of data is from right to left when a portion of Z5 and Z6 is enabled. The data lines to the ROM pack are connected to the computer side of the buffers. The data lines are also buffered DB0–DB7 and connected to S1, a 16-pin wire-wrap socket.

The SCS signal is a spare device signal that goes low when addresses FF40H–FF5FH are selected (65344–65375 decimal). This signal and the R/W signal go to Z7A, an OR gate. When both are low, Z4 and a portion of Z5 are enabled allowing the data to pass through the buffers from left to right (a write operation).

The R/W also goes to Z8A where it is inverted, and it and the SCS signal go to Z8B to form a read signal that enables part of Z5 and Z6. This allows the data to pass from right to left for a read operation, thus the data buffers are only enabled when an address from FF40H–FF5FH is selected.

Both the data read (DR) and the data write (DW) go to Z10A, an AND gate, so when either one goes low, Z9, the one-of-16 decoder, is enabled. A0–A3 also go to Z9. These four address lines are decoded and provide 16 select signals for controlling various peripherals.

As stated earlier S1 and S2 are 16-pin wire-wrap sockets and you can bring out any signals you need here. I brought out D0–D7, the DW and DR signal, the +5V from the interface



The top of the interface. The breadboard sockets, used for prototyping circuits, are on the right.

power supply, circuit ground, and +21V using S1. The +21V is an optional second power supply in the interface box I use to burn 2716 EPROMs. The other pins of S1 and S2 bring out the first six select signals and A0–A1. These provide all the signals necessary for an Intel 8255 PPI chip, which I use to burn EPROMs.

A +5V power supply is needed to power this. I used +5V at 3A and it has been more than adequate for my purposes. I mounted the power supply and circuit board in a small scratch-built wooden box along with some breadboard strips to test circuits on.

After wiring this and double checking everything, plug it into the computer. Turn the interface on first, and then the computer. If you turn the computer on first, it will hang up when the interface is turned on. If, after everything is turned on, the computer seems to be lost in a loop, check your wiring in the data buffer section and R/W signals. A wiring error here is not forgiven by the computer.

If everything looks right, you can test the circuit using the following Basic commands:

```
POKE 65344,0
```

which applies all lows on the data bus at S1 and sets pin 1 of Z9 low; and

```
POKE 65345,255
```

which applies all ones to the data bus at S1 and sets pin 2 of Z9 low. Both of the above also set DW low. To test the input of this circuit, use the following:

```
LET A = PEEK (65344)
PRINT A
```

If everything is right, you should have a 255 printed on the screen. Pin 1 of Z9 and DR should go momentarily low. By applying highs and lows in various patterns to pins 1–8 of S1 and running the above routine, you should see the corresponding numbers printed on the screen.

Last of all, check and double check the wiring on the ROM-pack socket. An error here could destroy your cartridge. Then plug in your cartridge and proceed as you normally would. If you're using a cartridge like EDTASM+ this allows you to monitor your various I/O signals as you test your programs. ■

Address correspondence to Michael Kraft, 1830 E St., Merced, CA 95340.

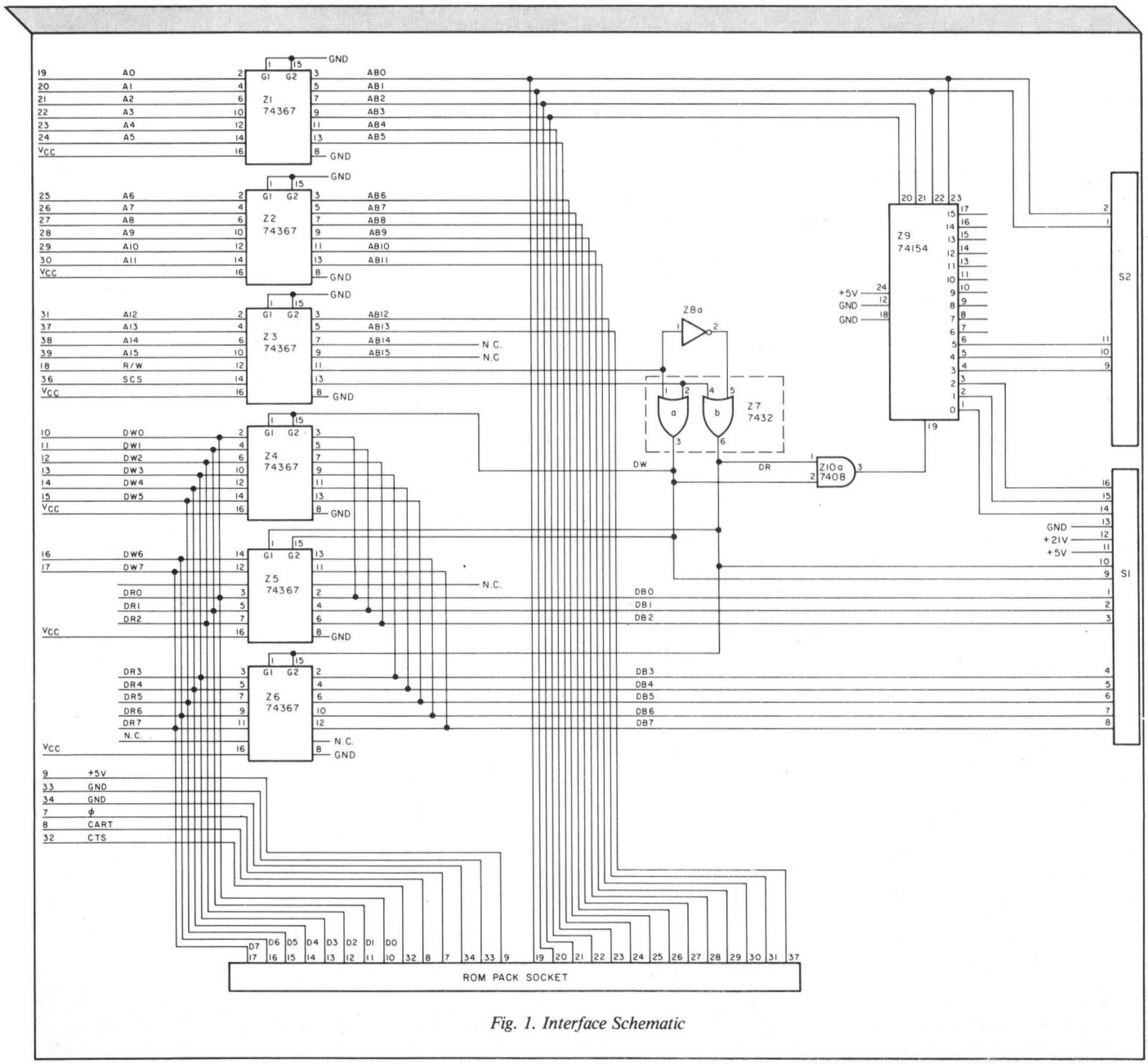


Fig. 1. Interface Schematic

BY MARK SILVERBLATT

COLORMANIA—PART II

More for the Colormaniac! High-speed graphics in a machine-language routine—yours for the reading.

The first "Colormania" article (*HOT CoCo*, January 1984, p. 100) explored number systems, showed how CoCo machine language compares to Basic programming, and gave you enough rudimentary Assembly mnemonics to program a simple loop counter. If you are a beginner one major difficulty you probably encounter is the complexity of machine-language programming, so I explained some fundamentals before proceeding to the programming.

By now the dedicated Colormaniac probably has some type of editor/assembler, has learned more of the 6809 instruction set, and is eager to proceed to more productive program-

System Requirements

16K RAM
Color, Extended, or Disk Basic
Editor/Assembler (optional)

Program Listing 1. Ping-Pong—Assembly Version

```
0001 0E00                       ORG $2600       PROGRAM WILL START HERE
0002 2600 2002                  BRA START

0003 2602              BALL     RMB 2           PATTERN STORAGE LOCATION

                       ******* SET HI RES MODE G6C

                       * SET UP VDG BITS V0-V2
                       * (SEE "GETTING STARTED..." PG 260 AND PG 263 TABLE 2)
0004 2604 7FFFC0       START    CLR $FFC0       CLEAR V0
0005 2607 7FFFC3                CLR $FFC3       SET   V1
0006 260A 7FFFC5                CLR $FFC5       SET   V2

                       * SET GRAPHICS MODE CONTROL VALUE (SEE PG. 260-261)
0007 260D B6FF22                LDA $FF22
0008 2610 8407                  ANDA #%00000111  DROP 5 MSBs
0009 2612 8AE0                  ORA #%11100000  LOAD 5 MSBs
0010 2614 B7FF22                STA $FF22       STORE THE VALUE

                       * SELECT PAGES 1 THRU 4 <3584-9727><$0E00-$25FF>
                       * (SEE PG. 263 TABLE 3)
0011 2617 7FFFC7                CLR $FFC7       SET   F0
0012 261A 7FFFC9                CLR $FFC9       SET   F1
0013 261D 7FFFCB                CLR $FFCB       SET   F2
0014 2620 7FFFCC                CLR $FFCC       CLEAR F3
0015 2623 7FFFCE                CLR $FFCE       CLEAR F4
0016 2626 7FFFD0                CLR $FFD0       CLEAR F5
0017 2629 7FFFD2                CLR $FFD2       CLEAR F6

                       ******* PAINT SCREEN RED ($0E00-$25FF)

0018 262C 8E0E00                LDX #$0E00      POINT TO TOP LEFT
0019 262F 86FF                  LDA #$FF        CONTROL FOR RED
0020 2631 A780         P@       STA ,X+         PAINT A PIXEL
0021 2633 8C25FF                CMPX #$25FF     DONE?
0022 2636 2DF9                  BLT P@          IF NOT, LOOP

                       ******* PAINT YELLOW BORDER
                       *FIRST THE VERTICAL
0023 2638 8655                  LDA #%01010101  SET PATTERN
0024 263A 8E1080                LDX #$1080      POINT TO TOP LEFT CORNER
0025 263D A7881F       P@       STA 31,X        PAINT RIGHT VERTICAL LINE
0026 2640 A784                  STA ,X          PAINT LEFT VERTICAL LINE
0027 2642 308820                LEAX 32,X       MOVE DOWN ONE PIXEL
0028 2645 8C25E0                CMPX #$25E0     DONE?
0029 2648 2DF3                  BLT P@          LOOP IF NOT DONE

                       *AND NOW THE HORIZONTAL
0030 264A 8E1080                LDX #$1080
0031 264D A7891500     P@       STA $1500,X     PAINT BOTTOM HORIZ LINE
0032 2651 A780                  STA ,X+         PAINT TOP HORIZ LINE
0033 2653 8C1100                CMPX #$1100     DONE?
0034 2656 2DF5                  BLT P@          IF NOT, LOOP

                       ******* INITIALIZE BALL POSITION AND PATTERN
0035 2658 8E1122                LDX #$1122      INITIAL POSITION
```

ming. This month's article concentrates on a high-speed machine-language graphics routine.

This graphics program draws a high-resolution screen with a rectangular yellow border on a red background and bounces a ball within the rectangle. It is a vehicle for discussing machine-language programming and the foundation for an arcade game (after the addition of paddles and scoring).

Before examining the program, you need to understand pages 252–266 of the Radio Shack manual, *Getting Started with Color Basic*. If you get lost in the initial graphics setup procedures, consult an expert.

Also, this article shows hex values prefaced by the standard dollar sign with decimal equivalents following in parentheses.

Program Listing 1 uses the highest-resolution, four-color mode, set up by lines 4–17. The CLR (clear) mnemonic is used here since writing anything to the appropriate addresses sets the SAM chip to the desired configuration. Placing the proper value in address $FF22 (65314) is tricky though, since the 3 least-significant bits should not be disturbed.

". . . to make smooth diagonal moves, the ball shifts one pixel at a time right or left and one pixel up or down, depending on the direction desired."

Do this by getting the value in this byte (line 7), using AND to drop the high 5 bits (line 8), ORing in 5 new high bits (line 9), and putting the result back (line 10).

Once the screen initialization is completed, you have a 128-by-192-pixel screen controlled by $1800 (6,144) bytes of RAM. The value at RAM address $0E00 (3,584) controls the four pixels starting at the screen's top left corner.

The next $1F (31) bytes control the remainder of the top horizontal row, and so on, until the bottom right screen corner is controlled by $25FF (9,727). This corresponds to Basic graphics pages 1–4 in the disk CoCo, but causes no problems in the nondisk system.

Clear the screen in lines 18–22 by loading index register X with the address of the top left control pixel, loading accumulator A with the value for red, and looping across and down the screen until the bottom right pixel is reached.

The border is drawn by a process like the screen clearing, but more selectively. Rather than using the entire screen, the border leaves a blank area at the top to display scores if you expand the routine into a complete game.

The pattern for yellow pixels is

```
0036 265B CC55FF            LDD #%0101010111111111  SET PATTERN
0037 265E ED8CA1            STD BALL,PCR    SAVE IT
0038 2661 1700F9            LBSR DOTON      GO LITE DOT

                    ******* DOT MOVEMENT ROUTINES

                    *SOUTHEAST
0039 2664 170124    SE      LBSR BEEP       MAKE SOUND
0040 2667 A6890080  D@      LDA 4*32,X      GET BYTE BELOW DOT
0041 266B 81FF              CMPA #$FF       IS IT RED?
0042 266D 2702              BEQ A@          IF SO,  CONTINUE
0043 266F 2033              BRA NE          ELSE GO NE
0044 2671 EC8C8E    A@      LDD BALL,PCR    GET PATTERN
0045 2674 C155              CMPB #%01010101  IS DOT @ FAR RIGHT?
0046 2676 2718              BEQ B@          IF SO, CHECK FOR BORDER
0047 2678 1700F4            LBSR DOTOFF     ELSE DOTOFF, THEN:
0048 267B EC8C84            LDD BALL,PCR    REFRESH
0049 267E 1A01              ORCC #%00000001  SET C
0050 2680 46                RORA            C-->A-->C
0051 2681 56                RORB            C-->B-->C
0052 2682 46                RORA            REPEAT
0053 2683 56                RORB
0054 2684 ED8DFF7A          STD BALL,PCR    SAVE PATTERN
0055 2688 308820    C@      LEAX 32,X       DROP A LINE
0056 268B 1700CF            LBSR DOTON      LITE IT
0057 268E 20D7              BRA D@          AND LOOP BACK
0058 2690 A602      B@      LDA 2,X         GET BYTE TO RIGHT
0059 2692 81FF              CMPA #$FF       IS IT RED?
0060 2694 264F              BNE SW          NOT RED=BORDER; JUMP SW
0061 2696 1700D6            LBSR DOTOFF     ELSE EXTINGUISH DOT
0062 2699 3001              LEAX 1,X        AND MOVE RIGHT
0063 269B CCD57F            LDD #%1101010101111111  RESTORE PATTERN
0064 269E ED8DFF60          STD BALL,PCR    SAVE PATTERN
0065 26A2 20E4              BRA C@          AND LOOP BACK


                    *NORTHEAST
0066 26A4 1700E4    NE      LBSR BEEP
0067 26A7 A688E0    D@      LDA -32,X
0068 26AA 81FF              CMPA #$FF
0069 26AC 2702              BEQ A@
0070 26AE 20B4              BRA SE
0071 26B0 EC8DFF4E  A@      LDD BALL,PCR
0072 26B4 C155              CMPB #%01010101
0073 26B6 2719              BEQ B@
0074 26B8 1700B4            LBSR DOTOFF
0075 26BB EC8DFF43          LDD BALL,PCR
0076 26BF 1A01              ORCC #%00000001
0077 26C1 46                RORA
0078 26C2 56                RORB
0079 26C3 46                RORA
0080 26C4 56                RORB
0081 26C5 ED8DFF39          STD BALL,PCR
0082 26C9 3088E0    C@      LEAX -32,X
0083 26CC 17008E            LBSR DOTON
0084 26CF 20D6              BRA D@
0085 26D1 A602      B@      LDA 2,X
0086 26D3 81FF              CMPA #$FF
0087 26D5 264B              BNE NW
0088 26D7 170095            LBSR DOTOFF
0089 26DA 3001              LEAX 1,X
0090 26DC CCD57F            LDD #%1101010101111111
0091 26DF ED8DFF1F          STD BALL,PCR
0092 26E3 20E4              BRA C@

                    *SOUTHWEST
0093 26E5 1700A3    SW      LBSR BEEP
0094 26E8 A6890080  D@      LDA 4*32,X
0095 26EC 81FF              CMPA #$FF
0096 26EE 2702              BEQ A@
0097 26F0 2030              BRA NW
0098 26F2 EC8DFF0C  A@      LDD BALL,PCR
0099 26F6 8155              CMPA #%01010101
0100 26F8 2713              BEQ B@
0101 26FA 8D73              BSR DOTOFF
0102 26FC 1A01              ORCC #%00000001
0103 26FE 59                ROLB
0104 26FF 49                ROLA
0105 2700 59                ROLB
0106 2701 49                ROLA
0107 2702 ED8DFEFC          STD BALL,PCR
0108 2706 308820    C@      LEAX 32,X
0109 2709 8D52              BSR DOTON
0110 270B 20DB              BRA D@
0111 270D A61F      B@      LDA -1,X
0112 270F 81FF              CMPA #$FF
0113 2711 1026FF4F          LBNE SE
0114 2715 8D58              BSR DOTOFF
0115 2717 301F              LEAX -1,X
0116 2719 CCFD57            LDD #%1111110101010111
0117 271C ED8DFEE2          STD BALL,PCR
0118 2720 20E4              BRA C@

                    *NORTHWEST
0119 2722 8D67      NW      BSR BEEP
0120 2724 A688E0    D@      LDA -32,X
0121 2727 81FF              CMPA #$FF
0122 2729 2702              BEQ A@
0123 272B 20B8              BRA SW
0124 272D EC8DFED1  A@      LDD BALL,PCR
0125 2731 8155              CMPA #%01010101
0126 2733 2713              BEQ B@
0127 2735 8D38              BSR DOTOFF
0128 2737 1A01              ORCC #%00000001
0129 2739 59                ROLB
0130 273A 49                ROLA
0131 273B 59                ROLB
0132 273C 49                ROLA
0133 273D ED8DFEC1          STD BALL,PCR
0134 2741 3088E0    C@      LEAX -32,X
0135 2744 8D17              BSR DOTON
0136 2746 20DC              BRA D@
0137 2748 A61F      B@      LDA -1,X
0138 274A 81FF              CMPA #$FF
0139 274C 1026FF54          LBNE NE
0140 2750 8D1D              BSR DOTOFF
```

Listing continued

Listing continued

```
0141 2752 301F              LEAX -1,X
0142 2754 CCFD57            LDD #%1111110101010111
0143 2757 ED8DFEA7          STD BALL,PCR
0144 275B 20E4              BRA C@

                       ******* SUBROUTINES

                       * LIGHT THE DOT
0145 275D EC8DFEA1     DOTON  LDD BALL,PCR     GET PATTERN
0146 2761 ED84                STD ,X           LIGHT FIRST ROW
0147 2763 ED8820              STD 32,X         SECOND ROW
0148 2766 ED8840              STD 2*32,X       THIRD ROW
0149 2769 ED8860              STD 3*32,X       FOURTH ROW
0150 276C 8D14                BSR DELAY        PAUSE
0151 276E 39                  RTS              RETURN

                       *EXTINGUISH THE DOT
0152 276F CCFFFF       DOTOFF LDD #$FFFF       GET RED PATTERN
0153 2772 ED84                STD ,X           PAINT IT
0154 2774 ED8820              STD 32,X
0155 2777 ED8840              STD 2*32,X
0156 277A ED8860              STD 3*32,X
0157 277D EC8DFE81            LDD BALL,PCR     RECOVER DOT PATTERN
0158 2781 39                  RTS

                       *PAUSE ROUTINE
0159 2782 108E0280     DELAY  LDY #$280        SET PAUSE LENGTH
0160 2786 313F         A@     LEAY -1,Y        DECREMENT
0161 2788 26FC                BNE A@           LOOP IF NOT DONE
0162 278A 39                  RTS

                       *MAKE THE SOUND
0163 278B 3410         BEEP   PSHS X           STORE POINTER
0164 278D C664                LDB #100         SOUND FREQUENCY
0165 278F D78C                STB $008C        SAVE IT
0166 2791 5F                  CLRB             MINIMUM DURATION
0167 2792 BDA951              JSR $A951        CALL SOUND FROM ROM
0168 2795 3510                PULS X           RETRIEVE POINTER
0169 2797 39                  RTS
```

loaded into A (line 23) and X is initialized with the address of the top left corner of the border (line 24). Line 26 draws each byte of the left border while the OFFSET command in lines 30–34 draws the right border, mapped $1F (31) control bytes higher. Lines 30–34 then draw horizontal border lines.

The ball's position is initialized on the screen inside the top left corner of the border (line 35), and the fun begins. The DOTON subroutine (lines 145–151) draws the ball four pixels high by four pixels wide, and DOTOFF (lines 152–158) extinguishes it.

In order to make smooth diagonal moves, the ball shifts one pixel at a time right or left and one pixel up or down, depending on the direction desired. For the initial top left to bottom right movement, this occurs in a subroutine called Southeast, or SE. There are three corresponding subroutines for movement in other directions, labelled NE, SW, and NW. Figure 1 shows how movement to the right is accomplished in the SE section.

Double accumulator D (accumulators A and B concatenated) and index register X jointly control the dot's position. While X points to a screen location, D sets the pattern for eight pixels at that location. Initially, as shown in Fig. 1a, the first 8 bits of D are set at 01-01-01-01, making the first four pixels yellow.

The remaining 8 bits are 11-11-11-11, so the next four pixels are red. Formation of the dot's first layer occurs when this pattern is placed into memory (line 153) and the other three layers are drawn as the pattern is continually repeated $20 (32) bytes higher (lines 154-156).

The dot is shifted to the right in Fig. 1b, when the pattern is changed to 11-01-01-01-01-11-11-11. In order to move diagonally, line 55 simultaneously shifts the dot down one pixel when X is incremented by a value of $20 (32).

This process continues through Fig. 1e, where the dot reaches the far right side of the 2 bytes controlled by the D accumulator. Figure 1f then shows how, by incrementing X by one (half the width of D) and restoring the pattern of Fig. 1b, the rightward movement continues.

Within the SE routine, the following things take place, in order;

- Beep—Make some noise.
- Test—See if the movement is about to encounter a border; if so, jump to the appropriate subroutine for another direction.
- Off—Extinguish the dot.
- Shift—Move the dot pattern one pixel down and one pixel to the right.
- On—Relight the dot in its new position.
- Delay—Enter a brief loop counter to smooth the motion, then loop back to Test and do it all again.

"The Assembly source code for the bouncing ball can be assembled to memory and saved to tape or disk only if you have an editor/assembler."

Don't let the strange-looking labels like A@ throw you; my assembler, the Micro Works' SDS80C, allows "local labels," single letters followed by @, which are valid only in a particular local area. The A@ in the SE section and the A@ in the NE section are two different labels, and any blank line in the source listing serves as a delimiter. If you enter this program with an assembler that does not allow local labels, you'll have to replace them with, for example, A-SE and A-NE.

The ORCC statement in line 49 needs explanation. This sets bit C in the condition code register, so that a "1" bit can be shifted into A with the RORA in line 50. This and the following lines 51–53 accomplish the shifting of bits within the D accumulator to move the dot.

The three other movement routines operate in essentially the same manner as SE, except that up becomes down or right becomes left, according to direction of movement.

Lines 163–169 use the Sound routine contained in ROM to generate the beep when the dot encounters a border. This is an example of the many routines in ROM that you can call. Here, you call the sound by loading $008C (140) with the desired frequency and the B accumulator with the desired duration, then transferring control to the ROM at $A951 (43345).

There's an RTS (return from subroutine) in ROM that transfers control back to your program after the sound is generated. While a Basic Sound command is limited to durations of 1–255, this routine uses a duration of zero (line 166). Try SOUND 100,0 in Basic and you'll get a function-call error. Stacking the X register upon entry and recovering it just before exiting from this routine is a necessity, since the sound-generating code contained in ROM changes X's value and crashes the whole program unless X is protected.

The Assembly source code for the bouncing ball can be assembled to memory and saved to tape or disk only if you have an editor/assembler. The entire routine is written in relocatable

or position-independent code, so the ORG $2600 statement <0001> can be altered and the program can reside wherever you desire, as long as the display memory $0E00–$25FF (3584–9727) is avoided.

If you have no editor/assembler, the Basic driver (Program Listing 2) contains the same code. The driver version has been relocated to the high RAM addresses in the 16K system, but works in the 32K system as well.

Type in the driver and save it to tape or disk before it is run, since the graphics manipulation alters values in the same addresses occupied by the Basic driver program. Disk users might avoid the conflict by executing a PCLEAR 4 before entering the program, but it is a good practice to save before running.

Program Listing 3, Datasave, is a Basic routine that takes bytes of machine-language code from memory and creates Basic DATA strings with the same values. It creates either tape or disk files, with DATA lines starting with line 1000. Non-Extended Color Basic has no CSAVEM command, but this program simulates one.

The only trick is to get Datasave and the machine-language program to reside in different memory areas so they won't destroy each other. Imagine that my bouncing ball routine is a commercial program called Ping-Pong, for which you paid $19.95, and that you want to protect your investment by making a back-up copy.

You must first determine the program's start, end, and EXEC addresses (the first and last RAM addresses occupied by the program), and the address to which control is transferred to start the program running. Start and EXEC addresses are the same in many machine-language programs, but not all. From examining the Assembly output (Listing 1), you know that Ping-Pong's start, end, and EXEC addresses are $2600 (9,728), $2797 (10,135) and $2600 (9,728), respectively.

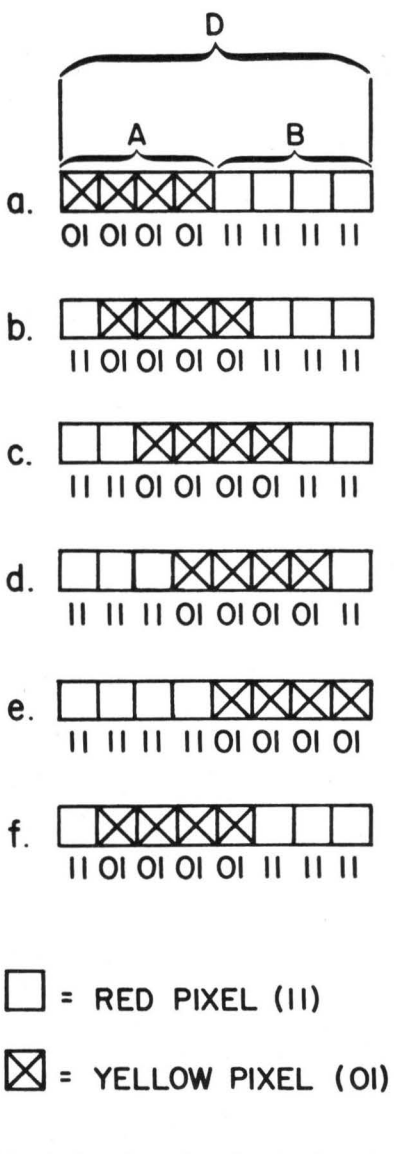


Fig. 1. Dot formations for the Bouncing Ball routine. Values held in the 2-byte (16-bit) D accumulator and placed into video RAM will specify eight consecutive pixels, 2 bits per pixel. When Fig. 1f is incremented by one (half the width of D) rightward movement occurs.

↵See List of Advertisers on page 130

You don't have this information if you buy Ping-Pong from a software company. You only have the object code on a cassette, and some detective work is necessary. Once the machine-language program is loaded from tape, each of these values is stored in 2 bytes of low RAM, with the most-significant byte first, as follows:

- The beginning address of the last machine-language program loaded from tape appears at $01E5 (485) and $01E6 (486).
- The end address plus one appears at $007E (126) and $007F (127).
- The EXEC address appears at $01E7 (487) and $01E8 (488).

Therefore, you can determine the start address by loading the machine-language program and, without executing it, typing:

```
PRINT 256*PEEK(485) + PEEK(486)
<ENTER>
```

and the CoCo responds with 9728. For end address plus one, you can then type:

```
PRINT 256*PEEK(126) + PEEK(127)
<ENTER>
```

and the response is 10136. Since this is end + 1, you know that the actual end address is 10135. Finally, you determine the EXEC address by typing:

```
PRINT 256*PEEK(487) + PEEK(488)
<ENTER>
```

and find that the EXEC address is also 9728. If you have Extended Color Basic you can now make a back-up tape

"You can add scoring and joystick controlled paddles..."

by inserting a blank cassette, pressing the play and record buttons, and observing the syntax for creating machine-language tapes, which is CSAVEM "file name", start address, end address, EXEC address, or:

```
CSAVEM "PINGPONG",9728,10135,9728
<ENTER>
```

or if you prefer hex, use:

```
CSAVEM "PINGPONG",&H2797,&H2600
```

This also holds true for the disk-system user, but only if the program was loaded from tape. Machine-language programs loaded from disk contain start, end, and EXEC addresses on track 17 of the disk.

If you have only non-Extended Basic (no CSAVEM command) or for any other reason you want to make a back-up in Basic driver form, you can use Datasave to create the driver, but, as mentioned above, avoid address conflicts. Remember also that Ping-Pong is written in position-independent code, meaning that it can be relocated to any available RAM address and still function properly. How do you determine if there is a memory conflict, and whether it is necessary to relocate the machine-language routine?

The easiest way is to load Datasave, protect the memory in which the machine-language routine resides, and see what happens. The syntax is CLEAR n,h, where n determines the number of bytes to be reserved for string storage and h specifies the highest address that can be used by Basic. If no CLEAR command is used, then upon power-up the CoCo reserves 200 bytes for strings

and allows Basic to use everything up to and including the highest available RAM address.

Since you know that Ping-Pong starts at $2600 (9728), you might try loading Datasave and executing a CLEAR200,9727, telling the CoCo not to let Basic use anything at or above Ping-Pong's start address. You can also execute the CLEAR statement before loading the Basic program.

In either case, the first sign that you haven't left enough RAM to do the job is an ?OM error. If no ?OM error occurs at this point you can assume that the CLEAR statement left enough RAM for Datasave to reside.

Although the CLEAR statement set aside 200 bytes for string storage, it goes looking for RAM space in which to store numeric variables once Datasave starts to run. If any arrays are used, they'll eat RAM like a glutton.

After successfully executing CLEAR n,h and loading the Basic program, it's a good idea to execute a PRINTMEM statement to get some idea of how much user RAM remains. The Datasave program uses only a small handful of numeric variables and no arrays, so if the PRINT MEM shows just 50 bytes or so, they should suffice.

Assume that you do have insufficient memory working with the 16K system, and that you want to get Ping-Pong out of the way by moving it to the highest available RAM address. Examining the CoCo manual's memory map, you see that the user RAM in the 16K system ends at $3FFF (16383). As Ping-Pong ends at $2797 (10135), simple subtraction tells you how far you can move it up:

$3FFF – $2797 = $1868, or
(16383) – (10135) = (6248)

Once you determine the offset value of $1868 (6248), find the new start and EXEC addresses by adding the offset to the original values:

$2600 + $1868 = $3E68, or
(9728) + (6248) = (15976)

If the original start and EXEC addresses differ, you use the same addition process to determine a new EXEC address.

At this point enter a CLEAR 200,15975 statement. This limits Basic to use of RAM up to the last address below the new machine-language start location. You can then offset load the machine-language program with the command: CLOADM "PINGPONG",6248.

Now you can load and run Datasave. Prompts ask for the machine-language routine's start and end addresses and for the driver file name. Do not use a file-name extension if writing the driver to disk. Datasave automatically uses "/BAS".

Once the driver has been written to tape or disk, it consists only of DATA statements, and you must add memory protection, a FOR...NEXT loop to read the data and POKE it into memory, and the EXECUTE command, as shown in Listing 3, lines 10–40, respectively. Also, it is written in ASCII format and can be loaded and resaved in Basic format to speed future loading.

You can add scoring and joystick-controlled paddles to make this a complete Ping-Pong game. You have only four directions of movement here. Should a finished product provide for "English," so that the ball's speed and angle off the paddle depend on which part of the paddle is contacted? I can answer this and other important questions in a future "Colormania," but meanwhile you might want to work on it yourself. ■

Address correspondence to Mark Silverblatt, HHC, 93d Signal BDE, Box 181, APO, New York 09279.

```
1 REM BASIC DRIVER FOR ML PONG R
OUTINE
2 REM LOADS PONG TO HIGH RAM
10 CLEAR 200,15975
20 FOR A=15976 TO 16383
30 READ D:POKEA,D:NEXT
40 EXEC15976
1000 DATA 32,2,42,0,127,255,192,
127,255,195,127,255,197,182
1010 DATA 255,34,132,7,138,224,1
83,255,34,127,255,199,127,255
1020 DATA 201,127,255,203,127,25
5,204,127,255,206,127,255,208
1030 DATA 127,255,210,142,14,0,1
34,255,167,128,140,37,255,45
1040 DATA 249,134,85,142,16,128,
167,136,31,167,132,48,136,32
1050 DATA 140,37,224,45,243,142,
16,128,167,137,21,0,167,128
1060 DATA 140,17,0,45,245,142,17
,34,204,85,255,237,140,161,23
1070 DATA 0,249,23,1,36,166,137,
0,128,129,255,39,2,32,51,236
1080 DATA 140,142,193,85,39,24,2
3,0,244,236,140,132,26,1,70
1090 DATA 86,70,86,237,141,255,1
22,48,136,32,23,0,207,32,215
1100 DATA 166,2,129,255,38,79,23
,0,214,48,1,204,213,127,237
1110 DATA 141,255,96,32,228,23,0
,228,166,136,224,129,255,39
1120 DATA 2,32,180,236,141,255,7
8,193,85,39,25,23,0,180,236
1130 DATA 141,255,67,26,1,70,86,
70,86,237,141,255,57,48,136
1140 DATA 224,23,0,142,32,214,16
6,2,129,255,38,75,23,0,149,48
1150 DATA 1,204,213,127,237,141,
255,31,32,228,23,0,163,166,137
1160 DATA 0,128,129,255,39,2,32,
48,236,141,255,12,129,85,39
1170 DATA 19,141,115,26,1,89,73,
89,73,237,141,254,252,48,136
1180 DATA 32,141,82,32,219,166,3
1,129,255,16,38,255,79,141,88
1190 DATA 48,31,204,253,87,237,1
41,254,226,32,228,141,103,166
1200 DATA 136,224,129,255,39,2,3
2,184,236,141,254,209,129,85
1210 DATA 39,19,141,56,26,1,89,7
3,89,73,237,141,254,193,48,136
1220 DATA 224,141,23,32,220,166,
31,129,255,16,38,255,84,141
1230 DATA 29,48,31,204,253,87,23
7,141,254,167,32,228,236,141
1240 DATA 254,161,237,132,237,13
6,32,237,136,64,237,136,96,141
1250 DATA 20,57,204,255,255,237,
132,237,136,32,237,136,64,237
1260 DATA 136,96,236,141,254,129
,57,16,142,2,128,49,63,38,252
1270 DATA 57,52,16,198,100,215,1
40,95,189,169,81,53,16,57
```

Program Listing 2. Ping-Pong—Basic Driver Version

```
10 REM DATASAVE- (C) 1983, COLOR
MANIA CO.
20 CLS:CLEAR200
30 PRINT"DATASAVE - UTILITY TO C
REATE"
40 PRINT"BASIC DATA LINES FROM M
L PROGRAM"
50 PRINT:PRINT"CAUTION- ML PROGR
AM MUST BE PUT"
60 PRINT"INTO MEMORY BEFORE RUNN
ING THIS"
70 PRINT"ROUTINE, AND ADDRESSES
MUST NOT"
80 PRINT"CONFLICT- USE CLEAR OR
PCLEAR"
90 PRINT"STATEMENTS AS NECESSARY
":PRINT
100 PRINT"ML PROGRAM IS TO BE SA
VED TO-":PRINT"1-TAPE":PRINT"2-D
ISK":PRINT:PRINT"SELECT ONE"
110 I$=INKEY$:IFI$=""THEN110
120 I=VAL(I$):IFI<>1ANDI<>2THEN1
10
130 CLS:PRINT"ML PROGRAM'S START
 ADDRESS":INPUTS
140 PRINT"ML PROGRAM'S END ADDRE
SS":INPUTE
150 CLS:PRINT"ENTER NAME OF ";:I
F I=1 THENPRINT"TAPE";ELSEPRINT"
DISK";
160 PRINT" FILE TO BE":PRINT"CRE
ATED"
170 LINEINPUTFI$:IF LEN(FI$)=0TH
EN150 ELSE FI$=LEFT$(FI$,8)
180 IF I=1 THEN DE=-1 ELSE DE=1:
FI$=FI$+"/BAS"
190 SL=1000'Initialize line numb
er
200 OPEN "O",#DE,FI$
210 DA$=STR$(SL)+" DATA "'Begin
data string
220 S$=STR$(PEEK(S))'Get value
230 S$=RIGHT$(S$,LEN(S$)-1)'Drop
 leading blank
240 DA$=DA$+S$  'Build data stri
ng
250 IF S=E THEN 270 'Test for la
st byte
260 IF LEN(DA$)<60 THEN DA$=DA$+
",":S=S+1:GOTO220'Add comma and
loop
270 PRINTDA$:PRINT#DE,DA$' Dump
to tape/disk
280 IF S<>E THEN SL=SL+10:S=S+1:
GOTO210 ELSE CLOSE#DE:END 'Loop
back if not done
```

Program Listing 3. Datasave

BY DENNIS H. WEIDE

DENTAL/MEDICAL BILL BALANCER

Keep track of your family's dental and medical expenses and make filling out your tax forms easier.

If you're in the same financial boat that I am (the Titanic), then you probably have a few dental and medical bills that become a headache at tax time. You have to figure the amount you paid and the amount your insurance covers because Uncle Sam wants you to be as accurate as possible. He doesn't want you to pay too much tax, right?

I've written two programs that allow you to keep track of medical and dental bills by entering the bills into the program as data and saving them to tape. When you receive reimbursement from the insurance company, you can load the data from tape and then enter the payment received. The computer will match the insurance payments to the original bills and, upon command, will print a summary for you.

The summaries will total the data for you and show each medical or dental service as well as the total cost of services, total paid by insurance and total cost to you (employee). In addition, a cross-hatch (#) will appear in the column to the right of the insurance payment for any entry that does not have a matching claim entered. This helps to spot the services that you might want to investigate.

The programs require Extended Color Basic because of the PRINT USING and LINE INPUT statements. You can delete USING and LINE for use with Color Basic.

Before entering either program, type:

```
POKE 25,6:NEW (enter)
```

to clear maximum memory. If you have 32K, this won't be necessary. This leaves you with zero graphics pages. Type in either program, but leave out the REMs. You will have more memory for data storage. If you have a 32K machine, you can combine both listings as one program—more about that later. The variables are compatible so you won't have to worry.

You can leave off the printer routines if you don't have a printer to save even more memory. The amount of memory used by the program depends upon the amount of data entered. If you enter long statements or names, you will use more memory. It is best to use initials instead of names and numbers for the date whenever possible. Because of the small screen format, data printed on the screen in both programs is in an abbreviated format. The printer data will be the same as the data you entered.

The major difference between these two programs is the way they summarize data. They use similar routines to match data, but the dental program (Listing 1) only does it when you request a summary and answer the prompt with a 3. This means you must ask for a summary before you ask for a printout or your results will show all zeros. This requires more action by the operator, but saves a lot of memory.

The medical program (Listing 2) is much more sophisticated. It does its own summary during the print routine, and therefore does not require a summary before printing. It also has an editor that allows the user to change any

System Requirements

16K RAM
Extended Color Basic
(Color Basic with minor changes)

entry. This is a handy feature when the insurance company makes a mistake and has to send you another check. You can edit the data in the dental program but you must stop the program execution to do so; I'll explain this later.

Both programs require that you load data from cassette (if you need to) before you make any keyboard entries. If you don't, you will overwrite your keyboard entries and they will be lost. You can, however, reenter them if you want. Both programs will warn you if you try this and an operator input is required to override the warning.

The dental program is much shorter because it contains only the basics for entering, saving, matching, and summarizing the data.

"If you end the program prematurely, don't worry. Just type CONT and press enter. The program will take you back to the menu without any loss of data."

The medical program, however, allows you to do everything the dental program does and more. It keeps track of your medicine purchases and provides for a description of the medical visit. In addition, line 880 figures the sales tax for medicine purchases. For New Mexico, the tax is 4 percent. If your tax is different, change this line to reflect the tax in your state or delete the line altogether. If you leave this line in, enter the price of the medicine without the tax. The program will do the rest for you.

If you end the program prematurely, don't worry. Just type CONT and press enter. The program will take you back to the menu without any loss of data.

Printer Control Codes

I have included printer control codes in both programs. If you don't have a Line Printer VII, you must adjust the

Program Listing 1. Dental Expenses

```
170 CLS:PRINT:CLEAR 1000
180 P1$=STRING$(32,"%"):P2$=CHR$
(10):P3$=CHR$(31):P4$=CHR$(30)
190 DIM D$(20),D1$(20),I$(20),I1
$(20),A(20),A1(20),A2(20)
200 PRINTTAB(9)"DENTAL PROGRAM"
210 PRINT:PRINTTAB(7)"BY DENNIS
H. WEIDE"
220 PRINT:PRINTTAB(9)"COPYRIGHT
1982"
230 FOR X=1 TO 550
240 NEXT X
250 GOTO 1510
260 IF P=1 THEN 280
270 IF P=2 THEN 650
280 CLS
290 IF N7=1 GOTO 580
300 N7=1
310 '
320 '
330 '     ENTER DENTAL VISITS
340 '
350 '
360 INPUT"ENTER NUMBER OF VISITS
";N1
370 CLS:FF=0:ZQ=1
380 PRINTTAB(9)"DENTAL VISITS"
390 GOSUB 1420
400 FOR X=N3+1 TO N1+N3
410 PRINT@I1,"";:LINEINPUT D$(X)
420 PRINT@I2,"";:LINEINPUT I$(X)
430 PRINT@I3,"";:INPUT A(X)
440 I1=I1+32:I2=I2+32:I3=I3+32
450 NEXT X
460 FOR X=N3+1 TO N1+N3
470 T1=T1+A(X)
480 NEXT
490 PRINT:PRINTTAB(5)"TOTAL CHAR
GES ";
500 PRINTUSING"$####.##";T1
510 PRINT:PRINT:N3=N3+N1
520 GOTO 630
530 '
540 '
550 '     ERROR ROUTINE
560 '
570 '
580 PRINT:PRINT
590 PRINTTAB(4)"YOU CANNOT REENT
ER DATA"
600 PRINT:PRINT:PRINT
610 INPUT"    PRESS <ENTER> TO CO
NTINUE";X
620 GOTO 1510
630 INPUT"    PRESS <ENTER> TO CO
NTINUE";I4:GOTO 1510
640 END
650 CLS
660 IF N8=1 THEN 580
670 N8=1
680 '
690 '
700 '     ENTER INSURANCE CLAIMS
710 '
720 '
730 INPUT"ENTER NUMBER OF CLAIMS
";N2
740 CLS:PRINTTAB(6)"INSURANCE PA
YMENTS"
750 ZQ=1:FF=0:GOSUB 1420
760 FOR X=N4+1 TO N2+N4
770 PRINT@I1,"";:LINEINPUT D1$(X
)
780 PRINT@I2,"";:LINEINPUT I1$(X
)
790 PRINT@I3,"";:INPUT A1(X)
800 I1=I1+32:I2=I2+32:I3=I3+32
810 NEXT X
820 FOR X=N4+1 TO N2+N4
830 T2=T2+A1(X):NEXT X
840 PRINT:PRINTUSING"TOTAL INSUR
ANCE REC'D $$####.##";T2
850 PRINT:PRINT:N4=N4+N2
860 INPUT"    PRESS <ENTER> TO CO
NTINUE";I4:GOTO 1510
```

Listing continued

codes to work for your printer. The following codes are for the Line Printer VII.

CHR$(10) = Line feed with carriage return
CHR$(30) = Standard print (normal condition)
CHR$(31) = Double-wide print
STRING$(5,10) = Five line feeds with carriage return (can be replaced with CHR$(12) for form feed)
STRING$(80,"–") = for 80-column printer (change to column width of your printer)

If your codes are different from these, just change the CHR$ values in line 180 in the dental program and lines 220 and 230 in the medical program to the proper values for your printer.

"Try a few practice entries so that you become familiar with how the program works. Error routines prevent any mistakes that could result in the loss of data."

Making Entries

It's very simple to make entries in either program. When you run the program, the main menu will appear. Just type the number of the function you want and press the enter key. Try a few practice entries so that you become familiar with how the program works. Error routines prevent any mistakes that could result in the loss of data.

If for any reason you must edit an entry in the medical program, return to the main menu and select the editor routine. Another menu will be displayed. Make your choice from this menu and enter the old data requested. You will be asked to enter the new data (follow the prompts). When you press the enter key, you overwrite the old data with the new data. If you make a mistake when editing, just edit the error again.

Editing the dental program is a little more difficult. First, you must determine which line of the printout needs to

Listing continued

```
870 '
880 '
890 '     SUMMARY
900 '     IF YOU HAVE MORE THAN
910 '     20 CLAIMS
920 '     CHANGE X & Y VALUES
930 '
940 '
950 CLS:PRINT:PRINT:PRINT:PRINT
P1$
960 PRINTTAB(8)"ONE MOMENT PLEAS
E":PRINT:PRINT P1$
970 FOR X=1 TO 20
980 FOR Y=1 TO 20
990 IF D1$(Y)=D$(X) AND I1$(Y)=I
$(X) THEN A2(X)=A1(Y)
1000 NEXT Y,X
1010 CLS
1020 PRINT:INPUT"   ENTER PAGE N
UMBER (1-3)";P
1030 IF P<1 OR P>3 THEN 1050
1040 ON P GOTO 1120,1130,1210
1050 PRINT:PRINTTAB(9)"INVALID I
NPUT"
1060 GOTO 1020
1070 '
1080 '
1090 '     PAGE PRINT ROUTINE
1100 '
1110 '
1120 P1=1:P2=12:GOSUB 1650:GOTO1
140
1130 P1=13:P2=20:GOSUB 1650
1140 FOR X=P1 TO P2
1150 IF A(X)<1 THEN 1180
1160 GOSUB 1740
1170 GOSUB 1700
1180 NEXT
1190 GOSUB 1720
1200 GOTO 1790
1210 CLS:FF=1
1220 TA=0:TP=0
1230 FOR X=1 TO 20
1240 TA=TA+A(X)
1250 TP=TP+A2(X)
1260 NEXT
1270 '
1280 '
1290 '     PRINT RESULTS ON SCRE
EN
1300 '
1310 '
1320 PRINT:PRINTTAB(2)"TOTAL AMO
UNT PAID BY PATIENT"
1330 PRINT:PRINTTAB(12)"";:PRINT
USING"$####.##";TA
1340 PRINT:PRINT" TOTAL AMOUNT P
AID BY INSURANCE"
1350 PRINT:PRINTTAB(12)"";:PRINT
USING"$####.##";TP
1360 PRINT:PRINTTAB(5)"TOTAL UNI
NSURED AMOUNT"
1370 PRINT:PRINTTAB(12)"";:PRINT
USING"$####.##";TA-TP
1380 PRINT:PRINT:PRINT
1390 INPUT"   PRESS <ENTER> TO C
ONTINUE";P
1400 GOTO 1790
1410 END
1420 I1=97:I2=106:I3=116
1430 PRINT"  DATE     NAME
AMOUNT"
1440 FOR X=1 TO 32:PRINT"*";:NEX
T
1450 RETURN
1460 '
1470 '
1480 '     DISPLAY MAIN MENU
1490 '
1500 '
1510 CLS:PRINT:PRINT"   DENTAL I
NSURANCE PROGRAM"
1520 PRINT
1530 PRINTTAB(3)"1=ENTER DENTAL
VISIT"
1540 PRINTTAB(3)"2=ENTER INSURAN
CE PAYMENT"
1550 PRINTTAB(3)"3=SUMMARIZE ALL
 DATA"
1560 PRINTTAB(3)"4=LOAD DATA"
```

Listing continued

be edited. Then you must stop program execution. Do this by pressing the break key or entering a 7 if the main menu is displayed. Then use the following table to determine the data variable that must be changed.

Dental Variables

Dental Visit Date = D$(X)
Dental Visit Name = I$(X)
Dental Visit Amount = A(X)
Insurance Payment Date = D1$(X)
Insurance Payment Name = I1$(X)
Insurance Payment Amount = A1(X)
(where X = line in Dental Expense Summary)

Let's look at an example. Suppose the data sample has an error in line 5. Here's line 5:

```
JAN 24  DEBBIE  $18.72  $9.00  $9.72
```

It shows an insurance payment of $9. That was the original amount paid by the insurance company. After inquiring about the claim, I received another check for $9.72. The total the insurance company paid on this claim then was $18.72. You now know that the error is on line 5 of the insurance payment. Enter the following:

```
A1(5) = 18.72:CONT
```

This will change the insurance payment amount and restart the program. Press the enter key one more time to get an illegal entry warning. The program will return to the main menu or the point at which you pressed the break key. You can now ask for a summary and verify that the data is now correct.

For 32K Machines

If you have 32K, you can combine the two programs and keep your medical and dental records separate while using only one program. First, type in the first program and save it to tape. Then clear the memory by typing NEW and pressing the enter key. Now type in the second program. After the second program has been entered, type:

```
PRINT PEEK(25),PEEK(26) (enter)
```

to locate the program pointers. Write down the values on the screen for later use. Now type:

Listing continued

```
1570 PRINTTAB(3)"5=SAVE DATA"
1580 PRINTTAB(3)"6=HARDCOPY"
1590 PRINTTAB(3)"7=END PROGRAM"
1600 PRINT
1610 INPUT"   ENTER ONE OF THE A
BOVE";MM
1620 ON MM GOTO 280,650,890,2030
,2290,2670,3060
1630 GOSUB 1910
1640 GOTO 1510
1650 CLS
1660 B1=64:B2=71:B3=79:B4=88
1670 PRINT" DATE   NAME    CHARG
E  INS PMT"
1680 FOR X=1 TO 32:PRINT"*";:NEX
T
1690 RETURN
1700 B1=B1+32:B2=B2+32:B3=B3+32:
B4=B4+32
1710 RETURN
1720 PRINT@451,"PRESS <ENTER> TO
 CONTINUE":INPUT P
1730 RETURN
1740 PRINT@B1,D$(X);
1750 PRINT@B2,I$(X);
1760 PRINT@B3,"";:PRINTUSING"$##
#.##";A(X);
1770 PRINT@B4,"";:PRINTUSING"$##
#.##";A2(X);
1780 RETURN
1790 CLS:PRINT
1800 PRINT"  IF YOU WANT ANOTHER
 SUMMARY"
1810 PRINT"  PAGE, ENTER PAGE NU
MBER(1-3)"
1820 INPUT"          PAGE"; P
1830 IF P>0 AND P<4 THEN 1040
1840 GOTO 1510
1850 FOR X=1 TO 1000:NEXT X:GOTO
 1510
1860 '
1870 '
1880 '    ERROR ROUTINE
1890 '
1900 '
1910 FOR X=1 TO 10:CLS
1920 FOR T=1 TO 50:NEXT T
1930 PRINT@233,"ILLEGAL ENTRY"
1940 SOUND 100,1
1950 FOR Y=1 TO 50:NEXT Y,X
1960 CLS
1970 RETURN
1980 '
1990 '
2000 '    LOAD DATA FROM CASSET
TE
2010 '
2020 '
2030 IF ZQ=1 THEN 2560
2040 CLS:PRINT P1$;:PRINTTAB(11)
"LOAD DATA":PRINT P1$
2050 PRINT:PRINTTAB(7)"1. LOAD D
ATA TAPE"
2060 PRINT:PRINTTAB(7)"2. PRESS
PLAY ONLY"
2070 PRINT:PRINTTAB(7)"3. PRESS
ENTER"
2080 LINEINPUT ZZ$
2090 CLS:PRINT:PRINT:PRINT:PRINT
 P1$;
2100 PRINTTAB(6)"YOU ARE LOADING
 DATA":PRINT P1$
2110 OPEN"I",-1,"NEW"
2120 INPUT#-1,N3
2130 INPUT#-1,N4
2140 FOR X=1 TO N3
2150 INPUT#-1,I$(X)
2160 INPUT#-1,D$(X)
2170 INPUT#-1,A(X)
2180 INPUT#-1,D1$(X)
2190 INPUT#-1,I1$(X)
2200 INPUT#-1,A1(X)
2210 NEXT X
2220 CLOSE#-1
2230 ZQ=1
2240 GOTO 1510
2250 '
2260 '
```

Listing continued

POKE 25,PEEK(27):POKE 26,PEEK(28) – 2 (enter).

This will set the pointers to the end of the previous program so that the next program will not overwrite it.

Now you can load the program you saved on tape and renumber it starting with a line number higher than the last line number of the previous program. Type:

POKE 25,X:POKE 26,Y (enter)

(where X and Y are the values of addresses 25 and 26 seen earlier) to set the program pointers back to the beginning of the previous program. Save a copy of both programs on tape in case you make an error while trying to load them back to back.

You can add a menu and combine the printer routines as well as the data saves and loads. I used the Basic Line Moer program by Jack Aker in the July 1982 issue of *Color Computer News*.

When deleting remarks in the dental program, don't delete line 890. The program jumps to this line number and without it you will get an error message. It should also be left in as a reference point. If you must make more than 15 entries, change the values in this routine.

When in doubt about what to enter in either program, simply press the enter key. The error routines will prevent any mistakes that might lead to loss of data. In the dental program, pressing enter without a page number during a summary will return you to the main menu.

These two programs have saved me a lot of headaches over the last two years and have helped keep an accurate record of medical and dental expenses. I think you will find them quite useful. ■

Address correspondence to Dennis Weide, 14201 Marquette N.E., Albuquerque, NM 87123 .

Listing continued

```
2270 '     SAVE DATA TO CASSETTE

2280 '
2290 '
2300 CLS:PRINT P1$;
2310 PRINTTAB(11)"SAVE DATA":PRI
NT P1$
2320 PRINT:PRINTTAB(7)"1. LOAD D
ATA TAPE"
2330 PRINT:PRINTTAB(7)"2. PRESS
RECORD & PLAY"
2340 PRINT:PRINTTAB(7)"3. PRESS
ENTER"
2350 LINEINPUT ZZ$
2360 CLS:PRINT:PRINT:PRINT:PRINT
 P1$;
2370 PRINTTAB(6)"YOU ARE SAVING
DATA":PRINT P1$
2380 OPEN"O",-1,"NEW"
2390 PRINT#-1,N3
2400 PRINT#-1,N4
2410 FOR X=1 TO N3
2420 PRINT#-1,I$(X)
2430 PRINT#-1,D$(X)
2440 PRINT#-1,A(X)
2450 PRINT#-1,D1$(X)
2460 PRINT#-1,I1$(X)
2470 PRINT#-1,A1(X)
2480 NEXT X
2490 CLOSE#-1
2500 GOTO 1510
2510 '
2520 '
2530 '     ERROR ROUTINE
2540 '
2550 '
2560 CLS:PRINT P1$:PRINT:PRINT
2570 PRINTTAB(6)"DATA ALREADY LO
ADED":PRINT
2580 PRINTTAB(2)"DO YOU WANT TO
RELOAD? (Y/N)";
2590 FOR II=1 TO 10:SOUND 100,3:
NEXT
2600 LINE INPUT KK$
2610 IF KK$="Y" THEN 2040 ELSE 1
510
2620 '
2630 '
2640 '     PRINT RESULTS TO PRIN
TER.
2650 '
2660 '
2670 CLS:IF FF=1 THEN 2730
2680 PRINT:PRINT:PRINT P1$;
2690 PRINTTAB(5)"YOU MUST SUMMAR
IZE DATA"
2700 PRINTTAB(5)"PAGE 3 BEFORE P
RINTING":PRINT P1$
2710 FOR X=1 TO 1000:NEXT X
2720 GOTO 1510
2730 PRINT P1$;:PRINTTAB(12)"PRI
NTING":PRINT P1$
2740 PRINT#-2,P2$;P3$;STRING$(40
,"#");P2$
2750 PRINT#-2,TAB(9)"DENTAL EXPE
NSE SUMMARY";P2$
2760 PRINT#-2,STRING$(40,"#");P2
$;P2$;P4$
2770 PRINT#-2,TAB(6)"DATE";TAB(1
7)"NAME";
2780 PRINT#-2,TAB(32)"CHARGE";
2790 PRINT#-2,TAB(47)"INSURANCE
PMT";
2800 PRINT#-2,TAB(66)"EMPLOYEE C
OST"
2810 PRINT#-2,STRING$(80,"-")
2820 FOR X=1 TO N3
2830 PRINT#-2,TAB(5);D$(X);TAB(1
6);I$(X);TAB(31)"";
2840 PRINT#-2,USING"$###.##";A(X
);:PRINT#-2,TAB(50)"";
2850 PRINT#-2,USING"$###.##";A2(
X);
2860 IF A2(X)=0 THEN 2870 ELSE 2
880
2870 PRINT#-2,TAB(59)"#";
2880 PRINT#-2,TAB(68)"";
2890 PRINT#-2,USING"$###.##";A(X
)-A2(X)
2900 NEXTX
2910 PRINT#-2,STRING$(80,"-")
2920 PRINT#-2,TAB(5)"TOTAL";
2930 PRINT#-2,TAB(30)"";
2940 PRINT#-2,USING"$####.##";TA
```

Listing continued

SPEED LIMIT FAST!

FAST! DATA I/O PRINTING DELIVERY

8010X INTERFACE

$29.95
$2 SHIPPING

110~9600 BAUD
CABLE INCLUDED

RAINBOW CERTIFICATION SEAL

GEMINI-10X WITH 8010X INTERFACE

$319.95 (PLUS $10 SHIPPING AND INSURANCE)

CABLE AND SCREEN PRINT SOFTWARE INCLUDED
READY TO PLUG IN/NOTHING MORE TO BUY

↙73

DAYTON ASSOCIATES • 7201 CLAIRCREST BLDG. B
DAYTON, OHIO 45424 • (513) 236-1454

SUN SOFTWARE

P.O. Box 7498
Wilmington, DE 19803
(302) 764-8328

SUN REPORTER

ULTIMATE WORD-PROCESSOR

85/24 64/24 51/24 42/24 -- True Upper-Lowercase. Switch to any of formats by a Single Key Command
Full Width Screen Display -- What you see on the Screen is what you will get from the Printer
Full Screen Editing -- Single Key Commands for Insert, Delete and Change
Embedded Control Codes for -- Underline, variable Letter sizes Subscript, Superscript
Global Search for -- Replacement or Deletion
Simple Disk or Cassette I/O
Merge files from Cassette or Disk

16K EXB Cassette / 29.95
32K EXB Disk / 34.95

SUN DISK MANAGER

Alphabetical Sorting - Filenames and Extension
Directory on the Screen - 10 at a time - or on the Printer
Single Key Commands for Rename, Delete, Run and Copy file
Save a Spare Directory Track and Recall it for an Emergency Routine for Cleaning Disk Head
16K or 32K Disk / 19.95

Do You Like **BRAHMS** ??
Famous Classic Musics by Computer - 16K EXB / 12.95

DISKETTES

3M 51/4" SS/DD $27.00
Head Cleaning Kit $22.00

We accept Visa, Master Card, Money Order or Personal Check
Please Add $2 for Shipping & Handling

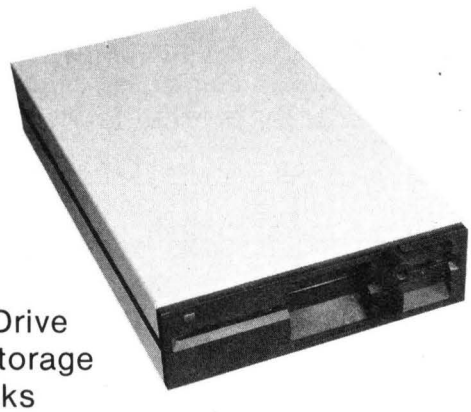
↙147

Micronix Systems Corporation
8147 Delmar
St. Louis, MO 63130
(314) 721-7969

Micronix Does It Again

Terms: Prepaid check or money order, Mastercard or Visa. Shipping charges US $3.00, COD $5.00, Canada $6.00.

High-speed, Accurate, Brushless DC Direct-Drive Motor. Most powerful and ideal Drive 1 for your CoCo.



- Direct-Drive
- 320K Storage
- 40 Tracks
- Low Noise
- Low Heat Generation

Incredible double sided, double density drive for only $284.00

↙203

40 COLUMN DOT MATRIX PRINTER

only $149.00

- 2-Color Printing (Black/Red)
- Using Normal Paper & Normal Ribbon
- 68 cps
- Friction Feed
- Graphic Printing
- Compatible to Epson Software Package
- Small-sized Design & Lightweight
- Centronics Interface

↙See List of Advertisers on page 130

Listing continued

```
;
2950 PRINT#-2,TAB(49)"";
2960 PRINT#-2,USING"$####.##";TP
;
2970 PRINT#-2,TAB(67)"";
2980 PRINT#-2,USING"$####.##";TA
-TP
2990 PRINT#-2,P2$;P2$
3000 GOTO 1510
3010 '
3020 '
3030 '     END PROGRAM RUN
3040 '
3050 '
3060 CLS:PRINT"PROGRAM ENDED"
3070 STOP:GOTO 1510
```

END

Program Listing 2. Medical Expenses

```
170 '     DIMENSION ARRAYS &
180 '     SET PRINTER CODES
190 '
200 '
210 CLS:CLEAR 1000
220 P1$=CHR$(10):P2$=CHR$(30):P3
$=CHR$(31):P4$=STRING$(5,10)
230 P5$=STRING$(32,"#"):P6$=STRI
NG$(80,"-")
240 DIM AV(27),AM(27)
250 DIM PI$(27),MP(27)
260 DIM DP$(27),PN$(27)
270 DIM IP(27),DT$(27)
280 DIM PT(27),MV$(27)
290 DIM DV$(27),PO$(27)
300 '
310 '
320 '     TITLE PAGE
330 '
340 '
350 PRINT@101,"MEDICAL EXPENSE P
ROGRAM"
360 PRINT@167,"BY DENNIS H. WEID
E"
370 PRINT:PRINTTAB(13)"(C)1982"
380 PRINT
390 PRINTTAB(5)"";:INPUT"WANT IN
STRUCTIONS(Y/N)";W$
400 IF W$="Y" THEN GOSUB 2090
410 '
420 '
430 '     PROGRAM MENU
440 '
450 '
460 CLS:T=6
470 PRINTTAB(4)"MEDICAL EXPENSE
INDEX"
480 PRINT:PRINTTAB(2)"1.ENTRY-ME
DICAL VISITS"
490 PRINTTAB(2)"2.ENTRY-MEDICINE
 PURCHASES"
500 PRINTTAB(2)"3.ENTRY-INSURANC
E PAYMENTS"
```

Listing continued

Listing continued

```
510 PRINTTAB(2)"4.SUMMARY-MEDICA
L VISIT"
520 PRINTTAB(2)"5.SUMMARY-MEDICI
NE PURCHASES"
530 PRINTTAB(2)"6.SUMMARY-INSURA
NCE PAYMENTS"
540 PRINTTAB(2)"7.SUMMARY-TOTAL
COST"
550 PRINTTAB(2)"8.MEDICAL EDITOR
 PROGRAM"
560 PRINTTAB(2)"9.LOAD DATA FROM
 CASSETTE"
570 PRINTTAB(2)"10.SAVE DATA TO
CASSETTE"
580 PRINTTAB(2)"11.HARDCOPY FROM
 PRINTER"
590 PRINTTAB(2)"12.END PROGRAM"
600 PRINT"   ENTER ONE OF THE AB
OVE";:INPUT X
610 ON X GOTO 680,820,970,1090,1
320,1540,1800,2220,2690,2930,323
0,3940
620 GOSUB 1980:GOTO 460
630 '
640 '
650 '     ENTRY-MEDICAL VISITS
660 '
670 '
680 CLS:MV=MV+1
690 PRINT:LINE INPUT"ENTER PATIE
NT'S NAME ";PI$(MV)
700 PRINT:LINE INPUT"ENTER DATE
OF VISIT ";DV$(MV):PRINT
710 PRINT"ENTER NATURE OF VISIT(
35 CHAR)"
720 LINE INPUT MV$(MV)
730 PRINT:INPUT"ENTER AMOUNT OF
VISIT";AV(MV)
740 GOSUB 4010
750 ON X GOTO 680,460
760 GOSUB 1980:GOTO 740
770 '
780 '
790 '     ENTRY-MEDICINE PURCHAS
ES
800 '
810 '
820 CLS:MP=MP+1
830 PRINT:LINE INPUT"ENTER PATIE
NT'S NAME ";IN$(MP):PRINT
840 LINE INPUT"ENTER DATE OF PUR
CHASE ";DP$(MP)
850 PRINT:LINE INPUT"ENTER PRESC
RIPTION  ";PN$(MP)
860 PRINT
870 INPUT"ENTER AMOUNT OF PURCHA
SE";AM(MP)
880 AM(MP)=AM(MP)+(AM(MP)*.04)
890 GOSUB 4010
900 ON X GOTO 820,460
910 GOSUB 1980:GOTO 890
920 '
930 '
940 '     ENTRY-INSURANCE PAYMEN
T
950 '
960 '
970 CLS:IP=IP+1
980 PRINT:LINE INPUT"ENTER PATIE
NT'S NAME ";PO$(IP)
990 PRINT:LINE INPUT"ENTER DATE
OF VISIT ";DT$(IP)
1000 PRINT:INPUT"ENTER AMOUNT OF
 PAYMENT";PT(IP)
1010 GOSUB 4010
1020 ON X GOTO 970,460
1030 GOSUB 1980:GOTO 1010
1040 '
1050 '
1060 '     SUMMARY-MEDICAL VISIT
S
1070 '
1080 '
1090 B1=1:B2=9
1100 CLS
1110 PRINTTAB(2)"1982 MEDICAL VI
SITS SUMMARY"
1120 PRINT:PRINT"DATE NAME  AMOU
NT   DIAGNOSIS"
1130 PRINT P5$
1140 GOSUB 1260:FOR X=B1 TO B2:G
OSUB 1170:NEXT X:GOSUB 1220
1150 IF MV<=B2 THEN 460
1160 B1=B1+9:B2=B2+9:GOTO 1100
1170 IF AV(X)<1 THEN 1210
1180 PRINT@A1,DV$(X);:PRINT@A2,L
EFT$(PI$(X),3);
1190 PRINT@A3,USING"$###.##";AV(
X);:PRINT@A4,LEFT$(MV$(X),12);
1200 A1=A1+32:A2=A2+32:A3=A3+32:
A4=A4+32
1210 RETURN
1220 AT=0:FOR X=1 TO 27:AT=AT+AV
(X):NEXT
1230 PRINT@423,"TOTAL COST=";:PR
INTUSING"$$####.##";AT
1240 PRINT@448,"";:LINE INPUT"
 PRESS <ENTER> TO CONTINUE";QW$
1250 CLS:RETURN
1260 A1=128:A2=133:A3=138:A4=146
:RETURN
1270 '
1280 '
1290 '     SUMMARY-MEDICAL PURCH
ASES
1300 '
1310 '
1320 B1=1:B2=9
1330 CLS:FOR Z=1 TO 50:NEXT
1340 PRINTTAB(1)"1982 MEDICINE P
URCHASE SUMMARY"
1350 PRINT:PRINT"DATE NAME  AMOU
NT PRESCRIPTION"
1360 PRINT P5$
1370 GOSUB 1260:FOR X=B1 TO B2:G
OSUB 1400:NEXT:GOSUB 1450
1380 IF MP<=B2 THEN 460
1390 B1=B1+9:B2=B2+9:GOTO 1330
1400 IF AM(X)<1 THEN 1440
1410 PRINT@A1,DP$(X);:PRINT@A2,L
```

Listing continued

Listing continued

```
EFT$(IN$(X),3);
1420 PRINT@A3,USING"$###.##";AM(
X);:PRINT@A4,LEFT$(PN$(X),12);
1430 A1=A1+32:A2=A2+32:A3=A3+32:
A4=A4+32
1440 RETURN
1450 AT=0:FOR X=1 TO 27:AT=AT+AM
(X):NEXT X
1460 PRINT@423,"TOTAL COST=";:PR
INTUSING"$$####.##";AT
1470 GOSUB 1240
1480 RETURN
1490 '
1500 '
1510 '    SUMMARY-INSURANCE PAY
MENT
1520 '
1530 '
1540 B1=1:B2=9:DD=0
1550 CLS:PRINTTAB(1)"1982 MEDICA
L INSURANCE SUMMARY"
1560 PRINT:PRINT"DATE NAME AMOUN
T INSURED  COST"
1570 PRINT P5$
1580 A1=128:A2=133:A3=137:A4=145
:A5=153
1590 FOR X=B1 TO B2:DD=0
1600 IF AV(X)<1THEN1710
1610 PRINT@A1,DV$(X):PRINT@A2,LE
FT$(PI$(X),3)
1620 PRINT@A3,USING"$###.##";AV(
X)
1630 FOR Y=1 TO 27
1640 IF DV$(X)=DT$(Y) AND PI$(X)
=PO$(Y) THEN 1660
1650 GOTO1680
1660 DD=1:PRINT@A4,USING"$###.##
";PT(Y)
1670 PRINT@A5,USING"$###.##";AV(
X)-PT(Y)
1680 NEXT Y
1690 IF DD=0 THEN PRINT@A4,USING
"$###.##";0:PRINT@A5,USING"$###.
##";AV(X)
1700 A1=A1+32:A2=A2+32:A3=A3+32:
A4=A4+32:A5=A5+32
1710 NEXT X
1720 GOSUB 1240
1730 IF IP<=B2 THEN 460
1740 B1=B1+9:B2=B2+9:GOTO 1550
1750 '
1760 '
1770 '    SUMMARY-TOTAL COST
1780 '
1790 '
1800 CLS
1810 TP=0:TI=0:TV=0
1820 FOR X=1 TO 27
1830 TP=TP+AM(X):TV=TV+AV(X):TI=
TI+PT(X):NEXT X
1840 PRINT P5$;:PRINTTAB(7)"TOTA
L MEDICAL COST":PRINT P5$;
1850 PRINT:PRINTUSING"COST OF SE
RVICES      $####.##";TV
1860 PRINT:PRINTUSING"COST OF PR
ESCRIPTIONS $####.##";TP
1870 PRINT:PRINTUSING"INSURANCE
PAYMENTS    $####.##";TI
1880 PRINT:PRINTUSING"COST TO EM
PLOYEE      $####.##";TV+TP-TI
1890 PRINT:PRINT:GOSUB 1240:GOTO
 460
1900 '
1910 '
1920 '    ERROR ROUTINES
1930 '
1940 '
1950 CLS:PRINT P5$:PRINTTAB(7)"Y
OU MUST LOAD TAPE"
1960 PRINTTAB(6)"BEFORE MAKING E
NTRIES":PRINT
1970 PRINT P5$:GOSUB 1240:GOTO 4
60
1980 FOR X=1 TO 10:CLS
1990 FOR T=1 TO 50:NEXT T
2000 PRINT@233,"ILLEGAL ENTRY":S
OUND 100,1
2010 FOR Y=1 TO 50:NEXT Y,X:CLS:
RETURN
2020 PRINTDV$(X)"   "PI$(X),AV(X
):RETURN
2030 PRINT:PRINT"   PRESS <ENTER
> TO CONTINUE";:INPUT ZZ:RETURN
2040 '
2050 '
2060 '    PROGRAM INSTRUCTIONS
2070 '
2080 '
2090 CLS:PRINTTAB(9)"INSTRUCTION
S":PRINT
2100 PRINT"1. LOAD DATA FROM CAS
SETTE FIRST"
2110 PRINT"2. ENTER PATIENTS NAM
E"
2120 PRINT:PRINT"3. ENTER DATE U
SING 4 DIGITS"
2130 PRINT:PRINT"4. ENTER DATE A
S MONTH/DAY"
2140 PRINT:PRINT"5. ENTER AMOUNT
S WITHOUT $"
2150 PRINT:PRINT"6. FOLLOW OTHER
 INSTRUCTIONS"
2160 PRINT:GOSUB 1240:RETURN
2170 '
2180 '
2190 '    MEDICAL EDITOR
2200 '
2210 '
2220 CLS:PRINT P5$;:PRINTTAB(9)"
PROGRAM EDITOR":PRINT P5$;
2230 PRINT:PRINTTAB(4)"1.EDIT ME
DICAL VISIT"
2240 PRINT:PRINTTAB(4)"2.EDIT ME
DICINE PURCHASE"
2250 PRINT:PRINTTAB(4)"3.EDIT IN
SURANCE PAYMENT"
2260 PRINT:PRINTTAB(4)"4.EXIT ED
ITOR PROGRAM"
2270 PRINT:PRINT:INPUT"    ENTER
 ONE OF THE ABOVE";QW
```

Listing continued

Listing continued

```
2280 ON QW GOTO 2300,2390,2480,4
60
2290 GOSUB 1980:GOTO 2220
2300 GOSUB 2560:GOSUB 2570
2310 FOR X=1 TO 35:PRINTX:IF C1$
=PI$(X) AND C2$=DV$(X) THEN 2340
2320 NEXT X
2330 GOTO 2590
2340 GOSUB 2630
2350 PRINT:LINE INPUT"ENTER PATI
ENT'S NAME ";PI$(X)
2360 PRINT:LINE INPUT"ENTER DATE
 OF VISIT ";DV$(X)
2370 PRINT:LINE INPUT"ENTER NATU
RE OF VISIT ";MV$(X)
2380 PRINT:INPUT"ENTER AMOUNT OF
 VISIT ";AV(X):GOTO 460
2390 GOSUB 2560:GOSUB 2570
2400 FOR X=1 TO 27:IF C1$=IN$(X)
 AND C2$=DP$(X) THEN 2430
2410 NEXT X
2420 GOTO 2590
2430 GOSUB 2630
2440 PRINT:LINE INPUT"ENTER PATI
ENT'S NAME ";IN$(X)
2450 PRINT:LINE INPUT"ENTER DATE
 OF PURCHASE ";DP$(X)
2460 PRINT:LINE INPUT"ENTER PRES
CRIPTION NAME ";PN$(X)
2470 PRINT:INPUT"ENTER AMOUNT OF
 PRESCRIPTION ";AM(X):GOTO 460
2480 GOSUB 2560:GOSUB 2570
2490 FOR X=1 TO 27:IF C1$=PO$(X)
 AND C2$=DT$(X) THEN 2520
2500 NEXT X
2510 GOTO 2590
2520 GOSUB 2630
2530 PRINT:LINE INPUT"ENTER PATI
ENT'S NAME ";PO$(X)
2540 PRINT:LINE INPUT"ENTER DATE
 OF VISIT ";DT$(X)
2550 PRINT:INPUT"ENTER AMOUNT OF
 INSURANCE ";PT(X):GOTO 460
2560 CLS:PRINT P5$:PRINTTAB(9)"E
NTER OLD DATA":PRINT P5$:RETURN
2570 PRINT:LINE INPUT"ENTER PATI
ENT'S NAME ";C1$
2580 PRINT:LINE INPUT"ENTER DATE
 OF PURCHASE OR VISIT ";C2$:RETU
RN
2590 CLS:PRINT:PRINT:PRINT P5$
2600 PRINTTAB(9)"NO DATA MATCH"
2610 PRINT P5$:FOR Y=1 TO 25:SOU
ND 1,1
2620 NEXT Y:GOTO 460
2630 CLS:PRINT P5$:PRINTTAB(9)"E
NTER NEW DATA":PRINT P5$:RETURN
2640 '
2650 '
2660 '    LOAD DATA FROM CASSET
TE
2670 '
2680 '
2690 IF ZQ=1 THEN 3120
2700 IF MV>0 OR MP>0 OR IP>0 THE
N 1950 ELSE 2710
2710 CLS:PRINT P5$;
2720 PRINTTAB(11)"LOAD DATA":PRI
NT P5$
2730 PRINT:PRINTTAB(7)"1. LOAD D
ATA TAPE"
2740 PRINT:PRINTTAB(7)"2. PRESS
PLAY ONLY"
2750 PRINT:PRINTTAB(7)"3. PRESS
ENTER"
2760 LINE INPUT ZZ$
2770 CLS:PRINT:PRINT:PRINT:PRINT
 P5$;
2780 PRINTTAB(6)"YOU ARE LOADING
 DATA":PRINT P5$
2790 OPEN"I",-1,"NEW"
2800 INPUT#-1,MV,MP,IP
2810 FOR X=1 TO MV
2820 INPUT#-1,MV$(X),DV$(X),AV(X
),PI$(X):NEXT X
2830 FOR X=1 TO MP
2840 INPUT#-1,DP$(X),PN$(X),AM(X
),IN$(X):NEXT X
2850 FOR X=1 TO IP
2860 INPUT#-1,DT$(X),PT(X),PO$(X
):NEXT X
2870 CLOSE#-1:ZQ=1:GOTO 460
2880 '
2890 '
2900 '    SAVE DATA TO CASSETTE
2910 '
2920 '
2930 N3=N3+N1
2940 N4=N4+N2
2950 CLS:PRINT P5$;
2960 PRINTTAB(11)"SAVE DATA":PRI
NT P5$
2970 PRINT:PRINTTAB(7)"1. LOAD D
ATA TAPE"
2980 PRINT:PRINTTAB(7)"2. PRESS
RECORD & PLAY"
2990 PRINT:PRINTTAB(7)"3. PRESS
ENTER"
3000 LINE INPUT ZZ$
3010 CLS:PRINT:PRINT:PRINT:PRINT
 P5$;
3020 PRINTTAB(6)"YOU ARE SAVING
DATA":PRINT P5$
3030 OPEN"O",-1,"NEW"
3040 PRINT#-1,MV,MP,IP
3050 FOR X=1 TO MV
3060 PRINT#-1,MV$(X),DV$(X),AV(X
),PI$(X):NEXT X
3070 FOR X=1 TO MP
3080 PRINT#-1,DP$(X),PN$(X),AM(X
),IN$(X):NEXT X
3090 FOR X=1 TO IP
3100 PRINT#-1,DT$(X),PT(X),PO$(X
):NEXT X
3110 CLOSE#-1:GOTO 460
3120 CLS:PRINT:PRINT
3130 PRINTTAB(6)"DATA ALREADY LO
ADED":PRINT
3140 PRINTTAB(2)"DO YOU WANT TO
```

Listing continued

Listing continued

```
RELOAD? (Y/N)";
3150 FOR II=1 TO 10:SOUND 100,3:
NEXT
3160 LINE INPUT KK$
3170 IF KK$="Y" THEN 2710 ELSE 4
60
3180 '
3190 '
3200 '     HARD COPY FROM PRINTE
R
3210 '
3220 '
3230 CLS:PRINT:PRINT
3240 PRINT P5$;:PRINTTAB(11)"PRI
NTING":PRINT P5$
3250 PRINT#-2,P1$;P3$;STRING$(40
,"#");P1$
3260 PRINT#-2,TAB(8)"MEDICAL EXP
ENSE RESULTS";P1$
3270 PRINT#-2,STRING$(40,"#")
3280 PRINT#-2,P1$;P1$
3290 PRINT#-2,TAB(9)"SUMMARY OF
SERVICES"
3300 PRINT#-2,P1$;P2$:GOSUB 4020
3310 PRINT#-2,TAB(55)"DIAGNOSIS"
3320 PRINT#-2,P6$
3330 FOR X=1 TO MV
3340 PRINT#-2,TAB(5);DV$(X);TAB(
16);PI$(X);
3350 PRINT#-2,TAB(31)"";
3360 PRINT#-2,USING"$####.##";AV
(X);
3370 PRINT#-2,TAB(45);LEFT$(MV$(
X),35)
3380 NEXT X
3390 AT=0:FOR X=1 TO 27
3400 AT=AT+AV(X):NEXT X
3410 PRINT#-2,P6$
3420 GOSUB 4040
3430 PRINT#-2,P4$
3440 PRINT#-2,P3$;TAB(10)"SUMMAR
Y OF PURCHASES"
3450 PRINT#-2,P2$;P1$:GOSUB 4020
3460 PRINT#-2,TAB(55)"MEDICINE P
URCHASE"
3470 PRINT#-2,P6$
3480 FOR X=1 TO MP
3490 PRINT#-2,TAB(5);DP$(X);TAB(
16);IN$(X);
3500 PRINT#-2,TAB(31)"";
3510 PRINT#-2,USING"$####.##";AM
(X);
3520 PRINT#-2,TAB(45);LEFT$(PN$(
X),35)
3530 NEXT X
3540 AT=0:FOR X=1 TO 27:AT=AT+AM
(X):NEXT X:MC=AT
3550 PRINT#-2,P6$
3560 GOSUB 4040
3570 PRINT#-2,P4$
3580 PRINT#-2,P3$;TAB(10)"SUMMAR
Y OF INSURANCE"
3590 PRINT#-2,P2$;P1$:GOSUB 4020
3600 PRINT#-2,TAB(45)"INSURANCE
PMT";
3610 PRINT#-2,TAB(63)"EMPLOYEE C
OST"
3620 PRINT#-2,P6$
3630 AT=0:CT=0:EC=0
3640 FORX=1 TO MV:Z=0
3650 PRINT#-2,TAB(5);DV$(X);TAB(
16);PI$(X);
3660 PRINT#-2,TAB(31)"";:PRINT#-
2,USING"$####.##";AV(X);
3670 AT=AT+AV(X)
3680 FOR Y=1 TO IP
3690 IF DV$(X)=DT$(Y) AND PT$(X)
=PO$(Y) THEN 3700 ELSE 3730
3700 Z=1:PRINT#-2,TAB(47);:PRINT
#-2,USING"$####.##";PT(Y);
3710 PRINT#-2,TAB(64)"";:PRINT#-
2,USING"$####.##";AV(X)-PT(Y)
3720 CT=CT+PT(Y):EC=EC+(AV(X)-PT
(Y))
3730 NEXT Y
3740 IF Z=1 THEN 3790 ELSE 3750
3750 PRINT#-2,TAB(47)"";:PRINT#-
2,USING"$####.##";0;
3760 PRINT#-2,TAB(56)"";:PRINT#-
2,"#";
3770 PRINT#-2,TAB(64)"";:PRINT#-
2,USING"$####.##";AV(X)
3780 EC=EC+AV(X)
3790 NEXT X
3800 PRINT#-2,P6$
3810 GOSUB 4040
3820 PRINT#-2,TAB(47)"";:PRINT#-
2,USING"$####.##";CT;
3830 PRINT#-2,TAB(64)"";:PRINT#-
2,USING"$####.##";EC
3840 PRINT#-2,P3$;STRING$(2,10)
3850 PRINT#-2,TAB(5)"TOTAL MEDIC
AL COST";
3860 PRINT#-2,TAB(24)"";:PRINT#-
2,USING"$$####.##";EC+MC
3870 PRINT#-2,P2$;P4$
3880 GOTO 460
3890 '
3900 '
3910 '     END PROGRAM
3920 '
3930 '
3940 CLS:PRINT"PROGRAM STOPPED"
3950 STOP:GOTO 460
3960 '
3970 '
3980 '     SUBROUTINES
3990 '
4000 '
4010 PRINT:INPUT"1=ENTRY 2=INDEX
   WHICH ONE";X:CLS:RETURN
4020 PRINT#-2,TAB(5)"DATE";TAB(1
7)"NAME";
4030 PRINT#-2,TAB(32)"CHARGE";:R
ETURN
4040 PRINT#-2,TAB(10)"TOTAL";TAB
(31)"";
4050 PRINT#-2,USING"$####.##";AT
;:RETURN
```

END

of the hill!

It's a jungle out there, but the latest news on the Color Computer grapevine is that, above the swirling mists of confusion, more and more people are discovering the Rainbow.®

Now in its third year, the Rainbow has become the standard by which all other Color Computer magazines are compared. And no wonder! The Rainbow towers above the crowd, now offering more than 300 pages each month, including more than two dozen type-in-and-run program listings, a host of articles and in excess of 30 hardware and software product reviews.

We lead the pack in Color Computer publications and are devoted *exclusively* to the TRS-80® Color, TDP-100 and Dragon-32. We made our climb to the top by continually offering the best and the most by such well-known authors and innovators as Bob Albrecht and Don Inman, and games from top programmers like Chris Latham, Fred Scerbo and John Fraysse. The Rainbow offers the most in entertainment and education, home uses, technical details and hardware projects, tutorials, utilities, graphics and special features like Rainbow Scoreboard and our new CoCo Clubs section.

For only $28 a year, you get the keys to all the secrets locked in your CoCo!

Are you searching through the jungle of claims and clamor? Climb above it all. Look up. Find the Rainbow.

---

the Rainbow
502/228-4492

9529 U.S. Highway 42
P.O. Box 209
Prospect, Ky. 40059

YES! Sign me up for a year (12 issues) of the RAINBOW

Name _____

Address _____

City _____ State _____ Zip _____

☐ Payment Enclosed

Charge ☐ VISA ☐ MasterCard ☐ American Express

My Account# _____ Interbank# (MC only) _____

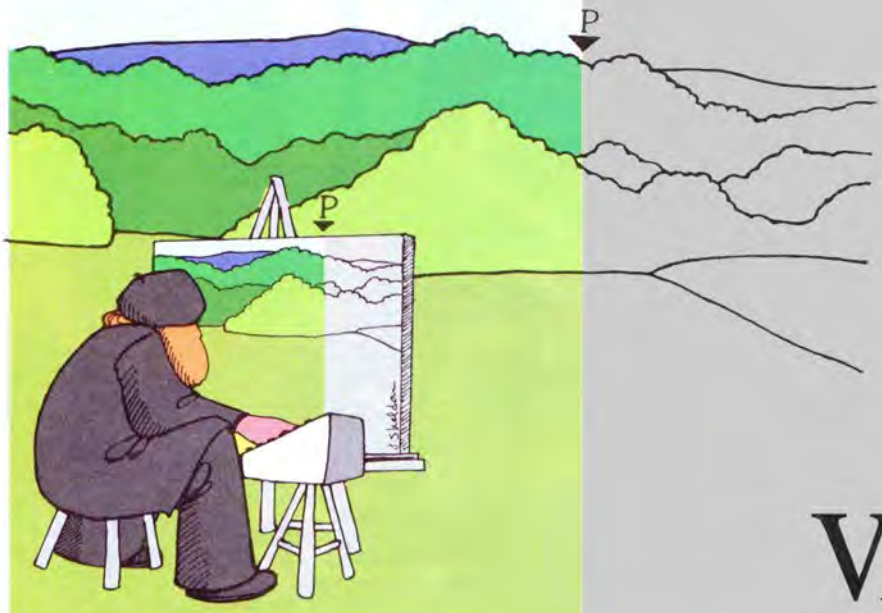
Signature _____ Card Expiration Date _____

Subscriptions to **the RAINBOW** are $28 a year in the United States. Canadian and Mexican rate U.S. $35. surface rate to other countries U.S. $65; air rate U.S.$100. All subscriptions begin with the current issue. Please allow up to 5-6 weeks for first copy.

Just for your TRS-80® COLOR TDP System 100 Dragon-32 MC-10

✓296

BY ERIC EINEM



VIDEO VAN GOGH

You don't have to be a programmer to bring out your artistic creativity with the CoCo's graphics. Type in the Program Listing, Color Computer Artist, and you can design a graphics screen or copy an existing one.

This program takes advantage of most of the computer's graphics commands such as LINE, CIRCLE, PAINT, GET, and PUT.

Color Computer Artist has 29 commands to draw pictures. You won't need to memorize all of them since you can list them by hitting the ? key. The first command set is for movement. The arrows move the cursor up, down, left, and right. Q, W, A, and S move the cursor at angles.

Express your artistic creativity using this program and the graphics capabilities of the CoCo.

You can change the distance the cursor moves by hitting D followed by a number zero through nine for the number of dots the cursor moves. A zero is equal to a distance of 10 dots.

To place a dot on the screen, push the space bar. To draw a line, push L and the program draws a line from the last dot set to the cursor position.

To change the color you draw with, push C followed by a number one through eight for the color you want. To draw a circle, hit the space bar where you want the edge of the circle, then hit 0 where you want the center. If you want to draw a box, hit the space bar for one corner of the box, then hit the B for the opposite corner. If you push the hyphen, the program draws a line wherever you move the cursor without having to hit L. To return to normal mode hit the period key.

If you want to use the joysticks push J and use both joysticks to move the cursor around the screen. The right joystick moves the cursor across the entire screen. The left joystick moves 63 dots horizontally or vertically and can be used for more exact movement. To return to keyboard movement, push K.

To clear the screen push the clear key twice. This avoids accidentally clearing the screen. To paint in an area, locate the cursor over the area you want to paint, push P followed by the number of the color you want to use, and the number of the border color.

To GET an image from the screen, push enter at the upper left corner of the image, and push G at the lower right corner. To PUT the image back on the screen push R as in replace (since the P is used for paint) where you want the upper left corner to be.

If you want to change modes, push M followed by a number zero through four. When you want to change color sets, push N to switch between color set zero and color set one. To change graphic pages, push " " to advance one page, and " " to go back one page. When you want to save a drawing or load one in, push F. A menu appears and asks if you want to save a picture,

X,Y—location of cursor
X1,Y1—location of last dot drawn
C—color
D—distance cursor moves
PG—graphics page being used
CS—color set
MD—graphics mode being used
MP—maximum pages that can be used
P(n)—for storing with GET
A,B—temporary variables
I$—used with INKEY$
I—location of I$ in C$
X2,Y2—size of image stored with GET
J—flag, 1 = joystick mode, 0 = keyboard mode
M—flag, 1 = continuous line mode, 0 = normal mode
C$—string of commands
P—keeps track of the color at the cursor location

Table 1. Variable List

System Requirements

16K RAM
Extended Color Basic

TOM MIX SOFTWARE

TELEX 706139

✓177 •FOR THE COLOR COMPUTER & TDP 100 • 3424 College N.E., Grand Rapids, MI 49505 **(616) 364-4791**•

CU*BER

32K Mach. Lang.
$27.95 TAPE
$30.95 DISK

Approaches the excitement and challenges of any Video Arcade. The hazards of CU*BER are many. Help CU*BER change the colors on the pyramid while avoiding many of the dangers always present. Vipers, the Nurd, the Dork, bonus points all add up to another exciting release from Tom Mix Software.

Arcade Action. Method of play you are the Grabber. The object is to grab the 8 treasures and store them in the center boxes. You start with 3 Grabbers and get extra ones at 20,000 points. Watch out for the googlies! Super high resolution graphics.

32K Machine Language

$27.95 TAPE **$30.95 DISK**

AIR TRAFFIC CONTROLLER

32K Ext. Basic
$28.95 TAPE
$31.95 DISK

Air Traffic Controller is a computer model of an air traffic control situation in which Remotely Piloted Vehicles (RPV's) are operated by the controller in landing on and taking off from designated runways.

DEVIL ASSAULT

16K Machine Language
$27.95 TAPE
$30.95 DISK

Devil Assault is a multi-level multi-screen game in which bird-like creatures, robots and the devil himself assault your home base which you must defend.

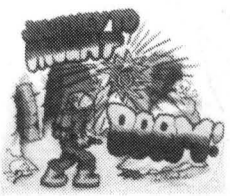
BUZZARD BAIT

By RUGBY CIRCLE

32K Machine Language
$27.95 Tape $30.95 Disk

We've done it again! You thought the King was great? wait 'till you see this!! Outstanding high resolution graphics, tremendous sound make this "Joust" type game a must for your software collection. As you fly from cloud to cloud you will enjoy sky high excitement dealing with the challenges presented to you by this newest release by Tom Mix Software.

JOURNEY TO MT. DOOM



32K Mach. Lang.
$27.95 DISK ONLY

The Necromancer is about to wage war on earth. He needs his lost gold ring to acquire the power to do so. You must find the ring, take it to Mt. Doom and destroy it in the flames from which it came, thus eliminating the Necromancer's evil powers.

"THE FROG"

*****ARCADE ACTION*****

This one will give you hours of exciting play. . . Cross the busy highway to the safety of the median and rest awhile before you set out across the swollen river teaming with hidden hazards. Outstanding sound and graphics.

16K MACHINE LANGUAGE
$27.95 TAPE
$30.95 DISK

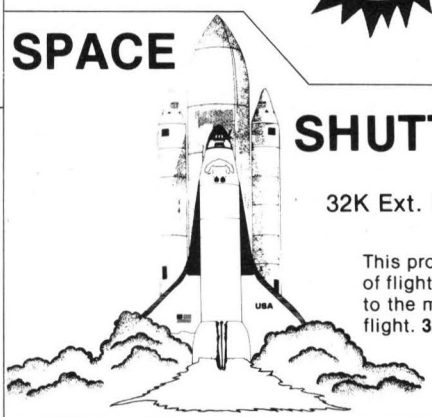
JUNIOR'S REVENGE

Climb vines, avoid obstacles & creatures to save your father from Luigi.

32K CASS $28.95
32K DISK $31.95

NEW

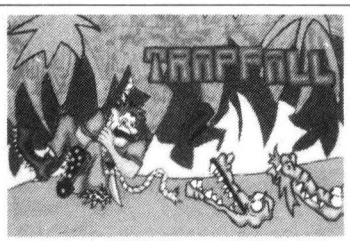
SPACE SHUTTLE



$28.95 TAPE
$31.95 DISK

32K Ext. Basic

This program gives you the real feeling of flight. Full instrumentation complete to the max. Actual simulation of space flight. **32K Extended Basic**



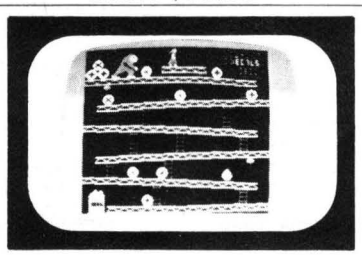
16K MACHINE LANGUAGE
TAPE $27.95
DISK $30.95

"TRAPFALL"

By KEN KALISH

*****ARCADE ACTION*****

The "Pitfalls" in this game are many. Hidden treasures, jump over the pits, swing on the vine, watch out for alligators, beware of the scorpion. Another game for the Color Computer with the same high resolution graphics as "The King."



THE KING

32K Machine Language
$26.95 TAPE
$29.95 DISK

ARCADE ACTION - How high can you climb? Four full graphic screens. Exciting Sound - Realistic graphics. Never before has the color computer seen a game like this. Early reviews say: Just like the arcade Simply outstanding!

OTHER GREAT GAMES

PROTECTORS - Exciting fast paced arcade game that looks and plays like the popular arcade game "DEFENDER".
32K Machine Code **Tape $24.95** **Disk $27.95**

COLOR GOLF - Now sit at your computer and play nine or eighteen holes. Outstanding graphics in the fairway or on the green. Helps your game.**32K Extended Basic** **$17.95**

'YAAZEE" (C) 1983 - Yaazee is a 2 player game using five dice to get the best poker hand. After game is loaded flashing digit below player number determines which player rolls dice at the start of the game. **16K Machine Language Ext. Basic** **$19.95**

BIRD ATTACK - A fast paced machine language arcade game. Shoot the birdmen before they descend upon you. Watch out for their bombs! **16K Machine Language** **$21.95**

MAZE RACE - Maze race is a one or two player game. Play either against the built in timer or against your favorite opponent. **16K Machine Code** **$17.95**

VISA MasterCard

Call our BBS Number
616-364-8217 24 Hours a Day

ADD $1.00 POSTAGE & HANDLING • TOP ROYALTIES PAID
MICHIGAN RESIDENTS ADD 4% SALES TAX • LOOKING FOR NEW SOFTWARE

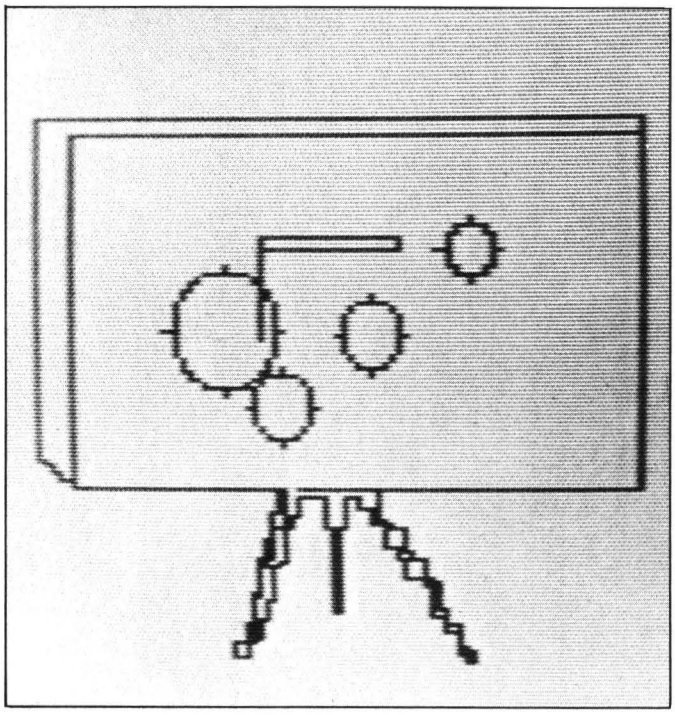


Photo 1. This black and white drawing was created in just a few minutes with Video Van Gogh.

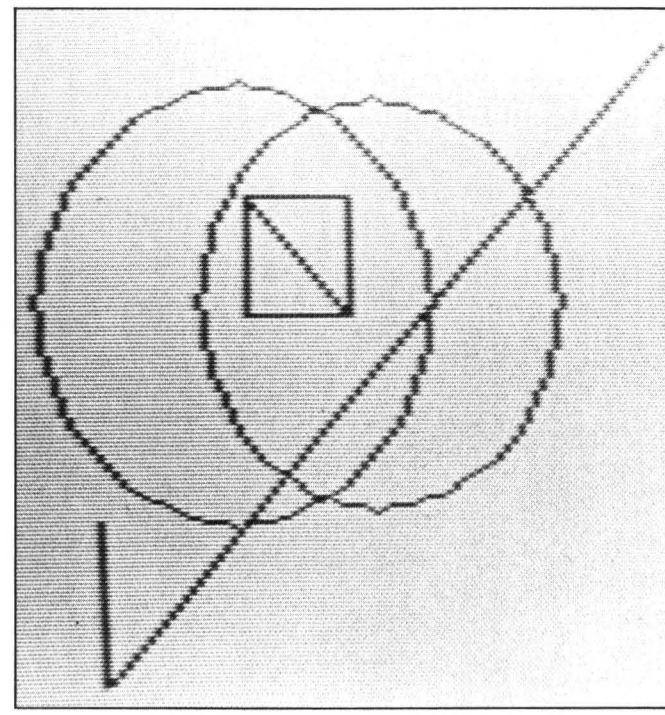


Photo 2. A simple drawing created quickly using the Line, Circle, and Box commands.

load in a picture, or exit the program.

If you want to copy pictures onto the screen, an easy way to do it is to use a sheet of clear plastic. First trace the picture onto the plastic, then tape the plastic onto the screen and copy the outline.

Before you run or save the program, make sure the first line of the program is exactly the same as shown in the listing. Then, type PCLEAR4:POKE7690,10; POKE7691,8:POKE7692,9:POKE7697, 12:POKE7698,13. This alters line 10 of the program by replacing all the asterisks in C$. C$ contains all the commands, some of which cannot be entered from the keyboard and need to be POKEd in. This change is permanent and will be saved with the program.

How the Program Works

To parse the commands efficiently the program takes advantage of the INSTR and the ON GOTO commands. When a key is pushed, it is searched for in C$, which contains all the commands, using INSTR. The INSTR command returns a number 0–29 where a zero means the key pushed was not found in C$ and is an invalid command.

If any other number is returned, the ON GOTO statement directs the program to the proper routine for the key pushed. To save space the program uses a subroutine at line 1000, which waits for a key to be pressed and stores it in I$. Then wherever the program needs to wait for a key I put a GOSUB 1000.

Now you can impress your friends with fancy graphics on your Color Computer. If you get frustrated trying to make recognizable scenery, try making abstract art. ■

Address correspondence to Eric Einem, 16985 Timber Ridge Drive, Granada Hills, CA 91344.

Program Listing. Color Computer Artist

```
10 C$="^***QAS** LCPOD-.JKFBGR?M
N<>"
 LCPOD-.JKFBGR?MN<>"
30 A$=CHR$(128):CLS0:PRINTSTRING
$(32,175);:FORA=32TO448STEP32:PR
INT@A,CHR$(175);:PRINT@A+31,CHR$
(175);:NEXTA:PRINT@480,STRING$(3
1,175);:POKE1535,175:PRINT@68,"t
he";A$;"color";A$;"computer";A$;
"artist";:PRINT@201,"by";A$;"eri
c";A$;"einem";:EXEC44539
40 DIMP(20,20):PMODE3,1:PCLS1
50 X=128:Y=96:C=4:D=4:MD=3:CS=0:
PG=1:CLS:INPUT"HOW MANY PAGES AR
E PCLEARED ";PC:MP=PC-3:SCREEN1,
CS
60 P=PPOINT(X,Y):PSET(X,Y):COLOR
1,1:PSET(X,Y):COLOR2,1:PSET(X,Y)
:COLORP,1:PSET(X,Y):COLORC,1
70 IFJ=1ANDM=0THENX=INT(JOYSTK(0
)+JOYSTK(2)*3.05):Y=INT(JOYSTK(1
)+JOYSTK(3)*2.04)
80 IFJ=1ANDM=1THENA=INT(JOYSTK(0
)+JOYSTK(2)*3.05):B=INT(JOYSTK(1
)+JOYSTK(3)*2.04):LINE(X,Y)-(A,B
),PSET:X=A:Y=B
90 I$=INKEY$:IFI$=""THEN60
100 I=INSTR(1,C$,I$):IFI=0THENPL
AY"T100V31O2CAV25CAV20CAV15CAV10
CAV5CA":GOTO60
110 B=0:A=0:ONI GOTO130,140,150,
160,170,180,190,200,290,320,340,
360,380,420,460,470,500,510,520,
530,540,600,620,640,660,720,780,
810,820
120 '***** KEYBOARD MOVEMENT***
130 B=-1:GOTO220
140 B=1:GOTO220
150 A=-1:GOTO220
160 A=1:GOTO220
170 A=-1:B=-1:GOTO220
180 A=1:B=-1:GOTO220
190 A=-1:B=1:GOTO220
200 A=1:B=1
210 '***** OFF SCREEN ? ******
220 IFX+A*D>255THENX=0:A=0:B=0EL
SEIFX+A*D<0THENX=255:A=0:B=A
230 IFY+B*D>191THENY=191:A=0:B=0
ELSEIFY+B*D<0THENY=0:A=0:B=A
240 'IF IN LINE MODE DRAW LINE
250 IFM=1THENLINE(X,Y)-(X+A*D,Y+
B*D),PSET
260 'MOVE CURSOR TO NEW POSITION
270 X=X+A*D:Y=Y+B*D:GOTO60
280 '***** CLEAR SCREEN *****
290 GOSUB1000
300 IFASC(I$)=12THENPCLSC:GOTO60
ELSE60
310 '*** STORE POINT FOR GET***
320 X1=X:Y1=Y:GOTO60
330 '***** SET POINT ********
340 PSET(X,Y):X1=X:Y1=Y:GOTO60
350 '***** DRAW LINE ********
360 LINE(X,Y)-(X1,Y1),PSET:X1=X:
Y1=Y:GOTO60
370 '***** CHANGE COLOR *****
380 GOSUB 1000
390 IFVAL(I$)=0ORVAL(I$)=9THENI$
="0":GOTO100
400 C=VAL(I$):GOTO60
410 '***** PAINT *******
420 GOSUB1000:A=VAL(I$):IFA=0ORA
=9THENI$="!":GOTO100
430 GOSUB1000:B=VAL(I$):IFB=0ORB
=9THENI$="!":GOTO100
```

Listing continued

Listing continued

```
440 PAINT(X,Y),A,B:GOTO60
450 '****** DRAW CIRCLE *******
460 A=INT(SQR((X-X1)^2+(Y-Y1)^2)
):CIRCLE(X,Y),A,C:GOTO60
470 GOSUB 1000
480 'CHANGE DISTANCE CURSOR MOVE
S
490 D=VAL(I$):IFD=0THEND=10:GOTO
60ELSE60
500 M=1:GOTO60 :'CONTINUOUS LINE
 MODE
510 M=0:GOTO60 :'NORMAL MODE
520 J=1:GOTO60 :'JOYSTICK MODE
530 J=0:GOTO60 :'KEYBOARD MODE
540 CLS:PRINT"1 - SAVE PICTURE":
PRINT"2 - LOAD PICTURE":PRINT"3
 - WORK ON PICTURE":PRINT"4 - EXI
T":INPUT"YOUR CHOICE ";A:ONA GOT
O550,560,580,570
550 LINEINPUT"NAME OF PICTURE";N
$:INPUT"STARTING PAGE, ENDING PA
GE ";S,E:CSAVEMN$,S*1536,(E+1)*1
536-1,0:GOTO540
560 LINEINPUT"NAME OF PICTURE :"
;N$:CLOADMN$:GOTO540
570 PRINT"GOOD BYE.":END
580 SCREEN1,CS:GOTO60
590 '***** DRAW BOX *******
600 LINE(X,Y)-(X1,Y1),PSET,B:X1=
X:Y1=Y:GOTO60
610 '***** GET IMAGE *******
620 GET(X1,Y1)-(X,Y),P:X2=X-X1:Y
2=Y-Y1:PLAY"V16L8CCL16DE":GOTO60
630 '***** REPLACE IMAGE *****
640 PUT(INT(X/2)*2,INT(Y/2)*2)-(
INT(X/2)*2+X2,INT(Y/2)*2+Y2),P:G
OTO60
650 '***** LIST COMMANDS *****
660 CLS:PRINTTAB(12)"COMMANDS":P
RINT"ARROWS, QWAS - MOVE CURSOR.
":PRINT"L - DRAW LINE FROM LAST
POINT       DRAWN TO PRESENT POS
ITION.  C - FOLLOWED BY # TO CHA
NGE         COLOR.":PRINT"D - FO
LLOWED BY # TO CHANGE          DI
STANCE CURSOR MOVES."
670 PRINT"O - DRAWS CIRCLE, HIT
SPACE FOR     EDGE, PRESENT POS.
 IS CENTERB - DRAWS BOX. HIT SPA
CE FOR ONE    CORNER, PRESENT PO
S. IS         OTHER CORNER.":PRI
NT"- - CONTINUOUS LINE MODE.":PR
INT". - NORMAL MODE (DRAWS LINE
ONLY    WHEN L IS HIT).";
680 GOSUB1000:CLS:PRINTTAB(12)"C
OMANDS":PRINT"J - JOYSTK MODE.":
PRINT"K - KEY BOARD MODE.":PRINT
"P - FOLLOWED BY PAINT #, BORDER
     # TO PAINT AT PRESENT POS."
:PRINT"CLEAR - HIT 2X TO CLEAR S
CREEN."
690 PRINT"G - GET, HIT ENTER FOR
 UPPER        LEFT CORNER, PRESE
NT POSIT.     IS LOWER RIGHT COR
NER.":PRINT"R - REPLACE, PRESENT
 POSITION IS    UPPER LEFT CORNE
R.":PRINT"M - FOLLOWED BY # TO C
HANGE MODEN - SWITCHES BETWEEN C
OLOR SETS."
700 PRINT"<,> - CHANGE PAGE.":PR
INT"F - FINISH":PRINT"? - LIST C
OMMANDS.";:GOSUB1000:SCREEN1,CS:
GOTO60
710 '***** CHANGE MODE ******
720 GOSUB1000:IFVAL(I$)>4THENPLA
YE$:GOTO60
730 MD=VAL(I$):IFMD=0THENMP=8ELS
EIFMD=1ORMD=2THENMP=7ELSEMP=5
740 MP=MP-8+PC
750 IFPG>MP THENPG=MP
760 PMODEMD,PG:SCREEN1,CS:GOTO60
770 '***** CHANGE COLOR SET ***
780 IFCS=0THENCS=1ELSECS=0
790 SCREEN1,CS:GOTO60
800 '***** CHANGE PAGE *****
810 PG=PG-1:IFPG<1THENPG=1:GOTO8
30ELSEGOTO830
820 PG=PG+1:IFPG>MP THENPG=MP
830 PMODEMD,PG:SCREEN1,CS:GOTO60
990 '**** INKEY$ SUBRUTINE ****
1000 I$=INKEY$:IFI$=""THEN1000
1010 RETURN
```

END

See List of Advertisers on page 130

BY KENNETH L. WUELZER

K-BUG

This helpful utility lets you list your Basic programs byte by byte along with line numbers.

This utility for Basic programs, called K-Bug (Program Listing 1), lists each line of a program and corresponding byte in memory. K-Bug interprets control codes and displays meaning for each memory location as well as the Basic line number. (See Fig. 1.) This is useful in changing a variable in a subroutine with a POKE to avoid duplicating lines of code.

You can also use it to add a measure of security to a program. For example, you can insert lines in a program that prevent it from working. The first character of these bogus lines can then be POKEd with an apostrophe to make them comment lines, thus ignored when the program runs.

To further complicate things, try PEEKing values from credits and POKEing those values into the program at crucial places. The resulting Basic program is no longer relocatable. This means that if you remove or change credits, the rest of the memory locations shift to close the gaps, putting all such POKEs within the program in the wrong places. K-Bug

```
10000 CLS3:CLEAR2000:PRINT@34,"C
OLOR COMPUTER CONTROL CODES";:PR
INT@100,"FOR EXTENDED DISK BASIC
";:PRINT@170,"WRITTEN BY";:PRINT
@359,"KENNETH L WUELZER";:GOSUB1
0013
10001 CLS0:PRINT@10,"INSTRUCTION
S";:PRINT@64,"<1>  MERGE WITH PR
OGRAM":PRINT@128,"<2>  RUN 10000
"
10002 LL=0:L=1:A=PEEK(25):B=PEEK
(26):ST=A*256+B-1
10003 PRINT@256,"START AT BEGINN
ING OF PROGRAM ?";:GOSUB10013:IF
W$<>"N"THEN10005ELSEPRINT@320,"S
TART <A>DRESS OR <L>INE NUMBER";
:GOSUB10013:IFW$="A"THENPRINT@38
4,"";:INPUT"START ADDRESS (HEX)"
;W$:W$="&H"+W$:ST=VAL(W$):GOTO10
005
10004 PRINT@384,"";:INPUT"STARTI
NG LINE NUMBER";LL
10005 CLS3:PRINT@12,"FORMAT";:PR
INT@64," MEM   VAL    LINE MEANI
NG":DIMCH$(600):OPEN"D",#1,"K-BU
G.DAT",16:FIELD#1,8 AS CH$:FORX=
1TO127:CH$(X)=CHR$(X):NEXT:FORX=
128TO224:Y=X-127:GET#1,Y:INPUT#1
,CH$(X):NEXT:FORX=383TO422:Y=X-1
27
10006 GET#1,Y:INPUT#1,CH$(X):NEX
T:CLOSE#1
10007 FORX=ST TO65535:P=PEEK(X):
IFZZ<>0THENZZ=0:NEXTX
10008 IFP<32 AND PEEK(X+3)<65THE
NL=PEEK(X+3)*256+PEEK(X+4):X=X+4
:NEXTX
10009 IFP=255THENP=255+PEEK(X+1)
10010 IFL>=LL THENPRINTHEX$(X);P
,L;CH$(P)
10011 IFP>254THENZZ=1
10012 NEXT
10013 W$=INKEY$:IFW$=""THEN10013
ELSERETURN
```

Program Listing 1. K-Bug

```
1 CLS3:PRINT@8,"K-BUG DATA LOADE
R";:DIMCH$(200):FORX=1TO137:READ
CH$(X):NEXT:OPEN"D",#1,"K-BUG.DA
T",16:FIELD#1,8 AS CH$:FORX=1TO9
7:WRITE#1,CH$(X):PUT#1,X:NEXT:FO
RX=98TO137:Y=158+X:WRITE#1,CH$(X
):PUT#1,Y:NEXT:PRINT@64,"DONE";:
STOP
2 DATAFOR,GO,REM,":REM",ELSE,IF,
DATA,PRINT,ON,INPUT,END,NEXT,DIM
,READ,RUN,RESTORE,RETURN,STOP,PO
KE,CONT,LIST,CLEAR,NEW,CLOAD,CSA
VE,OPEN,CLOSE,LLIST,SET,RESET,CL
S,MOTOR,SOUND,AUDIO,EXEC,SKIPF,T
AB(,TO,SUB,THEN,NOT,STEP,OFF,+,-
,*,/,^,AND,OR,>,=,<,DEL
3 DATAEDIT,TRON,TROFF,DEF,LET,LI
NE,PCLS,PSET,PRESET,SCREEN,PCLEA
R,COLOR,CIRCLE,PAINT,GET,PUT,DRA
W,PCOPY,PMODE,PLAY,DLOAD,RENUM,F
N,USING,DIR,DRIVE,FIELD,FILES,KI
LL,LOAD,LSET,MERGE,RENAME,RSET,S
AVE,WRITE,VERIFY,UNLOAD,DSKINI,B
ACKUP,COPY,DSKI$,DSKO$,SGN
4 DATAINT,ABS,USR,RND,SIN,PEEK,L
EN,STR$,VAL,ASC,CHR$,EOF,JOYSTK,
LEFT$,RIGHT$,MID$,POINT,INKEY$,M
EM,ATN,COS,TAN,EXP,FIX,LOG,POS,S
QR,HEX$,VARPTR,INSTR,TIMER,PPOIN
T,STRING$,CVN,FREE,LOC,LOF,MKN$,
AS
```

Program Listing 2. K-Bug Loader

System Requirements

16K RAM
Extended Color Basic

makes it easy to find the appropriate memory locations to POKE.

K-Bug is in two parts. The loader (Program Listing 2) creates the data file for the control codes and needs to be run only once. K-Bug is saved in ASCII to merge with the program to be listed. It starts with line 10000, so to avoid conflicts be sure that Basic program line numbers are less than 10000.

After merging, type RUN 10000. Options let you start listing at the beginning of a program, at a particular address, or at a specified line number (the slowest option).

The tape version is similar. Disk I/O commands need to be changed to tape, and merging K-Bug with a program is more involved. ■

Address correspondence to Ken Wuelzer, 752 West Main, Moore, OK 73160.

"You can also use it to add a measure of security to a program. For example, you can insert lines in a program that prevent it from working."

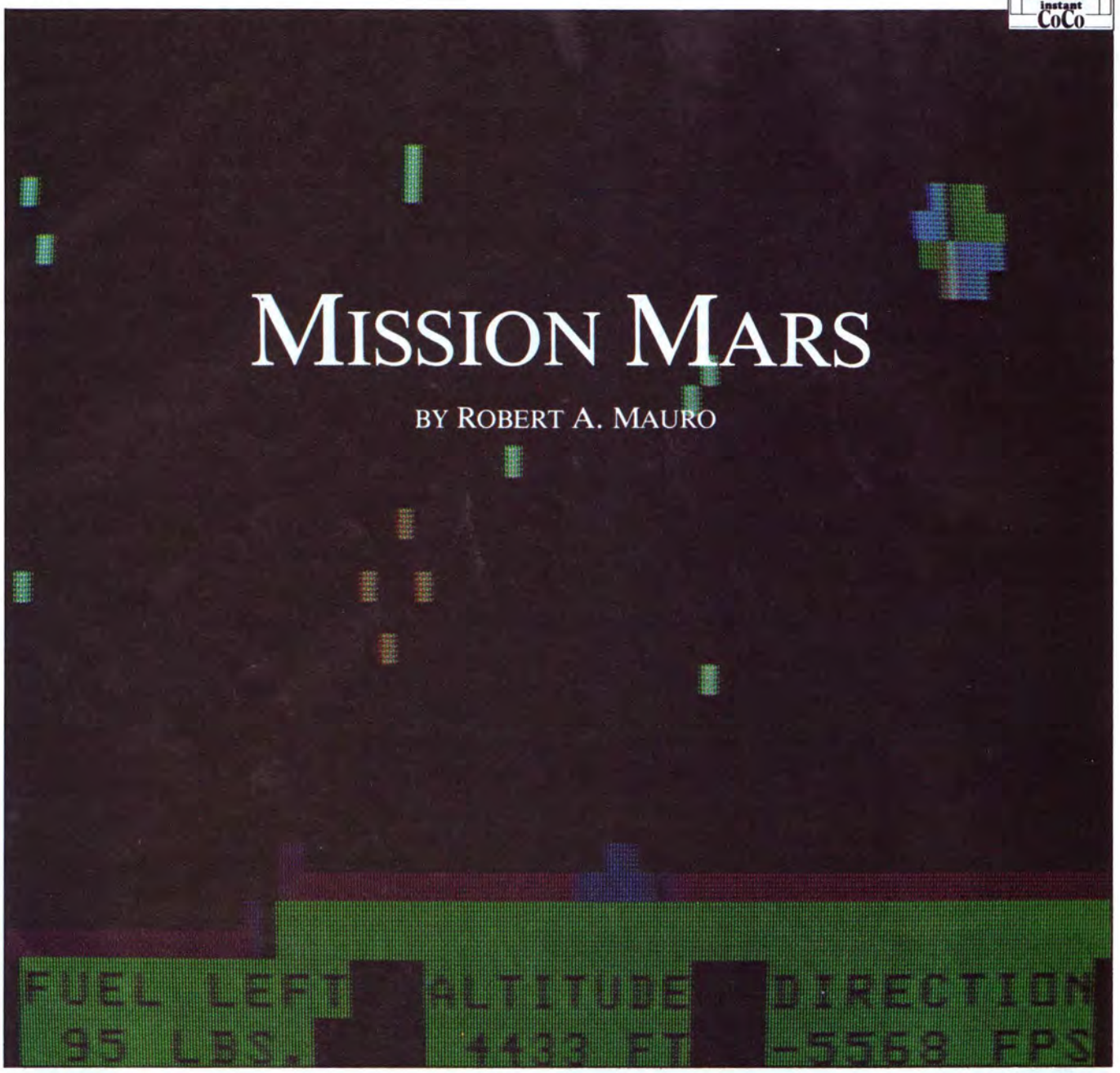
MEM	VAL	LINE	MEANING
2605	158	10000	CLS
2606	51	10000	3
2607	58	10000	:
2608	149	10000	CLEAR
2609	50	10000	2
260A	48	10000	0
260B	48	10000	0
260C	48	10000	0
260D	58	10000	:
260E	135	10000	PRINT
260F	64	10000	@
2610	51	10000	3
2611	52	10000	4
2612	44	10000	,
2613	34	10000	"
2614	67	10000	C
2615	79	10000	O
2616	76	10000	L

Fig. 1. Sample K-Bug Output

See List of Advertisers on page 130

MISSION MARS

BY ROBERT A. MAURO



What's it like on Mars? Do they have any good restaurants there? Find out in this 4K game.

You're about to go where no man or woman—not even Sally Ride—has gone before: Mars. But beware of the meteorites; they'll destroy you in a flash. And don't run out of fuel or you'll plummet to the Martian surface and crash.

Your mission is to stay alive and land safely on Mars' Blue Base, located in a crater 10,000 feet below, and moving east to west as the planet rotates beneath you. Remember, you must land before you run out of fuel.

If you're a hotshot space jock who laughs at danger and craves adventure, see how long you can remain in space. Dodge those meteorites and enjoy the view of Earth far off in the distance.

If you land, crash, or are hit by a meteorite, a flight timer tells you how long you flew. See if you can set a record for staying in orbit. And if you get homesick, or chicken out, you can abort your mission and head for home. To avoid a crash, be sure to land in the soft, blue Pacific Ocean and not on the hard green

System Requirements

4K RAM
Color or Micro Color (MC-10) Basic

Line	Notes
1–95	Introduction
110–210	Builds crater and Mars base
220–450	Sets operating parameters and gauge readouts
460–500	Creates Earth and meteorites
510–670	Control functions
680–860	Sound effects and final status reports
999–2018	Instructions

Table 1. Program Notes

continents. Navigation is very critical.

Your controls are keys 1–8. They fire your landing rockets. The higher the number the faster you can climb. To maintain near-level flight, keep pressing the 1 key, but watch that fuel gauge! If you don't press a key, you'll fall. Also, if you want to drop faster and faster, push the left-arrow key to refire your retros and plunge even more rapidly to the surface, leaving orange bits of your heat shield trailing behind you. If you fly too low, you might crash into a crater rim.

You can make the program shorter by eliminating the instructions. Just delete lines 1–95 and lines 999–2018. Mars Lander Simulator also operates on the MC-10, with the following changes. Delete lines 1–95 and lines 1000–2018. Change line 430 to read IF H>63 THEN 440. Change line 370 to read IF V=25 and A=0 THEN 810. Add line 100 to read CLEAR200. Add line 435 to read GOTO 450. ■

"Mars Lander Simulator also operates on the MC-10."

Address correspondence to Robert A. Mauro, 257 Center Lane, Levittown, NY 11756.

```
5 CLS0
10 PRINT@68,"WELCOME TO MARS' LA
NDER!";
20 PRINT@160,"DO YOU WANT TO SEE
 INSTRUCTIONS?";
30 PRINT@230,"IF YOU DO, PRESS '
Y'.";
40 PRINT@352,"IF YOU DON'T, PRES
S 'F' TO FIRE";
50 PRINT@384,"YOUR RETROS & BEGI
N THE DESCENT";
60 PRINT@428,"TO MARS!";
70 B$=INKEY$:IF B$="" THEN 10
80 IF B$= "Y" THEN CLS:GOTO 999
90 IF B$="F" THEN 110
95 IF B$<>"F" OR B$ <>"Y" THEN 1
0
110 CLS0:H=0:A=10000:TT=1:V=1:F=
100:T=0:TM=0:N=1:Q=415:D=32
120 J$=CHR$(143+48)
130 H$=CHR$(131+48)+CHR$(131+48)
140 HH$=H$+H$
150 HI$=HH$+HH$
160 K$=CHR$(135+32)+CHR$(139+32)
170 KK$=HI$+HI$+HH$
180 KL$=J$+HI$+K$+KK$+J$
190 PRINT@Q,KL$
200 Q=Q-1
210 IF Q=383 THEN Q=415:CLS0
220 PRINT@460,"ALTITUDE";:PRINT@
492,A;"FT";
230 PRINT@449,"FUEL LEFT";:PRINT
@481,F;"LBS.";
240 PRINT@470,"DIRECTION";:PRINT
@502,D;"FPS";
250 SOUND225,1
260 A$=INKEY$
270 IF POINT(H,V)=7 THEN 790
280 IF POINT(H,V)=1 THEN 720
290 IF POINT (H,V)=3 THEN 800
300 IF POINT (H,V)=5 THEN 730
310 TM=TM+1
320 D=D-(32*TT):A=1
330 A=INT((A+D)+10000)
340 V=ABS((25/(10000/A))-25):TT=
TT+1
350 IF H<0 THEN H=0
360 IF F<0 THEN 680
370 IF V=25 AND A=0 THEN 810 ELS
E 380
380 IF V<1 THEN V=0
390 IF V>25 THEN 710
395 IF POINT(H,V)=4 THEN 705
400 IF A =<0 THEN 710
410 IF A$ = CHR$(8) THEN F=F-9:G
OTO 120
420 H=H+1
430 IF H>63 THEN 440 ELSE 450
440 H=0:CLS0
450 SET(H,V,8)
460 SET(53,2,7):SET(54,3,1):SET(
54,2,7):SET(55,3,1):SET(55,2,7):
SET(56,3,1):SET(55,2,7):SET(52,4
,1):SET(52,3,7):SET(53,4,1)
470 SET(53,3,7):SET(54,4,1):SET(
55,4,1):SET(53,5,1):SET(56,4,7):
SET(54,5,7):SET(55,5,7)
480 SET(RND(50),RND(20),5):SET(R
ND(50),RND(20),5):SET(RND(50),RN
D(20),5)
490 N=N+1
500 IF N=6 THEN CLS0:N=1
510 IF VAL(A$)<>1 THEN 530
520 IF VAL(A$)=1 THEN D=D+(32*TT
): F=F-1:GOTO120
530 IF VAL(A$)<>2 THEN 550
540 IF VAL(A$)= 2 THEN D=D+(64*T
T):F=F-2:GOTO120
550 IF VAL(A$)<>3 THEN 570
560 IF VAL(A$)=3 THEN D=D+(96*TT
):F=F-3:GOTO 120
570 IF VAL(A$)<>4 THEN 590
580 IF VAL(A$)=4 THEN D=D+(128*T
T):F=F-4:GOTO120
590 IF VAL(A$)<>5 THEN 610
600 IF VAL(A$)=5 THEN D=D+(160*T
T):F=F-5:GOTO120
610 IF VAL(A$)<>6 THEN 630
620 IF VAL(A$)=6 THEN D=D+(193*T
T):F=F-6:GOTO120
630 IF VAL(A$)<>7 THEN 650
640 IF VAL(A$)=7 THEN D=D+(225*T
T):F=F-7:GOTO120
650 IF VAL(A$)<>8 THEN 670
660 IF VAL(A$)=8 THEN D=D+(257*T
T):F=F-8:GOTO120
670 IF A$="" THEN 120
680 CLS0:PRINT@70,"YOU RAN OUT O
F FUEL...";:FOR T=225 TO 205 STE
P-1:SOUNDT,1 :NEXTT:FORC=1TO5:CL
S5:CLS0:SOUND1,1:NEXTC
690 CLS8:PRINT@70,"YOU RAN OUT O
F FUEL!";
700 PRINT@102,"AND CRASHED ON MA
RS!";:D=0:GOTO740
705 CLS4:PRINT@32,"YOU'VE CRASHE
D INTO CRATER RIM!";:D=0:A=RND(5
00):GOTO740
710 CLS8:PRINT@37,"YOU'VE CRASHE
D ON MARS!";:D=0:GOTO740
720 CLS:PRINT@35,"YOU'VE CRASHED
 ON GREENLAND!";:D=0:GOTO740
730 CLS5:PRINT@35,"YOU WERE HIT
BY A METEORITE";
740 PRINT@232,"YOU FLEW FOR";
750 PRINT@266,TM;"DAYS";
760 PRINT@428,"ALTITUDE";:PRINT@
460,A;"FT";
770 PRINT@417,"FUEL LEFT";:PRINT
@449,F;"LBS.";
780 PRINT@438,"DIRECTION";:PRINT
@470,D;"FPS";:SOUND60,3:SOUND60,
3:SOUND60,3:SOUND20,7:END
790 CLS7:PRINT@34,"YOU'VE LANDED
 IN THE PACIFIC!";:D=0:A=0:GOTO8
20
800 CLS3:PRINT@34,"YOU'VE LANDED
 ON MARS' BASE!";:A=0:D=0:GOTO82
0
810 CLS8:PRINT@38,"YOU'VE LANDED
 ON MARS!";:A=0:D=0
820 PRINT@232,"YOU FLEW FOR";
830 PRINT@266,TM;"DAYS";
840 PRINT@428,"ALTITUDE";:PRINT@
460,A;"FT";
850 PRINT@417,"FUEL LEFT";:PRINT
@449,F;"LBS.";
860 PRINT@438,"DIRECTION";:PRINT
@470,D;"FPS";:SOUND125,2:SOUND14
7,2:SOUND170,2:SOUND197,3:SOUND1
85,2:SOUND197,4:END
999 CLS8
1000 PRINT"YOU'RE ABOUT TO GO WH
ERE NO"
1001 PRINT"MAN OR WOMAN HAS GONE
 BEFORE --"
1002 PRINT"MARS. YOU'RE ORBITING
 AND ARE"
1004 PRINT"NOW READY TO FIRE YOU
R RETROS"
1005 PRINT"AND DESCEND TO MARS'
BLUE BASE."
1006 PRINT" BUT BEWARE!  IT'S NO
T EASY!"
1007 PRINT"THEREFORE, IF YOU CHO
OSE TO"
1008 PRINT"ABORT, YOU CAN RETURN
 TO"
1009 PRINT"EARTH -- BUT BE SURE
TO LAND"
1010 PRINT"IN THE BLUE OCEAN AND
 NOT ON"
1011 PRINT"THE GREEN CONTINENTS
--"
1012 PRINT"OR YOU'LL CRASH! AND
DON'T GET"
1013 PRINT"HIT BY A METEORITE, O
R YOU'LL"
1014 PRINT"BE DESTROYED IN A FLA
SH!  NOW"
1015 PRINT"PUSH 'C' FOR CONTROL
INFO"
1016 D$=INKEY$:IF D$="" THEN 101
6
1017 IF D$="C" THEN CLS:GOTO 200
0
1018 IF D$<>"C" THEN 1016
2000 CLS8
2001 PRINT"THE CONTROLS CONSIST
OF BUTTONS"
2002 PRINT"1-8 AND <_>.  BUTTON
1 WILL"
2003 PRINT"LET YOU ORBIT (WITHOU
T FALLING)"
2004 PRINT"OR HOLD YOU ON A RETU
RN COURSE"
2005 PRINT"FOR EARTH.  BUTTONS 2
-8 WILL"
2006 PRINT"FIRE YOUR LANDING ROC
KETS. THE"
2007 PRINT"HIGHER THE NUMBER, TH
E FASTER"
2008 PRINT"YOU'LL GAIN ALTITUDE.
 EASY ON"
2009 PRINT"FUEL!  RUN OUT AND YO
U CRASH!"
2010 PRINT"PRESS NO BUTTON & YOU
 FALL"
2011 PRINT"FASTER & FASTER AND C
RASH ON"
2012 PRINT"MARS.  PRESS <_> AND
YOU DROP"
2013 PRINT"STRAIGHTER DOWN TO TH
E PLANET."
2014 PRINT"TO FIRE YOUR RETROS &
 DECEND"
2015 PRINT"TO MARS, PUSH 'F' TO
FIRE!"
2016 C$=INKEY$:IF C$="" THEN 201
6
2017 IF C$="F" THEN 110
2018 IF C$<>"F" THEN 2016
```

END

Program Listing. Mars Lander Simulator

BY KARL ANDREASSEN

COLORFUL CRYPTOLOGY— PART VII

Learn the Old Wizard's secret and other non-alphabetic ways to create your own secret messages.

December's Old Wizard's star contest was a close race, and many readers sent in correct solutions. I had no idea so many people would crack the cryptogram, especially since I hadn't given a hint of how to proceed. There are some sharp readers out there.

Glenn Reed of Houston, TX took first place, followed by Donald L. McGarry of Centerport, NY in second and Michael R. Hughey of Lafayette, IN, in third.

Many letters came in from Canada four or five days after the bulk of those from the U.S., indicating a problem with using the earliest postmark as the only determiner of first solutions. Some people get their *HOT CoCo* before others do. I'll try to come up with something better next time.

The Secret

The Program Listing created the star message. Figure 1 shows the star-field alphabet: Read the stars and spaces beneath the plaintext letters vertically. The five-unit star/space combinations allowed for more than enough substitutes for each letter in the 26-letter alphabet. The cell configurations were arbitrary.

While the approach to analysis from a cold start may have taken a number of paths, one viable beginning is to count the lines and see how many different groups of lines are possible. Since there


```
10 '==STARCRYP/CRP  Printer: set
 0 0 30
20 CLS:PRINT "STARCRYPT"
30 PRINT "by Karl Andreassen"
40 CLEAR 1000 :DIM A$(27),B$(250
),C$(250),D$(250),E$(250),F$(250
)
50 PRINT :PRINT "Enter plaintext
 ,no punctuation.  < * > to end.
" :PRINT
60 FOR X=1 TO 27 :READ A$(X) :NE
XT X :X=0 '==Read symbol alphabe
t
70 Z$=INKEY$ :IF Z$="" THEN 70
80 IF Z$="*" THEN PRINT :Y=1 :GO
TO 200
90 A=ASC(Z$)-63 '==Assign ASCII
number
100 IF A=-31 THEN A=1 '==Error c
atcher
110 X=X+1 '==Count characters; a
ssign stars to arrays
120 B$(X)=MID$(A$(A),1,1)
130 C$(X)=MID$(A$(A),2,1)
140 D$(X)=MID$(A$(A),3,1)
150 E$(X)=MID$(A$(A),4,1)
160 F$(X)=MID$(A$(A),5,1)
170 PRINT Z$; '==Print plaintext
180 GOTO 70
190 X=X-30 :Y=Y+30 '==Prepare to
 print 30-character ciphertext l
ines
200 FOR Z=Y TO Y+30:PRINT B$(Z);
 :NEXT Z :PRINT
210 FOR Z=Y TO Y+30:PRINT C$(Z);
 :NEXT Z :PRINT
220 FOR Z=Y TO Y+30:PRINT D$(Z);
 :NEXT Z :PRINT
230 FOR Z=Y TO Y+30:Print E$(Z);
 :NEXT Z :PRINT
240 FOR Z=Y TO Y+30:PRINT F$(Z);
 :NEXT Z :PRINT
250 IF X>30 THEN 190ELSE 300 '==
End of message cutoff
260 DATA "      ","    *","   * "
,"   **","  *  ","  * *","  ** "
,"  ***"
270 DATA " *   ","  *  *","  * * "
,"  * **","  ** *","  *** ","  ****"
,"*    "
280 DATA "*   *","*  * ","*  **"
,"* *  ","* ** ","* * *","* ***"
,"**   "
290 DATA "**  *","** * ","***  "
300 END
```

Program Listing. Creating the Star Message

are 15, the only factors are 3, 5, 15, and linear.

If all 15 lines were involved serially, the linear path from left to right would have been the most logical first investigation, suggesting a count of star and space total (1,185) to establish the probable number of characters per cell. Following the progression from left to right, by the second line you run into a most disconcerting series of spaces without stars, ruling out likelihood of the linear path.

Three characters per cell will not provide full coverage for the alphabet. Five characters per cell is the obvious choice, therefore, because 15 would be very wasteful.

"Your CoCo can handle numerals as strings, or read them into an array as integers. It can add, subtract, or treat them algebraically."

If you look at the entire star field, you will notice that a few lines contain more stars strung side by side than do the balance of lines, some of which carry many sequential spaces. The general line format creates a pattern suggestive of three sets of five lines, containing as it does a thrice-repeated rhythmic pattern.

Divide the field with vertical lines separating the individual groups of five characters. You're getting warm. Draw horizontal lines beneath each set of five star lines. Note the vertical sets of five star/space patterns, and observe that some patterns are repeated.

Convert these patterns into letters that you can enter into one of your crypto-helper programs. If I suspected that this was a cryptogram within a cryptogram, I would assign letters alphabetically to each unique cell as it appears in the message, and then treat the resultant cryptogram as though it had never been presented symbolically.

Lines 260–290 of Listing 1 reveal the cipher alphabet as string data. Study this symbol-alphabet: Many different alphabets are possible by changing the placement of stars and spaces.

This particular star field has a plaintext message that is fairly easy to identify from the repetitions and combinations of symbols. You can be easily misled by such an assumption, however, if the author transposes his plaintext letters before he enciphers them as star cells. Figure 2 reveals the answer to the Wizard's message.

My sincere thanks to all who entered, and congratulations to the winners. Further, I am delighted that so many of you were able to decipher the message. There are some keenly analytical minds out there.

Figure 3 gives the answers to the other five December cryptograms.

Using Numerals to Encipher

After studying the method of the star message, you should easily see that you can also use numerals to represent letters. The decimal series consists of ten characters, 0–9. Standing alone, they can represent but 10 letters. In pairs, they offer a potential total of 99—virtually four times the number required to represent a 26-letter alphabet. You can select 26 combinations as mixed singles and pairs, although the absence of certain combinations can alert a cryptanalyst to a probable alphabet.

The CoCo's ASCII keyboard code uses numerals 65–90 to represent the uppercase (capital) letters of the alphabet. As a result, a numerical alphabet already exists throughout the computer world. But you probably want your numerical alphabet to be just a bit more cryptic than standard ASCII code.

As a first step, compose a look-up table of digital pairs:

#	A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	0	1	2	3
1	2	3	4	5	6	7	8	9	0	2	3	4	5

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
4	5	6	7	8	9	0	1	2	3	4	5	6
6	7	8	9	0	3	4	5	6	7	8	9	0

To compose a message using this numerical alphabet, we simply read down from each letter:

N	U	M	E	R	I	C	A	L	
46	15	35	56	80	90	34	12	24	01
C	I	P	H	E	R	S			
34	90	68	89	56	80	93			
H	A	V	E		T	H	E	I	R
89	12	26	56	01	04	89	56	90	80	01
N	U	M	B	E	R	S			
46	15	35	23	56	80	93			

This method uses two symbols for each letter, plus a space between pairs. (A space has the value of a character. Spaces occur in English text far more often than does even the most frequent letter, E.) This points up an important axiom: To use space and time economically sometimes means you can't use ciphertext characters that are much longer than their plaintext equivalents.

Therefore, reduce the spacing, and consequently the message length, by creating six-numeral groups that still give your confidant a manageable message, as in this example:

461535 568090 341224 349068 895680 938912
265604 895690 804615 352356 809300

You could use five-numeral groups as easily, adding one more level of difficulty for an unauthorized receiver. Further, now that the message is in the familiar five- or six-block code form, you could superencipher it quite easily, making it much more difficult to unscramble.

To superencipher is to subject the enciphered message to yet a second or third standard method of encryption.

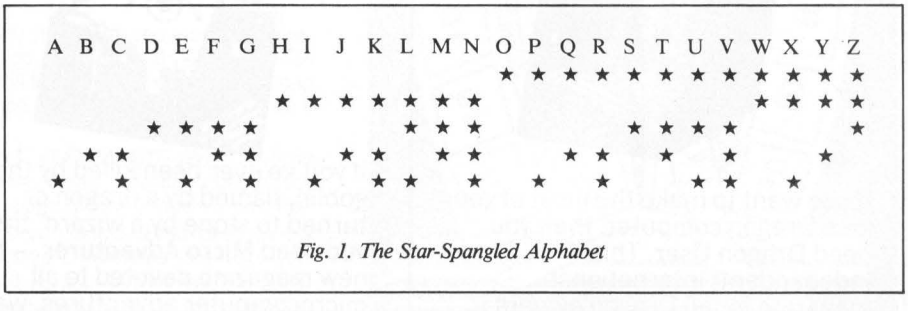


Fig. 1. The Star-Spangled Alphabet

THE STARS YOUR DESTINATION AND GOOD FORTUNE THREE SUBSCRIPTIONS OR EXTENSIONS TO HOT COCO AWAIT THE EARLIEST POSTMARKED CORRECT DECRYPTION OF THIS MESSAGE RECEIVED BY KARL BEFORE THE PUBLISHED DEADLINE WINNERS MAY THANK THESE LUCKY STARS

Fig. 2. The Old Wizard's Star Message in Translation

For instance, an understanding between sender and receiver could include swapping the first and last code groups, second and tenth, third and ninth, and so on, leaving the center group intact in the case of an odd number of groups. This creates no problem for the intended receiver, but does make cracking the cipher more difficult for others.

Your CoCo can handle numerals as strings, or read them into an array as integers. It can add, subtract, or treat them algebraically. Such ease of manipulation makes them almost ideal for computer-encrypted communication.

Later I'll look at octal, hex, and binary arithmetic for their cryptography potential. Binary applications are particularly useful, since you can use ROM chips to encrypt computer files and outputs for security.

I have received some very interesting programs from some of you lately. If you will use your growing crypto and programming skills to create a straightforward program using the above six-numeral format to encipher plaintext, I will feature the best one in a later article.

I also must add that, due to the growing number of articles that need a space in *HOT CoCo*, we're going to run Colorful Cryptology every other month, instead of monthly as we have been. So, I'll see you in May. ■

ONE OF THE WAYS TO MAKE YOUR COMPUTER MORE SECURE IS TO ENCIPHER IMPORTANT DATA BEFORE FILING IT ON TAPE OR DISK

THE COMPUTER IS AN IDEAL CRYPTO MACHINE FOR USE AT THE ORIGINATING POINT OF AN ENCIPHERED MESSAGE BUT UNTIL IT CAN BE PASSED OVER AS A WRISTWATCH YOU MUST USE HUMAN FACULTIES FOR SOME VITAL COMMUNICATIONS

A CLUE TO THE STARS CRYPTOGRAM SPACES ARE EQUALLY AS IMPORTANT AS THE STARS AND THE MESSAGE MAY NOT BE AS LONG AS IT MAY AT FIRST APPEAR

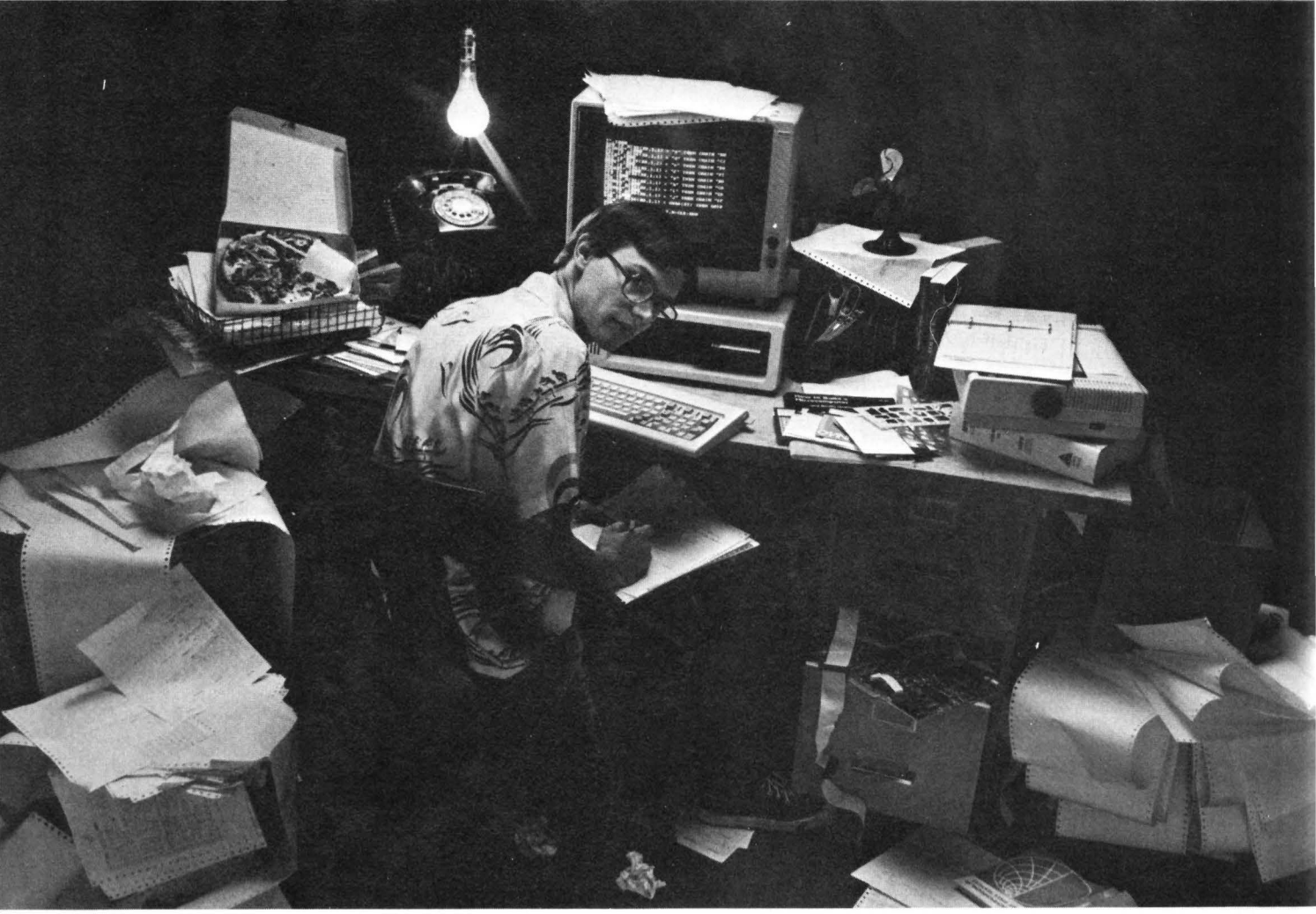
WHILE KEYWORDS ARE VERY IMPORTANT IN CREATING EFFECTIVE ENCIPHERED MESSAGES THE REAL KEY IS PRACTICE IF YOU BREAK AT LEAST ONE CRYPTOGRAM A DAY FOR A YEAR IT WILL BECOME ALMOST EASY NOTE THE ALMOST

EVERY FIXED STAR IS THE HUB OF A SYSTEM OF PLANETS AND EVERY PLANET HAS THE POTENTIAL OF SUPPORTING SOME KIND OF LIFE HOW LONG BEFORE WE ENGINEER AN ATMOSPHERE FOR MARS AND SEED THE MOONS OF JUPITER

Fig. 3. Answers to December Cryptograms

Write to Karl Andreassen at 24750 Chianti Road, Cloverdale, CA 95425.

✓See List of Advertisers on page 130



Got a great new program up your sleeve?

If you write programs, we need each other.
We know that some great software is being developed between the hours of midnight and 6:00 AM, in the silence of America's homes. If you're one of those disk-driven writers, buttoning up by day and hunkering down by night, take heart. Fame and fortune could be right around the corner.

You'll hear from us in 30 days.
I/O WARE is looking for innovative programs for the home and small business markets: word processing, graphic arts, finance, planning, home budgeting, "how-to", etc. Programs that will run on the Commodore 64 (Disk or Tape); IBM PC and PC Jr. (Disk); Apple II (Disk); TRS-80 I, III, IV (Disk), and Color Computer (Disk or Tape). You'll receive our decision in 30 days. If you have what we're looking for we'll immediately send you an advance of *at least* $250.

And that's just the beginning. I/O WARE can give you the kind of exposure (and royalties) you've been dreaming of.

We'll put a cash advance in your pocket.

The I/O WARE Professionals: Duane Manseau, Jim Eastman, Tom Cullity
"We're looking for great new software."

See your name up in bytes.
Every program needs professional documentation, packaging, distribution, and promotion. That's our job. We'll transform your program into a polished, and successful software package.

I/O WARE is part of the team of computer experts publishing the leading computer magazines in the country. Which means your program will have access to major national advertising, direct mail, and hundreds of retail outlets across the country.

So hit us with your best shot—today.
Because we're putting together a very strong line of software. Fast. And there's a good chance you could be a part of the team. Just call Tom Cullity at our software hotline, 603/924-9897, and let us know what you have up your sleeve.

I/O ware

I/O WARE INC.
Peterborough, NH 03458
Attention: Tom Cullity

318

BY MARK D. GOODWIN

JOURNEY TO THE CENTER OF THE ROM—PART V

In this fifth installment of Mark Goodwin's dissection of the Color Basic ROM, he reveals what lies at memory addresses A000 to A754. Routines that reside in this area include Color Basic initialization, keyboard, ROM pack, input and output, CSAVE and CLOAD, and device.—eds.

Address correspondence to Mark Goodwin, Star Route 79, Box 103, Orland, ME 04472.

A000–A00D System Routines

A000–A001	Address of keyboard input routine
A002–A003	Address of character output routine
A004–A005	Address of cassette on routine
A006–A007	Address of cassette block in routine
A008–A009	Address of cassette block out routine
A00A–A00B	Address of joystick in routine
A00C–A00D	Address of write cassette leader routine

A00E–A026 Color Basic Initialization Routine

A00E–A011	Initialize the stack pointer
A012–A013	A = PIA initialization value
A014–A016	Turn on the ROM-pack FIRQ
A017–A018	A = RESET flag
A019–A01A	Coldstart?
A01B–A01C	Jump if it's a coldstart
A01D–A01E	X = RESET address
A01F–A020	A = RESET value
A021–A022	RESET value = 12?
A023–A024	Jump if no RESET
A025–A026	Jump to the RESET address

A027–A073 Hardware Initialization Routine

A027–A029	Y = Return address
A02A–A02C	X = PIA2 pointer
A02D–A02E	Initialize data-direction register A (PIA1)
A02F–A030	Clear control register B (PIA1)
A031–A032	Clear control register A (PIA1)
A033–A035	D = Initialization values
A036–A037	Initialize data-direction register B (PIA1)
A038–A039	Initialize control register A (PIA1)
A03A–A03B	Initialize control register B (PIA1)
A03C–A03D	Clear control register A (PIA2)
A03E–A03F	Clear control register B (PIA2)
A040	Decrement the initialization value
A041–A042	Initialize data-direction register A (PIA2)
A043–A044	A = Initialization value
A045–A046	Initialize data-direction register B (PIA2)
A047–A048	Initialize control register A (PIA2)
A049–A04A	Initialize control register B (PIA2)
A04B–A04C	Set the video control register for the SG4 graphics mode
A04D–A04E	B = RS-232 value
A04F–A050	Send it
A051–A053	U = VDG register pointer
A054–A055	B = number of bits to set
A056–A057	Set a register bit
A058	All bits set?
A059–A05A	Loop until all bits set
A05B–A05D	Set start of video memory to 0400
A05E–A05F	DP = 0
A060–A061	B = RAM jumper bit value
A062–A063	Send RAM jumper test value to peripheral register B (PIA2)
A064–A065	4K RAMs?
A066–A067	Jump if 4K RAMs
A068–A069	Send RAM jumper test value to peripheral register B (PIA2)
A06A–A06B	64K RAMs?
A06C–A06D	Jump if 64K RAMs
A06E–A06F	Back up the RAM size pointer
A070–A071	Set the RAM size register bit
A072–A073	Jump to the return address

A074–A0E7 Cold-Start Initialization Routine

A074–A076	X = End of the Basic Communications Area pointer

Listing continued

Save 75 to 100 Hours of Typing Time with ...

instant CoCo T.M.

Don't miss another great program because you didn't have the time to type it in. **instant CoCo,** the monthly cassette loader from HOT CoCo, will save you hours of keyboarding time. Just load **instant CoCo** into your Color Computer* and within seconds you get to use:

- new applications—reliable programs for education, business, home and hobby help the whole family grow in computer knowledge.
- time saving utilities—will make your routine tasks a breeze to perform, while expanding the capabilities of your Color Computer.
- action-packed games—save your quarters and get hours of video fun without leaving home.

The Best of '83 cassette contains more than 33 of the best programs from all the 1983 issues of HOT CoCo Magazine. And it's yours for the astonishingly low price of $14.97! That means you get quality programs entered automatically for less than 45¢ each. There are monthly cassettes available containing all the major programs from each issue of HOT CoCo, beginning with the Janaury 1984 issue. Every month you'll be able to load great programs in seconds and follow right along with your issue of HOT CoCo Magazine.

So don't delay! Mail the attached card or the coupon below NOW! You don't want to miss one more great program because you didn't have the time to type it in.

Or call toll-free—1-800-258-5473—and use your MasterCard, Visa, or American Express card. Foreign and New Hampshire customers, please call 603-924-9471.

*TRS-80 Color Computer is a trademark of Radio Shack, a division of Tandy Corp.

instant CoCo T.M.

Please allow 6-8 weeks for delivery

YES! I want to save time with INSTANT COCO® 3-84

- ☐ Please send me ____ copies of the "Best of '83" at $16.47 each.
- ☐ Please send me ____ copies of this month's issue at $11.47 each.
- ☐ Please sign me up for a one year subscription beginning with this month's issue at $99.97

Price includes postage and handling. Foreign Air Mail please add an addition 45¢ per cassette or $25 per subscription. US funds drawn on US banks ONLY.

☐ CHECK/MO ☐ MC ☐ VISA ☐ AE

Card # _____ Exp. Date_____

Signature _____

Name _____

Address _____

City _____ State_____ Zip_____

instant CoCo • 80 Pine Street • Peterborough, NH 03458

Listing continued

A077–A078 Clear the memory location
A079–A07A Bump the memory pointer
A07B–A07C Loop till the Basic Communications Area has been cleared
A07D–A07F Clear the screen
A080–A081 Clear the start of the Basic program area minus one
A082–A083 Save the start of the Basic program area pointer
A084–A085 A = Memory value
A086 Invert the memory value
A087–A088 Store the inverted memory value
A089–A08A Valid memory location?
A08B–A08C Jump if it's not a valid memory location
A08D–A08E Bump the memory pointer
A08F–A090 Restore the memory value
A091–A092 Loop until the top of RAM has been located
A093–A094 Save the top of RAM pointer
A095–A096 Save the start of the reserved-memory pointer
A097–A098 Save the next available location in string-space pointer
A099–A09C Adjust the memory pointer for the start of the string space
A09D–A09E Save the start of the string-space pointer
A09F–A0A0 Set the stack pointer to the start of the string space
A0A1–A0A3 X = ROM area pointer
A0A4–A0A6 U = Destination address pointer
A0A7–A0A8 B = number of bytes to move
A0A9–A0AB Do block move
A0AC–A0AE U = Destination address pointer
A0AF–A0B0 B = number of bytes to move
A0B1–A0B3 Do block move
A0B4–A0B5 Adjust the memory pointer
A0B6–A0B7 Save the memory pointer
A0B8–A0B9 Save the memory pointer
A0BA–A0BC X = Extended Color Basic links pointer
A0BD–A0BF D = RTS op-code and number of bytes to set
A0C0–A0C1 Save a RTS op-code
A0C2 All links set?
A0C3–A0C4 Loop until all the links have been set
A0C5–A0C7 Save a RTS op-code
A0C8–A0CA Do NEW
A0CB–A0CD X = Extended Color Basic test value
A0CE–A0D0 Extended Color Basic?
A0D1–A0D4 Jump if Extended Color Basic exists
A0D5–A0D6 Enable the interrupts
A0D7–A0D9 X = Color Basic 1.1 message pointer
A0DA–A0DC Display the message
A0DD–A0DF X = RESET address
A0E0–A0E1 Save the RESET address
A0E2–A0E3 A = RESET flag
A0E4–A0E5 Save the RESET flag
A0E6–A0E7 Jump to the command mode

A0E8–A0F5 Warm-Start Routine

A0E8 NOP for RESET
A0E9–A0EA Set the current device to the video display
A0EB–A0ED Reset the Basic pointers
A0EE–A0EF Enable the interrupts
A0F0–A0F2 Clear the screen
A0F3–A0F5 Jump to the command mode

A0F6–A10C ROM-Pack Routine

A0F6–A0F8 ROM pack?
A0F9–A0FA Jump if a ROM pack exists
A0FB Return
A0FC–A101 Delay
A102–A104 Y = return address
A105–A107 Do hardware initialization
A108–A109 Clear the RESET flag
A10A–A10C Jump to the ROM pack

A10D–A146 Area Downloaded to RAM

A147–A170 Color Basic 1.1 Message

A171–A175 Keyboard Routine

A171–A172 Scan the keyboard
A173–A174 Clear bit 7 of the key pressed
A175 Return

A176–A198 Input Routine

A176–A178 Call the Extended Color Basic link
A179–A17A Flag not end of file
A17B–A17C Current device = keyboard?
A17D–A17E Jump if the current device is the keyboard
A17F–A180 Any bytes left in cassette buffer?
A181–A182 Jump if there are bytes left in the cassette buffer
A183–A184 Flag end of file
A185 Return
A186–A187 Save the registers
A188–A189 X = next byte in the cassette-buffer pointer
A18A–A18B Get a byte from the cassette buffer
A18C–A18D Save the byte on the stack
A18E–A18F Save the new cassette-buffer pointer
A190–A191 Decrement the number of bytes left in the cassette buffer
A192–A193 Jump if there are any bytes left in the cassette buffer
A194–A196 Read the next cassette block
A197–A198 Get the registers and return

A199–A1B0 Blink-the-Cursor Routine

A199–A19A Blink the cursor?
A19B–A19C Jump if no blink
A19D–A19E B = new counter value
A19F–A1A0 Save the new cursor counter
A1A1–A1A2 X = Cursor location
A1A3–A1A4 A = Cursor character
A1A5–A1A6 Bump the cursor's color value
A1A7–A1A8 Mask the cursor character
A1A9–A1AA Display the new cursor character
A1AB–A1AD X = delay value
A1AE–A1B0 Delay and return

A1B1–A1C0 Keyboard Routine

A1B1–A1B2 Save the registers
A1B3–A1B4 Blink the cursor
A1B5–A1B6 Scan the keyboard
A1B7–A1B8 Loop until a key is pressed
A1B9–A1BA B = space
A1BB–A1BE Display the space at the cursor location
A1BF–A1C0 Get the registers and return

A1C1–A1C7 Keyboard Routine

A1C1–A1C2 Save the registers
A1C3–A1C4 Scan the keyboard
A1C5 Set the flags for the keyboard value
A1C6–A1C7 Get the registers and return

A1C8–A26C Keyboard-Driver Routine

A1C8–A1CA U = PIA1 pointer
A1CB–A1CD X = keyboard work-area pointer
A1CE A = 0 and clear carry
A1CF A = FF
A1D0–A1D1 Make a 3-byte hole on the stack
A1D2–A1D3 Send the first column output value

Listing continued

Listing continued

A1D4–A1D5	Ignore
A1D5	Set carry
A1D6–A1D7	Shift the column output
A1D8–A1D9	Jump if no key pressed
A1DA–A1DB	Bump the column number
A1DC–A1DD	Get the row value
A1DE–A1DF	Save the row value
A1E0–A1E1	Mask it for the same as the last key pressed
A1E2–A1E3	Mask it for the same as the last key pressed
A1E4–A1E5	B = row value
A1E6–A1E7	Save the new keyboard work-area value
A1E8	Same as the last key pressed?
A1E9–A1EA	Loop if it's the same as the last key pressed
A1EB–A1EC	B = column output value
A1ED–A1EE	Save the column output value
A1EF–A1F0	B = start value
A1F1–A1F2	Bump it for a row
A1F3	Adjusted for the row?
A1F4–A1F5	Loop until it's adjusted for the row
A1F6–A1F7	Adjust it for the column
A1F8–A1F9	Jump if the @ key was pressed
A1FA–A1FB	Nonalphabetic?
A1FC–A1FD	Jump if it's nonalphabetic
A1FE–A1FF	Make it ASCII
A200–A201	Shift?
A202–A203	Jump if the shift key was pressed
A204–A206	Uppercase only?
A207–A208	Jump if uppercase only
A209–A20A	Make it lowercase
A20B–A20C	Save the ASCII value
A20D–A20F	X = delay value
A210–A212	Delay
A213–A214	B = column output value
A215–A216	Get the row value
A217	Bounce?
A218–A219	Jump if bounce
A21A–A21B	B = column output value
A21C–A21D	Get the row value
A21E–A21F	Another key pressed?
A220–A221	A = ASCII value
A222–A223	Jump if another key was pressed
A224–A225	Shift 0?
A226–A227	Jump if it isn't shift 0
A228–A22A	Invert the shift-0 flag
A22B	Zero the ASCII value
A22C–A22D	Clean up the stack and return
A22E–A22F	A = column-output value
A230–A231	Send the column output
A232–A233	A = row value
A234–A235	Mask it for the shift key
A236	Return
A237–A238	Send the column output
A239–A23A	A = row value
A23B–A23C	Mask the row value
A23D–A23E	Last column?
A23F–A240	Jump if it isn't the last column
A241–A242	Mask the row value
A243	Return
A244–A245	B = @ value
A246–A248	X = table pointer
A249–A24A	Up arrow to 0?
A24B–A24C	Jump to a 0 if it's an up arrow
A24D–A24F	X = table pointer
A250–A251	Enter to @?
A252–A253	Jump if it's an enter to an @
A254–A255	Shift?
A256–A257	Less than a semicolon?
A258–A259	Jump if it's less than a semicolon
A25A–A25B	Invert the shift value

Listing continued

Listing continued

A25C	Shift?
A25D–A25E	Jump if no shift
A25F–A260	Adjust it for shift
A261–A262	Jump
A263	B = table offset
A264–A265	Shift?
A266–A267	Jump if no shift
A268	Bump the table offset
A269–A26A	B = ASCII value
A26B–A26C	Jump

A26D Nothing Here

A26E–A281 Keyboard Lookup Table

A26E	Up arrow: 5E
A26F	Shifted up arrow: 5F
A270	Down arrow: 0A
A271	Shifted down arrow: 5B
A272	Left arrow: 08
A273	Shifted left arrow: 15
A274	Right arrow: 09
A275	Shifted right arrow: 5D
A276	Space: 20
A277	Shifted space: 20
A278	Zero: 30
A279	Shifted zero: 12
A27A	Enter: 0D
A27B	Shifted enter: 0D
A27C	Clear: 0C
A27D	Shifted clear: 5C
A27E	Break: 03
A27F	Shifted break: 03
A280	@: 40
A281	Shifted @: 13

A282–A2BE Output Routine

A282–A284	Call the Extended Color Basic link
A285–A286	Save B
A287–A288	B = current device flag
A289	Set the flags for the current device
A28A–A28B	Get B
A28C–A28D	Jump if it's the printer
A28E–A28F	Jump if it's the video display
A290–A291	Save the registers
A292–A293	B = OPEN/CLOSE flag
A294	OPEN"I"?
A295–A296	Jump if it's OPEN"I"
A297–A298	B = Length of the data block
A299	Bump it
A29A–A29B	Jump if the buffer isn't full
A29C–A29D	Write the data block
A29E–A29F	X = next available buffer-location pointer
A2A0–A2A1	Save the character in the buffer
A2A2–A2A3	Save the new buffer pointer
A2A4–A2A5	Bump the length of the data block
A2A6–A2A7	Get the registers and return
A2A8–A2A9	B = block type
A2AA–A2AB	Save the block type
A2AC–A2AE	X = buffer pointer
A2AF–A2B0	Save the buffer pointer
A2B1–A2B2	B = 0
A2B3–A2B4	Save it as the block length
A2B5–A2B6	Save the registers
A2B7–A2B9	Write the block
A2BA–A2BB	Get the registers
A2BC–A2BE	Go reset the buffer values

Listing continued

↙See List of Advertisers on page 130

Listing continued

A2BF–A2F4 Printer-Driver Routine

A2BF–A2C0	Save the registers
A2C1–A2C2	Disable the interrupts
A2C3–A2C4	Send a start bit
A2C5	B = RS-232 value
A2C6–A2C7	Send a start bit
A2C8–A2C9	B = number of bits to send
A2CA–A2CB	Save the bit counter
A2CC	B = starting RS-232 value
A2CD	Put the next bit into carry
A2CE	Move the bit from carry to B
A2CF	Put it into bit 1
A2D0–A2D1	Send the bit
A2D2–A2D3	Get the bit counter
A2D4	All bits sent?
A2D5–A2D6	Loop until all of the bits have been sent
A2D7–A2D8	Send a stop bit
A2D9–A2DA	Get CC and A
A2DB–A2DC	Was a carriage return just sent?
A2DD–A2DE	Jump if a carriage return was sent
A2DF–A2E0	Bump the current carriage position
A2E1–A2E2	B = carriage position
A2E3–A2E4	End of the line?
A2E5–A2E6	Jump if it isn't the end of the line
A2E7–A2E8	Carriage position = 0
A2E9–A2EC	Delay for the carriage return
A2ED–A2EF	B = printer status
A2F0	Printer ready?
A2F1–A2F2	Loop until the printer is ready
A2F3–A2F4	Get the registers and return

A2F5–A2FA Nothing Here

A2FB–A309 RS-232 Output Routine

A2FB–A2FC	B = RS-232 value
A2FD–A2FF	Send the value to the RS-232
A300–A301	Delay for the baud rate
A302–A303	X = baud rate delay value
A304–A306	Ignore
A305–A306	X = carriage-return delay value
A307–A309	Delay and return

A30A–A35E Video-Driver Routine

A30A–A30B	Save the registers
A30C–A30D	X = current cursor position
A30E–A30F	Backspace?
A310–A311	Jump if it's not a backspace
A312–A314	Cursor position = start of video memory?
A315–A316	Jump if the cursor position is equal to the start of video memory
A317–A318	A = space
A319–A31A	Display a space at the last cursor position
A31B–A31C	Jump
A31D–A31E	Carriage return?
A31F–A320	Jump if it's not a carriage return
A321–A322	X = current cursor position
A323–A324	A = space
A325–A326	Display a space at the cursor location
A327–A328	D = updated cursor position
A329–A32A	Mask B for the cursor line position
A32B–A32C	Loop until the start of the next line has been reached
A32D–A32E	Jump
A32F–A330	Control code?
A331–A332	Jump if it's a control code
A333	Graphic character?
A334–A335	Jump if it's a graphic character
A336–A337	Less than an A?
A338–A339	Jump if it's less than an A
A33A–A33B	Uppercase?
A33C–A33D	Jump if it's uppercase
A33E–A33F	Mask the character
A340–A341	Invert the character
A342–A343	Display the character at the cursor location
A344–A345	Save the new cursor position
A346–A348	End of video memory reached?
A349–A34A	Jump if the end of video memory hasn't been reached
A34B–A34D	X = start of video memory
A34E–A350	D = two characters from the next line
A351–A352	Move the characters up a line
A353–A355	Scroll done?
A356–A357	Loop until the scroll is done
A358–A359	B = space
A35A–A35C	Blank the last line
A35D–A35E	Get the registers and return

A35F–A38C Output Routine

A35F–A361	Call the Extended Color Basic link
A362–A363	Save the registers
A364–A365	Clear the cassette flag
A366–A367	A = current device number
A368–A369	Jump if it's the video display
A36A	Cassette?
A36B–A36C	Jump if it's the cassette
A36D–A36E	X = comma field width and the last comma field
A36F–A370	D = printer width and the carriage position
A371–A372	Jump
A373–A374	B = LSB of the current cursor position
A375–A376	Mask it for the cursor line position
A377–A379	X = comma field width and the last comma field
A37A–A37B	A = line length
A37C–A37D	Save the comma field width and the last comma field
A37E–A37F	Save the line position
A380–A381	Save the line length
A382–A383	Get the registers and return
A384–A385	Flag the cassette
A386–A388	X = comma field width and the last comma field
A389	Line length = 0
A38A	Line position = 0
A38B–A38C	Jump

A38D–A3EC Input Routine

A38D–A38F	Clear the screen
A390–A392	Call the Extended Color Basic link
A393–A394	Flag no key pressed
A395–A397	X = start of the input buffer
A398–A399	B = length of the input
A39A–A39C	Get a character from the current device
A39D–A39E	End of file?
A39F–A3A0	Jump if it's the end of the file
A3A1–A3A2	Current device = keyboard?
A3A3–A3A4	Jump if it isn't the keyboard
A3A5–A3A6	Clear key?
A3A7–A3A8	Jump if it's a clear key
A3A9–A3AA	Backspace?
A3AB–A3AC	Jump if it isn't a backspace
A3AD	Decrement the length of the input
A3AE–A3AF	Jump if the buffer is empty
A3B0–A3B1	Decrement the input-buffer pointer
A3B2–A3B3	Jump
A3B4–A3B5	Shift left arrow?
A3B6–A3B7	Jump if it isn't a shift left arrow
A3B8	Decrement the length of the input
A3B9–A3BA	Jump if the buffer is empty
A3BB–A3BC	A = backspace
A3BD–A3BF	Erase the last character

Listing continued

Listing continued

A3C0–A3C1 Loop until all the input has been erased
A3C2–A3C3 Break key?
A3C4–A3C5 Set carry to signal break key
A3C6–A3C7 Jump if it's a break key
A3C8–A3C9 Enter key?
A3CA–A3CB Jump if it isn't an enter key
A3CC Clear carry to signal no break key
A3CD–A3CE Save the flags
A3CF–A3D1 Do a carriage return
A3D2–A3D3 Flag the end of the input
A3D4–A3D6 X = start of the input minus one
A3D7–A3D8 Get the flags and return
A3D9–A3DA Control code?
A3DB–A3DC Jump if it's a control code
A3DD–A3DE Graphic character?
A3DF–A3E0 Jump if it's a graphic character
A3E1–A3E2 Buffer full?
A3E3–A3E4 Jump if the buffer is full
A3E5–A3E6 Save the character in the buffer
A3E7 Bump the length of the input
A3E8–A3EA Display the character
A3EB–A3EC Jump

A3ED–A415 Device Routine

A3ED–A3EF Call the Extended Color Basic link
A3F0–A3F1 A = current device number
A3F2–A3F3 Jump if it's the video display
A3F4 Printer?
A3F5–A3F6 Jump if it's the printer
A3F7–A3F8 A = OPEN/CLOSE flag
A3F9–A3FA Jump if a file is OPEN
A3FB–A3FC B = NO error code
A3FD–A3FF Display the message
A400 OPEN"I"
A401–A402 Jump if OPEN"I"
A403–A405 Display FM error message
A406–A408 Call the Extended Color Basic link
A409–A40A A = current device number
A40B Cassette?
A40C–A40D Jump if it isn't the cassette
A40E–A40F A = OPEN/CLOSE flag
A410–A411 NO error if CLOSE
A412 OPEN"I"?
A413–A414 Display FM error message if OPEN"I"
A415 Return

A416–A425 Color Basic CLOSE Command

A416–A417 Jump if it's the end of the Basic statement
A418–A41A Evaluate the device number
A41B–A41C CLOSE the file
A41D–A41E Get the next Basic character
A41F–A420 Jump if it's the end of the Basic statement
A421–A423 Check the syntax
A424–A425 Loop until all the files have been CLOSEd

A426–A44B Device Routine

A426–A428 Call the Extended Color Basic link
A429–A42A A = cassette device number
A42B–A42C Current device = cassette
A42D–A42F Call the Extended Color Basic link
A430–A431 A = current device number
A432–A433 Current device = video display
A434 Was the current device the cassette?
A435–A436 Jump if the current device wasn't the cassette
A437–A438 A = OPEN/CLOSE flag
A439–A43A OPEN"O"?
A43B–A43C Jump if it's not OPEN"O"
A43D–A43E A = number of bytes in the cassette buffer
A43F–A440 Jump if the buffer is empty
A441–A443 Write the remaining bytes
A444–A448 Write an EOF block
A449–A44A Flag CLOSE
A44B Return

A44C–A497 Color Basic CSAVE Command

A44C–A44E Evaluate the file name
A44F–A450 Get the next Basic character
A451–A452 Jump if it's the end of the Basic statement
A453–A455 Check the syntax
A456–A457 B = syntax check character
A458–A45A Check the syntax
A45B–A45C Return if not ASCII
A45D Zero A
A45E–A460 Write the file header
A461–A462 A = cassette device number
A463–A464 Current device = cassette
A465 Zero A
A466–A468 Join the LIST code
A469 Zero A
A46A–A46B Zero X
A46C–A46E Write the file header
A46F–A470 Flag CLOSE
A471–A472 Flag data block
A473–A475 Write the leader
A476–A477 X = start of the Basic-program-area pointer
A478–A479 Save it as the cassette-buffer pointer
A47A–A47B A = block length
A47C–A47D Save the block length
A47E–A47F D = end of the Basic-program-area pointer
A480–A481 Figure the number of bytes left to write
A482–A483 Jump if done
A484–A487 Greater than FF bytes left?
A488–A489 Jump if greater than FF bytes left to write
A48A–A48B Save the block length
A48C–A48E Write a data block
A48F–A490 Loop until done
A491–A492 Flag EOF block
A493–A494 Block length = 0
A495–A497 Write the EOF block, turn off the cassette, and return

A498–A53D Color Basic CLOAD Command

A498–A499 Flag CLOSE
A49A–A49B CLOADM?
A49C–A49D Jump if CLOADM
A49E–A49F Clean up the stack
A4A0–A4A2 Evaluate the file name
A4A3–A4A5 Locate the file
A4A6–A4A8 Binary file?
A4A9–A4AA Jump if it's a binary file
A4AB–A4AD Basic file?
A4AE–A4AF Jump if it's not a Basic file
A4B0–A4B2 Do NEW
A4B3–A4B4 A = cassette device number
A4B5–A4B6 Current device = cassette
A4B7–A4B8 Flag OPEN"I"
A4B9–A4BB Read a block
A4BC–A4BE Join the command-mode code
A4BF–A4C1 Call the Extended Color Basic link
A4C2–A4C4 Do CLOSE
A4C5–A4C7 Jump to the command mode
A4C8–A4CA Basic file?
A4CB–A4CC Jump if it's a Basic file
A4CD–A4CF Display FM error message
A4D0–A4D2 Do NEW
A4D3–A4D5 Read the leader
A4D6–A4D7 X = start of the Basic-program-area pointer

Listing continued

Listing continued

A4D8–A4D9 Save it as the cassette-buffer pointer
A4DA–A4DB D = cassette-buffer pointer
A4DC Bump the MSB of the cassette-buffer pointer
A4DD–A4DF Do memory check
A4E0–A4E2 Read a block
A4E3–A4E4 Jump if IO error
A4E5–A4E6 A = block type
A4E7–A4E8 Jump if it's a file-header block
A4E9–A4EA Loop if it's a data block
A4EB–A4EC Save the cassette-buffer pointer as the end of the Basic-program-area pointer
A4ED–A4EE Turn off the cassette
A4EF–A4F1 X = OK message pointer
A4F2–A4F4 Display the message
A4F5–A4F7 Jump
A4F8–A4FA Do NEW
A4FB–A4FD Display IO error message
A4FE–A4FF Bump the ESP to the next character
A500–A501 Evaluate the file name
A502–A504 Locate the file
A505–A506 X = default offset
A507–A508 End of the Basic statement?
A509–A50A Jump if it's the end of the Basic statement
A50B–A50D Check the syntax
A50E–A510 X = offset
A511–A513 A = file type
A514–A515 Machine-language program?
A516–A517 Jump if it isn't a machine-language program
A518–A51A D = EXEC address
A51B–A51C U = adjusted EXEC address
A51D–A51E Save the EXEC address
A51F–A521 Is the file ASCII?
A522–A523 Jump if it's ASCII
A524–A526 D = starting address
A527–A528 X = adjusted starting address
A529–A52A Save it as the cassette-buffer pointer
A52B–A52D Read the leader
A52E–A530 Read a block
A531–A532 Jump if IO error
A533–A534 Save the new cassette-buffer pointer
A535–A536 Set the flags for the block type
A537–A538 Jump if it's a file-header block
A539–A53A Loop if it's a data block
A53B–A53D Turn off the cassette and return

A53E–A548 Color Basic EXEC Command

A53E–A53F Jump if it's the end of the Basic statement
A540–A542 X = EXEC address
A543–A544 Save the EXEC address
A545–A548 Jump to the EXEC address

A549–A553 Device Routine

A549–A54B Call the Extended Color Basic link
A54C–A54D A = current device number
A54E Cassette?
A54F–A550 Jump if it's the cassette
A551–A553 Jump

A554–A563 Video Routine

A554–A556 D = @ location
A557–A559 @ location > 01FF?
A55A–A55D FC error if the @ location > 01FF
A55E–A560 D = new cursor position
A561–A562 Save the new cursor position
A563 Return

A564–A577 Color Basic INKEY$ Command

A564–A565 A = last key pressed
A566–A567 Jump if a key was pressed
A568–A56A Scan the keyboard
A56B–A56C Last key pressed = 0
A56D–A56E Save the value of the key pressed
A56F–A572 Join the CHR$ code if a key has been pressed
A573–A574 String length flag = 0
A575–A577 Reset the string pointers

A578–A5A1 Evaluate-File-Name Routine

A578–A57A X = start of the file-name buffer
A57B–A57C File-name length = 0
A57D–A57E A = space
A57F–A580 Save a space in the buffer
A581–A583 Buffer filled with spaces?
A584–A585 Loop until the buffer has been filled with spaces
A586–A587 End of the Basic statement?
A588–A589 Jump if it's the end of the Basic statement
A58A–A58C Build a string entry
A58D–A58F X = string address and B = string length
A590–A592 U = start of the file-name buffer
A593–A594 File-name length = string length
A595–A596 Jump if the file name is a null string
A597–A599 Ignore
A598–A599 B = number of bytes to move
A59A–A59B Get a character to be moved
A59C–A59D Move it
A59E Block move done?
A59F–A5A0 Loop until the block move is done
A5A1 Return

A5A2–A5C4 Device Routine

A5A2–A5A4 Check the syntax
A5A5–A5A6 Character = #?
A5A7–A5A8 Jump if it's a #
A5A9–A5AA Bump the ESP to the next character
A5AB–A5AD Evaluate the expression
A5AE–A5B0 D = device number
A5B1 B = adjusted device number
A5B2–A5B3 Device number out of range?
A5B4–A5B5 Jump if the device number is out of range
A5B6 B = adjusted device number
A5B7–A5B8 Save the current device number
A5B9–A5BB Call the Extended Color Basic link
A5BC–A5BD Jump if it's the video display
A5BE–A5BF Jump if the device number is positive
A5C0–A5C1 Device number < −2?
A5C2–A5C3 Jump if the device number < −2
A5C4 Return

A5C5–A5CD Cassette Routine

A5C5–A5C6 Evaluate the file name
A5C7–A5C8 End of the Basic statement?
A5C9–A5CA Jump if it's the end of the Basic statement
A5CB–A5CD Display SN error message

A5CE–A5EB Color Basic EOF Command

A5CE–A5D0 Call the Extended Color Basic link
A5D1–A5D2 A = current device number
A5D3–A5D4 Save the current device number
A5D5–A5D6 Set the new device number
A5D7–A5D9 Check the device number
A5DA Zero B
A5DB–A5DC A = current device number
A5DD–A5DE Jump if it's the video display
A5DF–A5E0 Cassette?
A5E1–A5E2 Jump if it isn't the cassette
A5E3 B = −1
A5E4–A5E5 Get the old device number

Listing continued

SUPER PRO KEYBOARD

- **Only $69.95**
- Original key layout.
- No special software required.
- Fast, simple installation—no soldering.
- Individually boxed with full instructions.
- Professional, low profile, finished appearance.
- U.S. made—high quality, quad gold contacts.
- Smooth "Touch Typist" feel—no sagging.

Rainbow, April '83

A fine piece of hardware from Mark Data Products...It is super and it is professional too...If you are searching for a replacement keyboard, it is an excellent buy...

Hot CoCo, August '83

Like putting leather upholstery in your Volkswagen...Very impressed with the appearance and performance...Could easily pass as original equipment...Installation is very simple...

Color Computer Magazine, June '83

The installation procedure is well detailed and quite simple...Has a professional feel, reacts well to the touch... has held up to some purposeful pounding...

Color Computer News, June '83

Mark Data Products is well known to us "longtimers"... Every bit as finished as if Tandy had done it...The Mark Data Super-Pro is your best buy...The one that is in my CoCo to stay ..

Great Computer Software Also

• Adventure Games • Arcade Games and Utility Software

MD

Mark Data Products

24001 ALICIA PKWY., NO. 207 • MISSION VIEJO, CA 92691 • (714) 768-1551

All Orders: Please add $2.00 shipping and handling in the continental U.S. All others, add air shipping and $3.00 handling. California residents add 6% sales tax. Foreign orders please remit U.S. funds. Software authors—contact us for exciting program marketing details.

Listing continued

A5E6–A5E7	Reset the current device number
A5E8	D = true/false value
A5E9–A5EB	Save it as the current result

A5EC–A5F5 Color Basic SKIPF Command

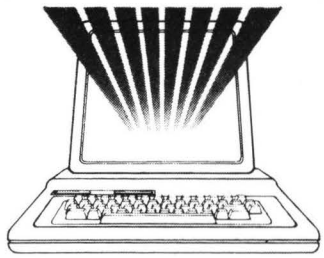
A5EC–A5ED	Evaluate the file name
A5EE–A5EF	Locate the file
A5F0–A5F2	Read a block
A5F3–A5F4	Jump if IO error
A5F5	Return

A5F6–A680 Color Basic OPEN Command

A5F6–A5F8	Call the Extended Color Basic link
A5F9–A5FB	Evaluate the expression
A5FC–A5FE	Get the "O" or the "I"
A5FF–A600	Save the "O" or the "I"
A601–A602	Evaluate the device number
A603–A605	Check the syntax
A606–A607	Evaluate the file name
A608–A609	A = current device number
A60A–A60B	Current device = video display
A60C–A60D	Get the "O" or the "I"
A60E–A60F	OPEN"I"?
A610–A611	Jump if OPEN"I"
A612–A613	OPEN"O"?
A614–A615	Jump if it's OPEN"O"
A616–A617	B = FM error code
A618–A61A	Ignore
A619–A61A	B = IO error code
A61B–A61D	Ignore
A61C–A61D	B = AO error code
A61E–A620	Ignore
A61F–A620	B = DN error code
A621–A623	Display the error message
A624	Bump the device number
A625–A626	Jump if it's the printer
A627–A628	Jump if it's the video display
A629–A62A	File already open?
A62B–A62D	A = file type
A62E–A630	Binary?
A631–A632	Jump if it's binary
A633–A634	Flag OPEN"I"
A635–A637	Read the leader and a block
A638–A639	Jump if IO error
A63A–A63B	Set the flags for the block type
A63C–A63D	Jump if it's a file-header block
A63E–A63F	Jump if it's an end-of-file block
A640–A641	A = block length
A642–A643	Loop if it's equal to zero
A644–A645	Save the block length
A646–A647	Jump
A648–A649	File already open?
A64A–A64B	Jump if a file is open
A64C–A64D	Locate the file
A64E–A64F	Jump if IO error
A650–A651	Cassette-buffer length = 0
A652–A654	X = start of the cassette buffer
A655–A656	Save it as the next location in the cassette buffer pointer
A657	Return
A658	Current device = cassette?
A659–A65A	Jump if it isn't the cassette
A65B	Bump the device number

Listing continued

SUPER SCREEN



- **A big 51 character by 24 line screen.**
- **Full upper and lower case characters.**
- **Easily combine text with hi-res graphics.**
- **PRINT @ is completely functional on the big screen.**
- **The powerful ON ERROR GOTO is fully implemented.**
- **Auto-key repeat for greater keyboard convenience.**
- **Control codes for additional functions.**
- **Works with 16K, 32K or 64K computers.**
- **Available on disc or cassette.**
- **Works with extended and/or disc BASIC.**

51 CHARACTERS BY 24 LINE DISPLAY

Super Screen is a powerful, machine language program that significantly upgrades the performance and usefulness of 16K or greater, Extended and Disc Basic Color Computers. The standard Color Computer display screen is totally inadequate for serious, personal or business applications so Super Screen replaces it with a brand new, 51 character wide by 24 line screen including full upper and lower case characters. Instead of a confusing checkerboard appearance, you now have true lower case letters along with a screen that is capable of displaying 1224 characters. The difference is startling! Your computer takes on new dimensions and can easily handle lines of text that were simply too long and complex to display on the old screen.

COMBINE TEXT WITH HI-RES GRAPHICS

You can now write truly professional looking programs that combine text with hi-res graphics. Super Screen allows you to create graphics displays with the Basic LINE, DRAW and CIRCLE statements and then notate the graphics with descriptive text. You can even use PRINT @ if you wish for greater programming convenience. Super Screen's versatility will amaze you.

PRINT @ IS FULLY IMPLEMENTED

The PRINT @ statement is a valuable asset to the programmer when formatting text on the screen. The standard Color Computer will report an error if you specify a location higher than 511 but Super Screen allows locations all the way to 1223! You get a big screen and a powerful formatting tool as well. Of course, Super Screen also supports the CLS command allowing you to clear the big screen using standard Basic syntax.

ON ERROR GOTO

That's right! Super Screen gives you a full implementation of ON ERROR GOTO including the ERR and ERL functions. Now you can trap errors and take corrective action to prevent crashed programs and lost data using the same standard syntax as other computers. The ON ERROR GOTO capability overcomes a serious deficiency of Color Computer Basic and greatly improves your capability to handle sophisticated tasks. All well written, 'user friendly' programs use error trapping techniques and yours can too! Now that's power!

AUTO KEY REPEAT

No more frustration as you edit a long line in your Basic program; just hold the space bar down and automatically step to the desired position in the line. Need a line of asterisks? Hold the key down and auto repeat will give them to you. Those of you who spend many hours at your keyboard will appreciate this outstanding addition to Super Screen's long list of impressive capabilities.

CONTROL CODES FOR ADDITIONAL FUNCTIONS

Super Screen recognizes several special control code characters that allow selection of block or underline, solid or blinking cursor and other functions. You can 'Home Up' the cursor or you may erase from the cursor to the end of a line or to the end of the screen just like many other computers. These special codes give you an extra dimension of versatility and convenience that put Super Screen in a class by itself.

AND MORE GOOD NEWS...

Super Screen comes with complete, well detailed instructions and is available on cassette or disc. It adjusts automatically to any 16K or greater, Extended or Disc Basic Color Computer or TDP-100 and uses only 2K of memory in addition to the screen memory reserved during power up. Guaranteed to be the most frequently used program in your software library...once you use it, you won't be without it! Super Screen's low price will really please you; only $29.95 on cassette or $32.95 on disc!

ACCOUNTING SYSTEM

The Mark Data Products accounting system is ideal for the small businessman needing a fast, efficient means to process income and expenses, prepare detailed reports and maintain most of the information required at tax time. The system is a family of programs which operate by means of a "menu" selection scheme. When the operator selects a task to perform, the computer loads a program designed to handle that task from the system disc. The system disc contains all of the programs required to create, update and maintain data files and prepare the necessary accounting reports including a transaction journal, a P&L or income report, an interim or trial balance and a balance sheet.

Up to 255 separate accounts may be defined and a single disc system can hold over 1,400 transactions. This system automatically enhances the monitor screen to a 51 character by 24 line display. 32K of memory is required along with an 80-column printer and one or more disc drives.

The MDP system:

- Is accurate, user friendly and simple to use.
- Is easy to customize for specific user requirements.
- Immediately updates the chart of accounts.
- Provides an audit trail.
- Includes end of period procedures.
- Is capable of future expandability.

This accounting software equals or exceeds higher priced packages for other computers and includes a detailed operating manual.

Requires 32K and a Single Disc Drive
PRICE: $99.95

ORDER ENTRY SYSTEM

The Mark Data Products sales order processing system provides a fast, efficient means to enter orders, print shipping papers and invoices, prepare sales reports, and monitor receivables. The system automatically enhances the monitor screen to a 51 character by 24 line display. 32K of memory is required along with an 80-column printer, and one or more disc drives.

The MDP order entry system is a family or programs which operate interactively by means of a "menu" selection scheme. Up to 900 products may be defined and a single disc system can hold over 600 transactions. When the operator selects a task to be performed, the computer loads a program designed to handle that task from the system disc. The system disc contains all of the programs required to create, update and maintain data files and prepare the necessary paperwork including shipping and invoice forms, daily sales reports, a monthly (or other period) sales report and a receivables report.

The MDP system:

- Is accurate, user friendly and simple to use.
- Is easy to customize for specific user requirements.
- Produces a traceable invoice.
- Handles receivables as well as closed orders.
- Is capable of future expandability.

This order entry software equals or exceeds higher priced packages for other computers and includes a detailed operating manual.

Requires 32K and a Single Disc Drive
PRICE: $99.95

THE MARK DATA PRODUCTS
SUPER PRO KEYBOARD...

The most popular replacement keyboard for your CoCo.

- Original key layout.
- No special software required.
- Fast, simple installation—no soldering.
- Individually boxed with full instructions.
- Professional, low profile, finished appearance.
- U.S. made—high quality, quad gold contacts.
- Smooth "Touch Typist" feel—no sagging.

Only $69.95

IMPORTANT NEW BOOKS

"Your Color Computer" by Doug Mosher. Over 300 pages of detailed information—A CoCo encyclopedia. $12.95.

"Programming the 6809" by Rodnay Zaks and William Labiak. One of the best 6809 machine language texts available—required reference material. $15.95.

64K Memory Expansion Kit

All parts and complete instructions

$64.95

WE STOCK SOFTLAW PRODUCTS

The VIP WRITER Text Processor is rated tops by Rainbow, Hot CoCo and Color Computer Magazine. After evaluation we rate it tops too. **Disc $59.95.**

MD

Mark Data Products

24001 ALICIA PKWY., NO. 207 • MISSION VIEJO, CA 92691 • (714) 768-1551

All Orders: Please add $2.00 shipping and handling in the continental U.S. All others, add air shipping and $3.00 handling. California residents add 6% sales tax. Foreign orders please remit U.S. funds. Software authors—Contact us for exciting program marketing details. We accept MasterCard and VISA. Distributed in Canada by Kelly Software

See List of Advertisers on page 130

Listing continued

A65C–A65E X = ASCII flag
A65F–A660 File already open?
A661–A662 Jump if a file is open
A663–A665 U = start of the cassette buffer
A666–A667 Save it
A668–A669 Save the file type
A66A–A66B Save the ASCII flag
A66C–A66E X = start of the file-name buffer
A66F–A671 Move the file header into the cassette buffer
A672–A673 Flag file-header block
A674–A675 A = block length
A676–A677 Save the block length
A678–A67A Write the file-header block
A67B–A67C A = OPEN"O" value
A67D–A67E Flag OPEN"O"
A67F–A680 Jump

A681–A6D0 Locate-File Routine

A681–A683 X = start of the cassette buffer
A684–A685 Save it
A686–A687 A = locate flag
A688 Locate?
A689–A68A Jump if no locate
A68B–A68D Clear the screen
A68E–A68F X = cursor position
A690–A691 B = S
A692–A693 Display the S
A694–A695 Save the new cursor position
A696–A697 Read the leader and a block
A698–A699 Set the flags for the block type and an error
A69A–A69B Jump if IO error or it isn't a file-header block
A69C–A69E X = start of the cassette buffer
A69F–A6A1 U = start of the file-name buffer
A6A2–A6A3 B = file-name length
A6A4–A6A5 Clear a hole on the stack
A6A6–A6A7 A = character from the cassette buffer
A6A8–A6AA Y = locate flag
A6AB–A6AC Locate?
A6AD–A6AE Jump if no locate
A6AF–A6B0 Current device = video display
A6B1–A6B3 Display the character
A6B4–A6B5 Subtract a file-name character from the cassette character
A6B6–A6B7 Combine it with the total
A6B8–A6B9 Save the new total
A6BA Comparison done?
A6BB–A6BC Loop until the comparison is done
A6BD–A6BE A = comparison total
A6BF–A6C0 Jump if the file names match
A6C1–A6C2 Locate the next file?
A6C3–A6C4 Jump if locate the next file
A6C5–A6C6 Check the block
A6C7–A6C8 Return if done
A6C9–A6CA Loop until done
A6CB–A6CC A = F
A6CD–A6CE Display the F
A6CF Zero A
A6D0 Return

A6D1–A6F2 Cassette Routine

A6D1–A6D3 Data block?
A6D4–A6D5 Jump if it's a data block
A6D6–A6D8 Read the leader
A6D9–A6DA Read the block
A6DB–A6DC Check the block type
A6DD–A6DE Loop until done
A6DF–A6E0 Read the leader and a block
A6E1–A6E2 Check the block type
A6E3–A6E4 Loop if it's a data block
A6E5–A6E6 Jump if IO error
A6E7–A6E8 A = block type
A6E9 Invert the block type
A6EA–A6EB Jump if it's a data block
A6EC Adjust for the error code
A6ED–A6EE Save the error code
A6EF–A6F0 Clean up the stack
A6F1–A6F2 Jump

A6F3–A700 Video Routine

A6F3–A6F5 A = character in the upper left of the video display
A6F6–A6F7 Invert the character
A6F8–A6F9 B = invert/noninvert flag
A6FA Noninvert?
A6FB–A6FC Jump if it's noninvert
A6FD–A6FF Display the new character
A700 Return

A701–A70A Cassette Routine

A701–A702 Turn on the cassette
A703–A704 Read a block
A705–A707 Turn off the cassette
A708–A709 B = cassette error code
A70A Return

A70B–A748 Read-Cassette-Block Routine

A70B–A70C Disable the interrupts
A70D–A70E Invert the video character
A70F–A710 X = start of the cassette buffer
A711 Zero the value
A712–A713 Read a bit
A714 Move it into A
A715–A716 Sync byte?
A717–A718 Loop until the sync byte has been read
A719–A71A Read the block type
A71B–A71C Save the block type
A71D–A71E Read the block length
A71F–A720 Save the block length
A721–A722 Figure the checksum
A723–A724 Save the checksum
A725–A726 A = block length
A727–A728 Save the block length
A729–A72A Jump if it's equal to zero
A72B–A72C Read a byte
A72D–A72E Save it in the buffer
A72F–A730 Memory error?
A731–A732 Jump if memory error
A733–A734 Figure the new checksum
A735–A736 Save the new checksum
A737–A738 Block done?
A739–A73A Loop until the block is done
A73B–A73C Read the checksum
A73D–A73E Checksums match?
A73F–A740 Jump if the checksums match
A741–A742 A = checksum-error code
A743–A745 Ignore
A744–A745 A = memory-error code
A746–A747 Save the error code
A748 Return

A749–A754 Read-Byte-from-the-Cassette Routine

A749–A74A A = number of bits to read
A74B–A74C Save the bit counter
A74D–A74E Read a bit
A74F Put the bit into A
A750–A751 All bits read?
A752–A753 Loop until all the bits have been read
A754 Return

END

KEY-264K

- ACCESS YOUR 64K RAM AS TWO 32K BANKS FROM BASIC IT'S LIKE HAVING TWO COMPUTERS IN ONE !!!
- HAVE SEPARATE PROGRAMS IN EACH BANK AND SWITCH INSTANTLY BETWEEN THEM WITH SIMPLE KEYSTROKES
- HAVE ONE LARGE PROGRAM THAT OCCUPIES BOTH BANKS
- RUN TWO PROGRAMS AT THE SAME TIME WITH FOREGROUND/BACKGROUND MULTI-TASKING
- ADDS 16 NEW COMMANDS TO EXTENDED OR DISK BASIC PASS VARIABLES BETWEEN BANKS - CALL SUBROUTINES ACROSS BANKS - VIEW TEXT OR GRAPHICS FROM EITHER BANK - COPY MEMORY ACROSS BANKS, START AND STOP MULTI-TASKING, ALL WITH BASIC COMMANDS
- INCLUDES 8 KEYBOARD COMMANDS TO ALLOW SWITCHING BANKS, MULTI-TASKING, BREAK, RESET, COLD STARTS AND DUPLICATING ONE BANK TO THE OTHER
- WORKS WITH CASSETTE OR DISK BASED SYSTEMS
- WORKS ON ANY 32K OR 64K COCO WITH EXTENDED OR DISK BASIC AND GOOD 64K MEMORY CHIPS

ORDER YOUR KEY-264K TODAY by sending check or money order for $39.95 (Cassette) or $44.95 (Disk) plus $2.00 postage U.S.A. ($5.00 outside U.S.A.) Mass. residents add 5% sales tax. COD (add $3.00), MASTERCARD, or VISA call (617) 263-1737

KEY COLOR SOFTWARE
P.O. BOX 360
HARVARD, MA. 01451

↙48

Color Computer Software

SUPER SLEUTH DISASSEMBLER (specify for 680x/6502 or for Z80/8080/5)	**Each $99-FLEX $101-OS/9**
Object-Only Version (680x/6502)	**$50.00**
For Color FLEX or OS/9 CoCo Sleuth	**$49.00**
CROSS-ASSEMBLERS (specify for 6800/1,6805, 6502,Z80, or 8080/5)	**Each $50-FLEX $55-OS/9**
DEBUGGING SIMULATORS (specify for 6805 or 6502)	**Each $75-FLEX $100-OS/9**
6502-TO-6809 ASSEMBLER XLATOR	**$75-FLEX $85-OS/9**
6800-9 & 6809 PIC XLATORS	**Both $50-FLEX $75-OS/9**

FULL SCREEN TSC XBASIC PROGRAMS FOR FLEX
(with full cursor control)

MAILING LIST	**$100**
INVENTORY/MRP	**$100**
TABULA RASA SPREADSHEET	**$100**

5.25" DSDD SOFT-SECTORED DISKETTES **$1.50 each in 50's**
(with hub rings, Tyvek jackets, write-protect tabs, labels)

Computer Systems Consultants, Inc.
1454 Latta Lane
Conyers, Georgia 30207
404-483-1717/4570

Programs in source on 5"/8" disk.
Detailed printed manuals provided.
Call for catalog or dealer information.
VISA and MASTERCARD accepted. US Funds Only.
Add 5% Shipping.
Add $4.00 per 50 Diskettes for Shipping.

" FLEX is a trademark of Technical Systems Consultants
" OS/9 is a trademark of Microware

↙223

Skyline MARKETING CORP

4510 W. Irving Park Rd. • Chicago, IL 60641
(312) 286-0762

VISA MasterCard

C.C. Three

A powerful 'electronic spreadsheet', a full-featured word processor, and a flexible database - for an unheard of low price! This may be the **ONLY SOFTWARE PACKAGE YOU'LL EVER NEED TO BUY** for your computer.

BOTH DISK AND TAPE VERSIONS OF ALL THREE PROGRAMS ARE INCLUDED (on tape) for the bargain price of $49.95! No need to pay for upgrades to disk later! Over 40 pages of documentation in an attractive vinyl binder. C. C. Writer and C. C. File require 16K, C. C. Calc needs 32K. All require Extended Color **BASIC**. Order yours **NOW!**

MYSTIC MANSION

MYSTIC MANSION – New!! You'll be hearing lots about this incredible **ALL GRAPHIC** adventure. Explore the mansion and escape from the island, if you can! This one is tough to solve, but you'll have fun trying. For 32K Disk only. **Disk $29.95.**

SUPER STATS – The most powerful statistics program available for the CoCo/TDP. Multiple linear regression, forecasting, mean, standard deviation, correlation coefficients, covariance, F and t tests. Powerful data editing and manipulation abilities. Data plotting, Student t tail areas, Gaussian cumulative probability distributions, and more. Flexible I/O (screen or printer, tape or disk). Very good documentation. 16K extended BASIC. **Cassette $29.95.**

↙531

UPS C.O.D. orders gladly accepted, $2.00 additional.
$2 shipping, handling, & insurance

↙See List of Advertisers on page 130

BY DON FLETCHER

GENERIC FOREIGN-LANGUAGE AID

¿ Habla Español? Let your CoCo be your tutor with this program for learning foreign vocabulary words.

As a struggling high-school student who needed to take a foreign language, I jumped into a Spanish class and was bombarded with multiple lists of Spanish vocabulary. After a few nights of studying with flash cards, I knew I needed a better way. This is when my CoCo came to the rescue. If you have the same problem with your foreign-language class, take a look at this program.

When you first run the program, you must input a list of vocabulary, rules, or whatever you want to study. Foreign translations are first and English next. For accents on certain Spanish words, I found it useful to use the inverse-video mode for that letter. When you are through, type "end" at the Foreign Translation prompt. You return to the menu and can make corrections (on a disk system only) or study the information.

If you need to make corrections, press 5 and follow the prompts.

You are now ready to study. Press 3 from the menu and the screen clears to ask if you want the incorrect-answer option. If you press Y, and then miss a word, the computer quizzes you on that word until you get it right. This option helps when you are first learning a list.

If you don't want to be tested again on a missed word, press N. The computer asks in which language you want to answer. Press 1 for English or 2 for foreign. The computer chooses a random word from the list and displays it. Type in the corresponding word in the other language (like flash cards) and press the enter key. The computer tells you if you are right or wrong. The correct answer appears if you input a wrong answer.

You can continue to study, or you might want to press R to return to the menu. If you go back to the menu, press 2 and the computer displays the number of words that you got right and wrong. If you got more right than wrong, it provides some music. You can return to study the list and the computer continues to keep score until you run the program again or load in another list.

To enhance the program you can add some different tunes in the See Score routine or create a way to add words to a previously saved list. You can also write a routine to prevent the problem of broken words caused by word-wrap. Table 1 lists the program's variables, and Table 2 describes the program line by line.

Remember, this program works with any language and for other things such as learning math formulas. The use of the LINE INPUT statement allows all kinds of information to be input, including commas, quotes, and extra spaces. ■

F$(x)	Foreign word
E$(x)	English word
LH$	Lists heading when creating list
S	Keeps track of number of words
F$(R)	Randomly chosen foreign word
E$(R)	Randomly chosen English word
N	Number of wrong answers
C	Number of correct answers

Table 1. Variable List

10–20	Clear memory and set variables
30–140	Menu
150–260	Create your list A disk file is made Foreign translation first Type End when finished
270–310	Check for Incorrect Answer Option
320–390	Check for Spanish or English input
400–570	Spanish input routine
490–510	Return to menu routine
580–750	English input routine
670–690	Return to menu routine
760–810	Score display routine
800	Check to see if more right than wrong If so, play tune
820–870	Routine to load in a previously saved list
890–1010	Update a previously saved list routine
1020	End

Table 2. Program Description

Address correspondence to Don Fletcher, 1910 Country Club Lane, Redlands, CA 92373.

System Requirements

32K RAM
Disk Basic Two Drives
16K RAM
Extended Color Basic

```
10 DRIVE0:PCLEAR1 :CLEAR1700:DIM
F$(500),E$(500),NF$(200),NE$(200
)
20 C=0:N=0:A=0:S=0:SS=0:P=RND(-T
IMER)
30 CLS:PRINT@ 71,"1)  CREATE LIS
T "
40 PRINT@135,"2)  SEE   SCORE"
50 PRINT@199,"3)  STUDY WORDS"
60 PRINT@263,"4)  LOAD   LIST"
70 PRINT@327,"5)  UPDATE LIST"
80 PRINT@359,"6)      END    "
90 PRINT@461,"(1-6)"
100 ZQ$=INKEY$:IF ZQ$=""THEN100
110 ZQ=VAL(ZQ$)
120 PLAY"O4T50FF"
130 ONZQ GOTO150,760,270,820,890
,1020
140 GOTO90
150 DRIVE1:CLS:INPUT"LIST HEADIN
G";LH$
160 IFLH$=""THEN150
170 OPEN"O",#1,LH$
180 CLS:PRINT@66,"FORIEGN TRANSL
ATION"
190 PRINT@129,;:LINEINPUTF$(A)
200 IFF$(A)="END"THEN CLOSE#1:DR
IVE0:GOTO30
210 PRINT@260,"ENGLISH TRANSLATI
ON"
220 PRINT@321,;:LINEINPUTE$(A)
230 WRITE#1,F$(A):WRITE#1,E$(A)
240 A=A+1
250 S=S+1
260 GOTO180
270 CLS:PRINT@162,"WOULD YOU LIK
E THE INCORRECT    ANSWER OPTION
"
280 PRINT@269,"(Y/N)"
290 A$=INKEY$:IFA$=""THEN290
300 PLAY"O4T50FF"
310 IFA$="Y"THENAO=1ELSEAO=2
320 CLS:PRINT@161,"WOULD YOU LIK
E TO ANSWER IN"
330 PRINT@193,"ENGLISH OR ESPANO
L"
340 PRINT@268,"(1/2)"
350 F$=INKEY$:IFF$=""THEN350
360 PLAY"O4T50FF"
370 IFF$="1"THEN580
380 IFF$="2"THEN400
390 GOTO340
400 R=RND(A)-1
410 CLS
420 SS=0
430 PRINT@SS,E$(R)
440 PRINT@160,;:LINEINPUTA$
450 IFA$=F$(R)THEN540
460 AA=0
470 N=N+1
480 PRINT@257,"THE CORRECT ANSWE
R IS:          "F$(R)
490 PRINT@448,"PRESS <SPACEBAR>
TO CONTINUE OR <R> TO RETURN";
500 I$=INKEY$:IFI$=""THEN500
510 IFI$="R"THEN30
520 IFAO=2THEN400
530 IFAA=1THEN400ELSE410
540 PRINT@352,"CORRECT!!!"
550 C=C+1
560 AA=1
570 GOTO490
580 R=RND(A)-1
590 CLS
600 SS=0
610 PRINT@SS,F$(R)
620 PRINT@160,;:LINEINPUTA$
630 IFA$=E$(R)THEN720
640 AA=0
650 N=N+1
660 PRINT@257,"THE CORRECT ANSWE
R IS:          "E$(R)
670 PRINT@448,"PRESS <SPACEBAR>
TO CONTINUE OR <R> TO RETURN";
680 I$=INKEY$:IFI$=""THEN680
690 IFI$="R"THEN30
700 IFAO=2THEN580
710 IFAA=1THEN580ELSE590
720 PRINT@352,"CORRECT!!!"
730 C=C+1
740 AA=1
750 GOTO670
760 CLS
770 PRINT@101,"CORRECT      INCOR
RECT"
780 PRINT@167,C:PRINT@179,N
790 FORX=1TO1500:NEXT
800 IFC>N THEN PLAY"T10O3GABO4T2
D"
810 GOTO30
820 CLS:A=0:DRIVE1
830 INPUT"TYPE IN FILE HEADING";
FH$
840 OPEN"I",#1,FH$
850 IFEOF(1)=-1THEN870
860 INPUT#1,F$(A):INPUT#1,E$(A):
PRINTF$(A),E$(A):A=A+1:GOTO850
870 CLOSE#1:DRIVE0:GOTO30
880 'UPDATE
890 DRIVE1:CLS:PRINT"FILE TO UPD
ATE":INPUT UF$:A=1
900 OPEN"I",#1,UF$
910 OPEN"O",#2,"NEW/DAT"
920 IFEOF(1)=-1THEN1010
930 INPUT#1,F$(A):CLS:PRINT"DATA
 ITEM:":PRINTF$(A):PRINT:PRINT"P
RESS <ENTER> IF NO CHANGE":PRINT
:PRINT:PRINT"CHANGE:"
940 LINEINPUT NF$(A):IF NF$(A)="
"THEN NF$(A)=F$(A)
950 WRITE#2,NF$(A)
960 F$(A)=NF$(A)
970 INPUT#1,E$(A):CLS:PRINT"DATA
 ITEM:":PRINTE$(A):PRINT:PRINT"P
RESS <ENTER> IF NO CHANGE":PRINT
:PRINT:PRINT"CHANGE:"
980 LINEINPUT NE$(A):IF NE$(A)="
"THEN NE$(A)=E$(A)
990 E$(A)=NE$(A)
1000 WRITE#2,NE$(A):A=A+1:GOTO92
0
1010 CLOSE#1,#2:KILL UF$+"/DAT":
RENAME"NEW/DAT"TO UF$+"/DAT":DRI
VE0:GOTO30
1020 CLS:RUN"DIR"
```

Program Listing 1. Language Aid for Disk

```
10 PCLEAR1
15 CLEAR5000:DIMF$(500).E$(500)
20 C=0:N=0:A=0:S=0:SS=0
25 CLS:PRINT@71,"1)  ADD WORDS"
30 PRINT@135,"2)  STUDY WORDS"
35 PRINT@199,"3)  SEE SCORE"
40 PRINT@263,"4)  SAVE WORDS"
45 PRINT@327,"5)  LOAD WORDS"
50 PRINT@460,;:INPUT"(1-5)";F
55 PLAY"O4T50FF"
60 ONF GOTO70,115,360,395,455
65 GOTO50
70 CLS
75 PRINT@130,"FORIEGN TRANSLATIO
N"
80 PRINT@194,;:INPUTF$(A)
85 IFF$(A)="END"THEN 25
90 PRINT@260,"ENGLISH TRANSLATIO
N"
95 PRINT@322,;:INPUTE$(A)
100 A=A+1
105 S=S+1
110 GOTO70
115 CLS:PRINT@162,"WOULD YOU LIK
E THE INCORRECT    ANSWER OPTION
"
120 PRINT@268,"(Y/N)"
125 A$=INKEY$:IFA$=""THEN125
130 PLAY"O4T50FF"
135 IFA$="Y"THENAO=1ELSEAO=2
140 CLS:PRINT@161,"WOULD YOU LIK
E TO ANSWER IN"
145 PRINT@193,"ENGLISH OR ESPANO
L"
150 PRINT@268,"(1/2)"
155 F$=INKEY$:IFF$=""THEN155
160 PLAY"O4T50FF"
165 IFF$="1"THEN270
170 IFF$="2"THEN180
175 GOTO150
180 R=RND(A)-1
185 CLS
190 IFLEN(E$(R))>28THENSS=128 EL
SESS=130
195 PRINT@SS,E$(R)
200 PRINT@192,;:INPUTA$
205 IFA$=F$(R)THEN250
210 AA=0
215 N=N+1
220 PRINT@256,"THE CORRECT ANSWE
R IS:          "F$(R)
225 PRINT@416,"PRESS <SPACEBAR>
TO CONTINUE OR <R> TO RETURN"
230 I$=INKEY$:IFI$=""THEN230
235 IFI$="R"THEN25
240 IFAO=2THEN180
245 IFAA=1THEN180ELSE185
250 PRINT@260,"CORRECT!!!"
255 C=C+1
260 AA=1
265 GOTO225
270 R=RND(A)-1
275 CLS
280 IFLEN(F$(R))>28THENSS=128 EL
SESS=130
285 PRINT@SS,F$(R)
290 PRINT@192,;:INPUTA$
295 IFA$=E$(R)THEN340
300 AA=0
305 N=N+1
310 PRINT@256,"THE CORRECT ANSWE
R IS:          "E$(R)
315 PRINT@416,"PRESS <SPACEBAR>
TO CONTINUE OR <R> TO RETURN"
320 I$=INKEY$:IFI$=""THEN320
325 IFI$="R"THEN25
330 IFAO=2THEN270
335 IFAA=1THEN270ELSE275
340 PRINT@260,"CORRECT!!!"
345 C=C+1
350 AA=1
355 GOTO315
360 CLS
365 PRINT@101,"CORRECT      INCOR
RECT"
370 PRINT@167,C:PRINT@179,N
375 FORX=1TO1500:NEXT
380 IFC>N THEN PLAY"T10O3GABO4T2
D"
385 GOTO25
390 'SAVE WORDS
395 CLS
400 PRINT"POSITION TAPE--PRESS P
LAY AND   RECORD"
405 INPUT"TYPE IN LIST #:PRESS <
ENTER>      ";TT$
410 OPEN"O",#-1,TT$
415 FORA=0 TO S-1
420 PRINT#-1,F$(A)
425 PRINTF$(A)
430 PRINT#-1,E$(A)
435 PRINTE$(A)
440 NEXT
445 CLOSE #-1
450 GOTO25
455 CLS
460 PRINT"REWIND RECORDER AND PR
ESS PLAY"
465 INPUT"TYPE IN LIST #:PRESS <
ENTER>      ";OO$
470 OPEN"I",#-1,OO$
475 A=1
480 IF EOF(-1)THEN505
485 INPUT #-1,F$(A),E$(A)
490 PRINTF$(A):PRINTE$(A)
495 A=A+1
500 GOTO 480
505 CLOSE #-1
510 GOTO25
```

Program Listing 2. Language Aid for Cassette

BY RICHARD D. UGLUM

EXQUISITE SCREEN SCROLL

Moving hi-res graphics in Basic is slow. But you can speed up the process by scrolling the screen.

If you've ever tried to move a large number of shapes on the screen in a Basic program, you know how slow high-resolution graphics can be. I'm going to show you a way to partially remedy that problem.

The idea is to move the whole screen instead of just shapes on it. The Color Computer lets you define the beginning of a high-resolution screen on any 512-byte boundary. By continuously incrementing the start location you can scroll the screen.

The 512-byte limit represents one twelfth of the screen at a time if you are using graphics mode 3 or 4. This makes for slightly jerky motion, but *everything* on the screen moves more or less simultaneously. Changing the start location of the screen actually involves changing the area of memory that is bit-mapped to the video image. I'll get back to this later.

Almost all the information needed to scroll the graphics screen is contained in *Getting Started with Color Basic*, section 4, part A. Most of the information I am using can be found on pages 258–263.

This section tells you how to set up the start of the graphics work area (actually of the screen bit-map) and how to set the graphics mode using only Color Basic. It tells you that you can start the graphics page on any 512-byte boundary. This divides a graphics page consisting of 6,144 bytes (mode 3 or 4) into 12 sections. Regardless of the start position, 12 of these minipages will be on the screen at one time. As the other graphics modes are mapped differently, they will display only three to six of these minipages. Graphics modes 3 and 4 seem to work the best, so I will stick to them for the rest of the discussion.

The starting location is designated by the page-select register. This register consists of a 7-bit word of information. If you do a PMODE 3,1 in Extended Basic, you would be at the first graphics page, which starts at location 1536 (non-disk system). This means that the page-select register was set to 3, or 0000011 in binary, as 1536 is the third 512-byte boundary in the system. More of this is explained in *Getting Started with Color Basic*.

The register is not represented by a single location, but by several; each bit in the register must be set separately. You can set the bits by sending POKE commands to the locations below:

Bit	Reset Bit	Set Bit
0	65478	65479
1	65480	65481
2	65482	65483
3	65484	65485
4	65486	65487
5	65488	65489
6	65490	65491

```
10 PCLEAR 8: PMODE 3,1: PCLS: SC
REEN 1,0
20 FOR I=1 TO 200
30 X=RND(255): Y=RND(191): C=RND
(4)
40 PSET(X,Y,C): NEXT I
50 PMODE 3,5: SCREEN 1,0
60 PCOPY 1 TO 5: PCOPY 2 TO 6
70 PCOPY 3 TO 7: PCOPY 4 TO 8
80 FOR I=15 TO 4 STEP -1
90 IF (I AND 1) = 1 THEN POKE 65
479,0 ELSE POKE 65478,0
100 IF (I AND 2) = 2 THEN POKE 6
5481,0 ELSE POKE 65480,0
110 IF (I AND 4) = 4 THEN POKE 6
5483,0 ELSE POKE 65482,0
120 IF (I AND 8) = 8 THEN POKE 6
5485,0 ELSE POKE 65484,0
130 IF (I AND 16) = 16 THEN POKE
 65487,0 ELSE POKE 65486,0
140 IF (I AND 32) = 32 THEN POKE
 65489,0 ELSE POKE 65488,0
150 IF (I AND 64) = 64 THEN POKE
 65491,0 ELSE POKE 65490,0
160 POKE 65472,0: POKE 65475,0:
POKE 65477,0
170 POKE 65314,224 OR (PEEK(6531
4) AND 7)
180 FOR J = 1 TO 200: NEXT J
190 NEXT I
200 GOTO 80
```

Program Listing 1. Rapid Screen Scroll

10	Set the graphics information
20–40	Set dot pattern
50–70	Copy 1–4 to 5–8
80	Decrement loop begin
90–150	Set start location of video bit map
160–170	Set graphics mode
180	Delay between scrolls
190	Decrement loop end
200	Reset for continuous scrolling

Table 1. Line Description for Listing 1

I	Pseudo-page-select register
J	Delay loop variable
X,Y,C	Dot pattern variables

Table 2. Variable List for Listing 1

System Requirements

16K RAM
Extended Color Basic
Joystick (optional)

✓See List of Advertisers on page 130

10–90	Set up screen and graphics
100–120	Draw and get image of flying saucer
130	Set initial X and Y coordinates
140–210	Poll joystick and update position, call screen reset when necessary
220	GOTO 140, continue
230–340	Reset screen bit-map image as program 1

Table 3. Line Description for Listing 2

I	Pseudo-page-select register
J,JI	Joystick variables
X,Y,C	Dot pattern variables
C	Used as ship position counter
X1,Y1	
X2,Y2	Ship position
A	Array that holds ship image

Table 4. Variable List for Listing 2

By using an integer to represent a pseudo-register, you can set bits 0–6 according to the bit configuration of the integer. Thus if integer I is 3 and the appropriate POKE commands are sent to the screen, the start location would be 1536. Now integer I can be incremented to 4, the POKE sequence followed, and the start location would be at the next 512-byte boundary.

In Program Listing 1, I is used as the pseudo-register. The program decrements I from 15 down to 4 then resets I to 15 and continues; this causes continuous scrolling on the screen. By setting I to 15 the screen bit-map start location is set to 7680. This is the beginning of graphics page 5 of Extended Basic. By decrementing I you scroll from the beginning of page 5 to the beginning of page 1. The algorithm for setting the bits in the page-select register to the configuration represented by integer I is as follows:

```
IF  (I AND 1) = 1    THEN set the bit
                     ELSE reset the bit
    (I AND 2) = 2
    (I AND 4) = 4
    (I AND 8) = 8
    (I AND 16) = 16
    (I AND 32) = 32
    (I AND 64) = 64
```

According to *Getting Started with Color Basic*, you must do two other things to set the graphics mode. These are fixed lines of code and are dependent upon which graphics modes you choose. The VDG register and the control register are set by the following, respectively.

```
POKE 65472,0: POKE 65475,0: POKE 65477,0
```

and

```
POKE 65314, 224 OR (PEEK(65314) AND 7)
```

These commands are covered in the back of the book.

The only bit of information not covered in the *Getting Started* book is how to decrement the memory location that holds the screen start location. This is not the start location covered above. The one above dictates which area of memory is bit-mapped to the video image. The second start location is used as a starting reference for the Extended Basic X,Y coordinate system. This resides in location 186 (MSB) and 187 (LSB).

"This is not an actual game, but a demonstration, so don't expect anything dazzling to happen."

Two must be added to the contents of location 186 for each 512-byte boundary you increment and two must be subtracted for each 512-byte boundary decremented. If this is not done, the X,Y coordinates will not be offset from the top of the screen. This was necessary in Program Listing 2 as it uses the coordinate system.

Listing 1 reserves eight pages of graphics memory, or 24 of the 512-byte segments. Graphics mode 3,1 is set and 200 dots are randomly put on the screen. The graphics mode is then set to page 5 and pages 1–4 are copied to 5–8. A decrementing loop is then run from 15 to 4 with the graphics screen start location being reset with each decrement.

This does all the scrolling from page 5 to page 1 in 12 steps. Instead of decrementing all the way down to boundary 3, the program resets I, the pseudo-register, to 15 and continues to cause continuous scrolling. This effect seems to work best in graphics mode 3 or 4, but you can use other modes.

What good is this technique anyway? Listing 2 gives an example of how you can use this technique in a game format. A flying saucer is put onto the dot pattern. You can control this saucer by the right joystick to go up and from side to side. The idea is to get the saucer to the top of the screen. Every time the ship goes up 32 lines in the Y coordinate, the screen scrolls down 16 lines. This is not an actual game, but a demonstration, so don't expect anything dazzling to happen.

When the screen scrolls, the appropriate number of lines must be subtracted from the saucer's Y coordinate, and the start location in locations 186 and 187 must be reset to keep the coordinate biased correctly. Changing 186 and 187 to different or random values could also cause some special effects. ■

Address correspondence to Richard Uglum, 1117A E. Potter Ave., Milwaukee, WI 53207.

```
10 PCLEAR 8: PMODE 3,1: PCLS: SC
REEN 1,0
20 DIM A(200)
30 I=14: C=1
40 FOR J=1 TO 200
50 X=RND(255): Y=RND(191): C=RND
(4)
60 PSET(X,Y,C): NEXT J
70 PMODE 3,5: SCREEN 1,0
80 PCOPY 1 TO 5: PCOPY 2 TO 6
90 PCOPY 3 TO 7: PCOPY 4 TO 8
100 DRAW "BM26,180;E2R3E1F1R3F2G
2L8H2"
110 PAINT(29,180),4,4
120 GET(22,176)-(44,186),A,G
130 X1=22: Y1=176: X2=44: Y2=186
140 J=JOYSTK(0): J1=JOYSTK(1)
150 IF J=63 THEN X1=X1+4: X2=X2+
4
160 IF J=0 THEN X1=X1-4: X2=X2-4
170 IF J1=0 THEN Y1=Y1-2: Y2=Y2-
2: C=C+1: IF C=17 THEN C=1: GOSU
B 230: Y1=Y1+16: Y2=Y2+16
180 IF X1<0 THEN X1=0: X2=22
190 IF X1>232 THEN X1=232: X2=25
4
200 IF Y1<0 THEN Y1=0: Y2=10
210 PUT(X1,Y1)-(X2,Y2),A,PSET
220 GOTO 140
230 IF (I AND 1) = 1 THEN POKE 6
5479,0 ELSE POKE 65478,0
240 IF (I AND 2) = 2 THEN POKE 6
5481,0 ELSE POKE 65480,0
250 IF (I AND 4) = 4 THEN POKE 6
5483,0 ELSE POKE 65482,0
260 IF (I AND 8) = 8 THEN POKE 6
5485,0 ELSE POKE 65484,0
270 IF (I AND 16) = 16 THEN POKE
 65487,0 ELSE POKE 65486,0
280 IF (I AND 32) = 32 THEN POKE
 65489,0 ELSE POKE 65488,0
290 IF (I AND 64) = 64 THEN POKE
 65491,0 ELSE POKE 65490,0
300 POKE 65472,0: POKE 65475,0:
POKE 65477,0
310 POKE 65314,224 OR (PEEK(6531
4) AND 7)
320 POKE 186,PEEK(186)-2
330 I=I-1: IF I=3 THEN FOR J=1 T
O 2000: NEXT J: STOP
340 RETURN
```

Program Listing 2. Demo Saucer Program

Survival Kit for the 80's

Don't program without it.

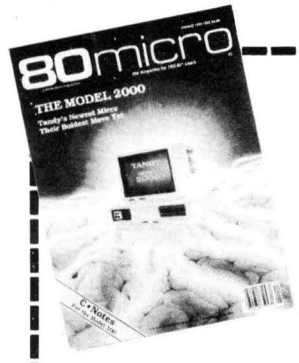
Technology in transition. Changes in hardware, software, disk systems, printers and modems—the list goes on. It's all happening so fast. How do you survive in the fast-paced computer world?

With the help of 80 MICRO, the largest single source of information for your TRS-80*. It's required reading for those who want to discover:

- •TUTORIALS—To help you become a better programmer whether you're a beginner or an old hand.
- •UTILITY PROGRAMS—Designed to let you get the most from your system.
- •NEW APPLICATIONS—In science, business, and the home.
- •NEW PRODUCT REVIEWS—Straightforward evaluations of state-of-the-art hardware and software that give you the facts before you buy.
- •NEW GAMES—Scores of innovative new games offering a real challenge and lots of fun.

With 80 MICRO in hand you won't just survive. You'll prosper in the Age of the Computer.

Subscribe today. Use the coupon below, the attached card, or call toll free **1-800-258-5473. In NH call 1-924-9471.**



Yes, I want to subscribe to **80 MICRO**. Send me 12 issues for only $24.97**. I understand that with payment enclosed or credit card order I will receive a free issue making a total of 13 issues for $24.97.

☐CHECK/MO ☐MC
☐VISA ☐AE ☐BILL ME

Card#_____

Signature_____ Exp. Date_____

Name_____

Address_____

City_____ State_____ Zip_____

Canada & Mexico $27.97/1 yr. only US Funds.
Foreign Surface $44.97/1 yr. only US Funds drawn on US Bank.
Please allow 6–8 weeks for delivery.

*TRS-80 is a trademark of Radio Shack, a division of Tandy Corp.
**This price voids all previous offers.

343F4

80 MICRO • PO Box 981 • Farmingdale, NY 11737

Instant CoCo

by Amee Eisenberg

Last month, in my column on cassette I/O (input/output) I promised to explain some techniques for getting a good load from a touchy tape. Most of my suggestions this month can be summed up in three words: Maintain your recorder. If your equipment's in bad shape, no tape is going to load.

Assuming that your recorder is O.K., the tape loading variable that you have the most control over is the volume. When presented with volume levels louder than it can handle, an audio amplifier clips, that is, it flattens out the extreme ends of the signal and produces noise. In an expensive stereo, this can blow your speakers; in a CCR-81, this yields garbage sound. If you try to load a clipped signal, you get an I/O error.

When your tape won't load, the first thing to fiddle with is the volume. If the computer finds the program, then gets an I/O error, turn down the volume. If the computer never finds the program, turn up the volume.

The quality of the tape will affect the ease with which you can load it. High-quality audio tape, like that used for Instant CoCo, doesn't add a lot of hiss to the recorded sound. Three-for-a-dollar bargain tapes do. Bad tape manifests itself in dropouts (sudden loss of volume), print-through (when you hear the recording on side B in the silences on side A), and skew (alternating between sharp and dull sound). When CSAVEing programs, cheap tape is no bargain.

A well-used tape recorder needs routine maintenance. When you listen to a tape, a dirty play/record head reveals itself by dimming the sound of the entire recording. If you're having trouble loading your tapes, try cleaning the head. If you use the recorder frequently do it once a week.

With the cassette door open and no cassette in the machine, press the play button. You'll see the play/record head and the capstan roller move into position. The play/record head is the squarish, silver block. Dip a cotton swab in some isopropyl alcohol (rubbing alcohol) and wipe the front of the head clean. If you haven't done this in a while, it might take three or four swabs to get it clean.

Head misalignment is the major cause of load failures. If the read/write head is out of vertical alignment with the cassette, the tape won't load. This is an easy problem to hear. Place a cassette in your recorder, press the play button and insert a small Phillips-head screwdriver through the hole above the rewind button onto the head-alignment screw. While listening to the tape, turn the screw. You will hear the sound get dimmer and brighter depending on which way you turn. When the sound is at its ear-piercing sharpest, the play head is correctly aligned with the tape.

If your tape recorder varies its speed while playing the tape, the tape won't load. The giveaway on this is that none of your tapes load. In extreme cases, you can hear the warble in the tape. The fix for this is to take it to your Radio Shack dealer and get it adjusted.

By listening to your computer tapes, you can diagnose the problems that are causing a loading failure. Sometimes a minor alignment or volume adjust makes the difference between hours of frustration and hours of pleasurable computing. ■

SIDE A

ARTICLE NAME/AUTHOR	FILE	PAGE#	SYSTEM
Copyright Statement	TITLE	---	All
What's Disk/McLaughlin	DIRECT	46	16K Disk
Possum Run/Bradbury	POSSUM	50	16K Ext
Give Your Disk Drive a One-Track Mind/Elfert	DIRDISK	54	16K Disk
PMODE0:POKE25,14POKE3584,0:NEW <enter> before running on 16K CoCos.			
Build a Better Error Trap.../Titchenal	ERRORBAS	60	16K Ext/Disk
Wordsearch/Tichborne	WORDSEAR	62	16K Ext
(m)	WS2		16K
CSAVEM "WS2",4096,10000,4096			
Colormania/Silverblatt (m)	PONG	74	16K Disk
CSAVEM "PONG",9728,10135,9728			
	DATASAVE		16K
	DRIVER		16K

SIDE B

Dental/Medical Bill Balancer/Weide	MEDICAL	80	16K Ext
POKE25,6:NEW <enter> before loading on 16K CoCos.			
	DENTAL		16K Ext
POKE25,6:NEW <enter> before loading on 16K CoCos.			
Video Van Gogh/Einam	COCOARTS	92	16K Ext
PCLEAR4:POKE7690,10:POKE7691,8:POKE7692,9:POKE7697,12:POKE7698,13			
K-Bug/Wuelzer	K-BUG	96	16K Ext
	K-LOAD		16K Ext
Mission:Mars/Mauro/Paplaskas	MARS	98	4K Basic
Generic Foreign-Language Aid/Fletcher	SPNISH32	118	32K Disk
	SPNISH16		16K Ext
The Educated Guest/Santee	SLOWGO	126	16K Ext
Elmer's Arcade/Ramella	TANTRUM1	17	16K Ext

The symbol (m) in the Article Name column indicates the program is machine language and must be loaded using the CLOADM command. Additional preparatory commands are listed under the article name where appropriate. CSAVEM addresses are listed for your use with the machine-language programs.

Instant CoCo Directory—March

1984 OFFICE AUTOMATION CONFERENCE®

LA's the place.

LOS ANGELES CONVENTION CENTER • CALIFORNIA
FEBRUARY 20-22, 1984

SPONSORED BY AMERICAN FEDERATION OF INFORMATION PROCESSING SOCIETIES, INC.

OFFICE AUTOMATION & YOU

The 1984 Office Automation Conference, "Office Automation & You," will highlight the most recent advances in OA, as they affect your role as a professional, manager, or user. This is YOUR conference.

OAC '84, the leading annual conference in the office automation industry, offers you the opportunity to:

* Learn more about OA; with over 45 Program Sessions, 8 in-depth Professional Development Seminars, and 6 Industry Workshops.
* Hear authoritative speakers, including Keynote—David T. Kearns, President & CEO of Xerox Corporation.
* See more than 150 major companies exhibiting their products & services.
* View numerous new products and developments introduced at OAC.
* Evaluate and compare specific products and services.

For additional information, complete the coupon below and send to: OAC '84, AFIPS, 1899 Preston White Drive, Reston, VA 22091.

Or, REGISTER TODAY via our toll-free number: 800-OAC-1984, using American Express, MasterCard, or Visa.

For more information, mail this coupon today

Name: _____

Title: _____

Company: _____

Address: _____

City: _____ State: _____ Zip: _____

Mail to:
OAC '84
AFIPS
1899 Preston White Dr.
Reston, VA 22091

162

The Educated Guest

SLOWGO OR NO-GO?

by Dr. Charles H. Santee

This month I will explore what is fast becoming the "sacred cow" of educational computing. Logo was developed by Seymour Papert, and first used as a method of manipulating robots for artificial intelligence. It consists of a set of commands called primitives that you can combine into groups called procedures. A procedure can consist of primitives or other procedures that are defined. Table 1 shows some of the primitives available in Logo.

The following Logo procedure draws a box:

```
TO BOX
    FD 30 RT 90
    FD 30 RT 90
    FD 30 RT 90
    FD 30 RT 90
END
```

Here is a program called Slowgo
That emulates graphics of Logo
Though the program is slow
You still get a show
And it's better than having a No-Go

Executing some Logo-type commands is more effective than describing them, so I have created a Basic program that emulates Logo. I call this program Slowgo because of its obvious limitations.

To use this program, type in the code carefully and run it. You see a title page and an instruction to name a procedure. For the first time simply press enter.

I have included three sample procedures for your review. It takes some time to convert these procedures, so wait for the next set of instructions. The name of each procedure appears as it is defined (converted to Basic DRAW statements). To run each procedure, press the number that appears in front of it.

After the procedure is drawn, you can press enter to get back to the selection menu where you can select a second procedure or press X to start over again. Experiment with the three procedures.

You can try to define your own procedure by first entering the name you want to give the procedure, then pressing enter. To enter the steps of the procedure, type each primitive separated by a space. Only the following primitives are available:

FORWARD or FD	BACK or BK
RIGHT or RT	LEFT or LT
PENCOLOR or PC	

Do not press enter until the procedure is completely defined. After finishing a procedure, you can press the enter key to enter another procedure, or press enter again to define the procedure for drawing. Here are some examples you might try.

```
Procedure Name: TRIANGLE
Procedure: FD 40 RT 120 FD 40
  RT 120 FD 40 RT 120
Procedure Name: VLETTER
Procedure: LT 30 FD 50 BK 50 RT
60 FD 50
```

Slowgo lets you use a procedure as long as it has been previously defined. Since SIXSIDE, BOX, and STAR are defined first, you can also use these in your procedures. Try these:

```
Procedure Name: MOVEBOX
Procedure:BOX RT 20 BOX RT 20 BOX RT
  20 BOX
Procedure Name: HONEY
Procedure:PC 2 SIXSIDE RT 120 SIXSIDE
  RT 120 SIXSIDE
```

Slowgo is a very limited application of Logo. You can only use 240 characters to define a procedure. Since Slowgo actually duplicates any previously

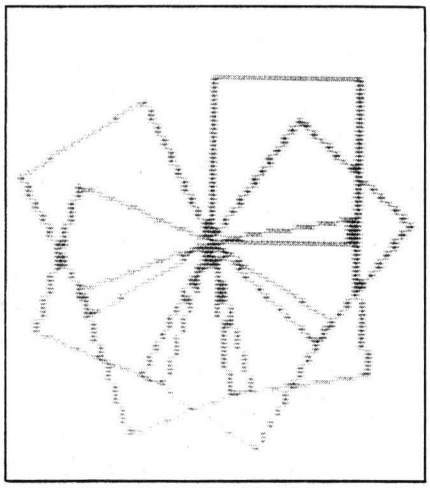


Photo 1. Procedure Name: MOVEBOX

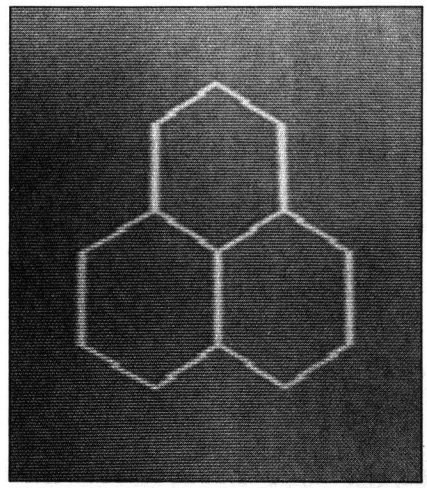


Photo 2. Procedure Name: HONEY

System Requirements

16K RAM
Extended Color Basic

defined procedure that is called you may run out of characters fast. Try this example:

Procedure Name: SIXSTAR
Procedure: STAR RT 20 STAR RT 20 STAR RT 20 STAR RT 20 STAR RT 20 STAR

The Radio Shack implementation of Logo is much more powerful. It supports the usual Logo primitives and some unique primitives such as HATCH. One of the powerful features of Logo is the ability to perform recursion. A recursive procedure is a procedure that calls itself. Here is an example:

```
TO SUN
  FD 30
  BK 30
  RT 1
  SUN
END
```

In this example, the procedure moves forward and back to draw a line and then turns right one degree. The procedure then calls itself and the process is repeated. This example is like a continuous loop in Basic. However, recursion can be used for more than continuous repetition. It is as though a first procedure is able to make a duplicate of itself, call that duplicate into action, and then return to the first procedure.

Other versions of Logo also support manipulation of lists of words with commands based on Lisp. It has been my experience that kids can learn the graphics portion of Logo with ease. Even the most rudimentary word manipulation with Lisp, however, can be quite complex. I would like to see a language for kids that allows easy manipulation and control of words the way turtle graphics can be used for experimentation.

Here, as promised, is my challenge for readers to respond. I will give a Slowgo award of recognition to the reader who can come up with the best Basic implementation of Logo. Perhaps you might allow for longer procedures, include more of the standard primitives such as REPEAT, and finally, recursion. I will talk more about these topics in a future column.

For you machine-language programmers, I offer the Mediumgo award of recognition for the best machine-language implementation.

The Fastgo award requires more

```
5 CLEAR 3000
10 REM ##### SLOWGO ######
20 REM ####### BY ########
30 REM # DR. C. H. SANTEE #
100 HD=0:P=0
110 READ D$:IF D$="END" THEN 120
 ELSE P=P+1:N$(P)=D$:READ P$(P):
P$(P)=P$(P)+" ":GOTO 110
120 REM
150 CLS:T$="SLOWGO":GOSUB 3000:T
$="by":GOSUB 3000:T$="DR. CHARLE
S H. SANTEE":GOSUB 3000:PRINT
160 PRINT@128,STRING$(192,32):PR
INT@128,"ENTER A name FOR A PROC
EDURE    OR PRESS return TO QUIT
"
165 INPUT D$:IF D$="" THEN CLS:P
RINT"PROCEDURES BEING DEFINED -
     please wait":PRINT:GOTO 200
170 P=P+1:N$(P)=D$
175 PRINT@224,STRING$(192,32):PR
INT @224,"ENTER THE PROCEDURE"
180 INPUT P$(P):P$(P)=P$(P)+" "
190 GOTO 160
200 FOR A=1 TO P
201 HD=360
202 DR$(A)="BM128,96"
205 P$=P$(A)
210 L=INSTR(P$," "):IF L=0 THEN
D$=P$:GOTO 250
215 IF LEN(P$)> 240 OR LEN(DR$(A
))>240  THEN GOSUB 890:GOTO 310
220 D$=LEFT$(P$,L-1):P$=RIGHT$(P
$,LEN(P$)-L)
230 IF D$="PC" OR D$="PENCOLOR"
THEN GOSUB 600
250 IF D$="RIGHT" OR D$="RT" THE
N T=1:GOSUB 700
260 IF D$="LEFT" OR D$="LT" THEN
 T=(-1):GOSUB 700
270 IF D$="FORWARD" OR D$="FD" T
HEN S1$="+":S2$="-":GOSUB 800
280 IF D$="BACK" OR D$="BK" THEN
 S1$="-":S2$="+":GOSUB 800
283 FOR NP=1 TO (A-1)
285 IF D$=N$(NP) THEN P$=P$(NP)+
" "+P$
287 NEXT
290 IF L<>0 THEN 210
300 PRINT N$(A);" IS DEFINED"
305 PLAY "T8L8O3CDEFGABO4C"
310 NEXT A
320 CLS:PRINT "PRESS THE NUMBER
OF A PROCEDURE":PRINT
330 FOR A=1 TO P:PRINT " ";A;N$(A
):NEXT A
332 PRINT:PRINT"PRESS x TO START
 AGAIN"
335 PRINT:PRINT"PRESS return TO
RETURN TO THIS  MENU AFTER VIEWI
NG PROCEDURE"
340 X$=INKEY$:IF X$="" THEN 340
342 IF X$="X" OR X$="x" THEN RUN
345 X=VAL(X$):IF X<1 OR X>P THEN
340
350 PMODE 3,1:PCLS:SCREEN 1,0:DR
AW DR$(X)
360 X$=INKEY$:IF X$<>CHR$(13) TH
EN 360
400 GOTO 320
600 L=INSTR(P$," "):D$=LEFT$(P$,
L-1):P$=RIGHT$(P$,LEN(P$)-L)
610 DR$(A)=DR$(A)+"C"+D$
620 RETURN
700 L=INSTR(P$," "):D$=LEFT$(P$,
L):P$=RIGHT$(P$,LEN(P$)-L):V=VAL
(D$)
710 HD=HD+T*V
720 IF HD>360 THEN HD=HD-360:GOT
O 720
730 RETURN
800 L=INSTR(P$," "):D$=LEFT$(P$,
L):P$=RIGHT$(P$,LEN(P$)-L):V=VAL
(D$)
810 AA=HD-INT(HD/90)*90
830 RA=AA/57.296
840 RB=(90-AA)/57.296
850 IF (HD=0 OR HD=360) THEN Q=1
 ELSE IF HD<90 THEN Q=1 ELSE IF
HD<180 THEN Q=2 ELSE IF HD<270 T
HEN Q=3 ELSE Q=4
860 H1=INT(SIN(RA)*V):V1=INT(SIN
(RB)*V)
865 H$=RIGHT$(STR$(H1),LEN(STR$(
H1))-1):V$=RIGHT$(STR$(V1),LEN(S
TR$(V1))-1)
870 ON Q GOSUB 900,920,940,960
880 RETURN
890 PLAY"T8L8O3":G$="DRTS LRTET
EI IETU H"
891 FOR G=1 TO 10
892 PL$=STR$(11-G):PLAY PL$:PRIN
T MID$(G$,11-G,1);:PRINT MID$(G$
,21-G,1);
894 NEXT:PRINT:N$(A)=N$(A)+" * *
 UNFINISHED * *"
896 PLAY "O2T2L4CP16FP16L6AP16GP
16FP16L2DP16DP16CP16CP16L1F"
898 RETURN
900 DR$(A)=DR$(A)+"M"+S1$+H$+","
+S2$+V$:RETURN
920 DR$(A)=DR$(A)+"M"+S1$+V$+","
+S1$+H$:RETURN
940 DR$(A)=DR$(A)+"M"+S2$+H$+","
+S1$+V$:RETURN
960 DR$(A)=DR$(A)+"M"+S2$+V$+","
+S2$+H$:RETURN
1000 DATA SIXSIDE,FD 30 RT 60 FD
 30 RT 60 FD 30 RT 60 FD 30 RT 6
0 FD 30 RT 60 FD 30 RT 60
1020 DATA BOX,FD 40 RT 90 FD 40
RT 90 FD 40 RT 90 FD 40 RT 90
1060 DATA STAR,FD 50 RT 144 FD 5
0 RT 144 FD 50 RT 144 FD 50 RT 1
44 FD 50 RT 144
2000 DATA END
3000 B=16-INT(LEN(T$)/2):PRINTTA
B(B);T$:RETURN
```

Program Listing. Slowgo

Primitive	Abbreviation	Purpose	Example
FORWARD	FD	move forward 'x' spaces	FD 10
BACK	BK	move back 'x' spaces	BK 40
RIGHT	RT	turn right 'x' degrees	RT 90
LEFT	LT	turn left 'x' degrees	LT 45
PENCOLOR	PC	lets you draw with colors	PC 2

Table 1. Logo Primitives

creativity. I would like to see a language or program that allows kids to easily manipulate words. How about creating the following types of commands:

Type Thing
Type Action
Type Description
Make Sentence With 'Thing', 'Action', and 'Description'
Move Sentence Up
Center Sentence
Make Billboard

Nonprogrammers, don't be scared away from this task. I am an advocate of top-down development: Start with a regular English-language description of the task. What you need first is an understandable description of what this type of program should do. If you think you can describe a language manipulation program that is easy and fun, let me hear from you.

There have been many claims about the benefits of using Logo. This ranges from claiming that Logo can be used to teach structured programming, to claiming that using Logo enhances problem-solving skills in children. I would like to see some hard evidence for these claims. Specifically, I would like to see evidence that answers the following questions:

"What evidence exists that Logo is more effective than alternative approaches?"

- What evidence exists that children (or adults) who are taught to use Logo acquire skills beyond those required to use the language?
- Does Logo enhance a generalized problem-solving skill?
- Does using Logo enable students to learn new computer languages more easily?
- Do students who first learn Logo apply a more structured approach to programming in other languages?
- Can students learn other academic skills through a course in Logo?
- What evidence exists that Logo is more effective than alternative approaches?
- Is Logo more effective than Basic (or other languages) as a first computer language?
- When academic skills are taught can Logo be more effective than traditional approaches?
- Are the problem-solving skills acquired through learning Logo more effective than alternative problem solving techniques?

I am skeptical about the rash of claims and mystique that is beginning to surround the wholesale adaptation of Logo in education. I will, however, reserve further judgement until I have had a good chance to review the literature. If you are aware of critical evidence either through direct experience or reading, please let me know. ■

Contact Charles Santee c/o HOT CoCo, *80 Pine St., Peterborough, NH 03458.*

↙See List of Advertisers on page 130

Index to Advertisers

*This advertiser prefers to be contacted directly.

For further information from our advertisers, please use the Reader Service card.

TRS-80+ MOD I, III, COCO, TI99/4a
TIMEX 1000, OSBORNE, others

GOLD PLUG - 80

Eliminate disk reboots and data loss due to oxidized contacts at the card edge connectors.
GOLD PLUG 80 solders to the board edge connector. Use your existing cables. (if gold plated)

COCO Disk Module (2)	new	$16.95
Ground tab extensions		INCL
Disk Drives (all R.S.)	SPECIAL PRICE	$7.95
Gold Disk Cable 2 Drive		29.95
Four Drive Cable		39.95

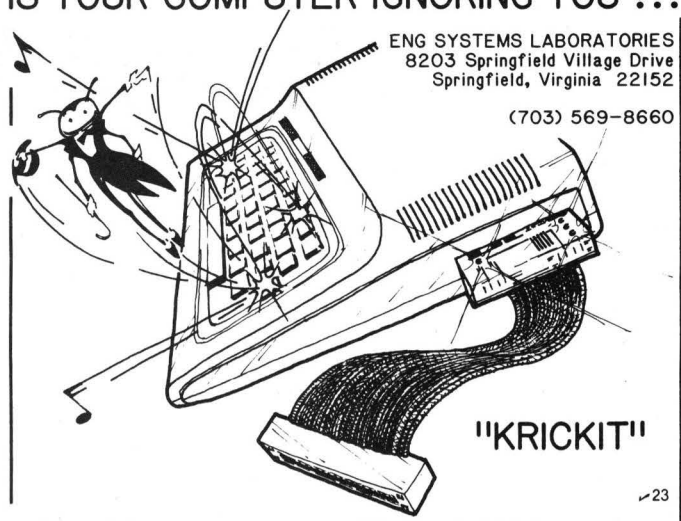
USA shipping $1.45 Can/Mex $4.
Foreign $7. **Don't wait any longer** TEXAS 5% TAX

Available at your favorite dealer or order direct from

E.A.P. CO.
P.O. BOX 14
KELLER, TEXAS 76248
(817) 498-4242 MC/VISA

+ trademark Tandy Corp ↙216

IS YOUR COMPUTER IGNORING YOU ???



ENG SYSTEMS LABORATORIES
8203 Springfield Village Drive
Springfield, Virginia 22152
(703) 569-8660

↙23

You've just typed in the Gettysburg Address and now you see that your computer only heard every other word!

THE KRICKIT CHIRPS ONLY WHEN YOUR COMPUTER HAS CAUGHT THE KEY. With the Krickit, there is no wasted effort pounding keys to make sure they contact or looking at the screen after every character. You can keep your eyes on the listing and not lose your place.

Works great with text editors and DBM's (Telewriter-64 and Homebase, to name a few) or when entering programs with just the BASIC line-editor in your computer. Just plugs in! Needs no software or hardware modifications. Built-in speaker! Uses no CPU time! Your computer has better things to do than sound key beeps.

The Krickit has other valuable features, too. A convenient switch controls the Cartridge Interrupt line. Instead of clumsy, messy tape on fingers 7 or 8, just flip the switch to access Basic with a game pak installed. It also has a more accessible reset switch, lighted power indicators, gold edge-fingers and an extension cable.

We are sure that after you try the Krickit you will never want to be without it. Take 30 days to decide you like it or return it for a full refund. 24-hour order line. Order yours today for only $59.95 (plus shipping and handling). COD, Visa, and Mastercard accepted.

FOR QUALITY EDUCATIONAL SOFTWARE LOOK FOR THIS EMBLEM

SOFTWARE AVAILABLE FOR:
- TRS-80® COLOR COMPUTER
- ATARI® 400/800/1200
- COMMODORE 64®
- APPLE® &
- VIC 20®

A T C E PROGRAM

Send for Free Catalog

DEALER INQUIRES INVITED

T.C.E. ↙389
P.O. Box 2477 Gaithersburg, Maryland 20879 (301) 963-3848

↙See List of Advertisers on page 130

Reader's Forum

Put Some Variety Into Those Random Numbers

How random are the numbers that you generate with Color Basic's RND command? Radio Shack hasn't given much information about the random function. RND isn't even listed in their *Quick Reference Guide*. Here is a one-liner that shows you how random the numbers really are.

```
10 PRINT RND(5000):GOTO 10
```

Every time I run this program on my computer, after a cold start, I get the same set of numbers. Try it and see what you get. This is just one of the CoCo's peculiarities.

There are several solutions to the problem. Here is one: Add a line 5 to the one-liner above.

```
5 POKE 280, PEEK(275)
```

This is what you are doing: RAM location 280 is one of the seed addresses for the RND function. The 275 address is one of the memory locations that continually changes value, like a clock, as part of the TIMER operation. So, you are POKEing a fresh numeral right where RND lives every time you use this POKE/PEEK technique.

Use line 5, with your own line number, somewhere near the start of any program that has a random-number generation routine. One time is sufficient for each program, no matter how many times RND is used.

Tom Garcia
Tucson, AZ

Word Processor Utility

This short utility (Program Listing 1) is useful in word or text processors based on string entering of text, as found in *HOT CoCo*, June 1983, p. 36.

The routine emits a sound when entering a text at so many characters under any inputting condition, either by INPUT, LINE INPUT, or typing on the screen.

Given in Assembly language, the routine uses the EDTASM+ followed by a Basic driver (Program Listing 2). It is position independent and can be located anywhere in memory. You can obtain the sound after you input so many characters by POKEing in address 492, the number of characters desired plus one. For example, to obtain the sound after 200 characters, POKE492,201.

Alain Dussault
Laval, Quebec
Canada

```
00100          ORG      1600      (16200 for 16K, 32700 for
                                  32K)
00110          SETDP    0
00120 INIT     LEAX     KBRD,PCR
00130          STX      $16B
00140          LDA      #$7E
00150          STA      $16A
00160          RTS
00170 KBRD     CMPB     492
00180          BNE      KB1
00190          PSHS     D,X,Y,U
00200          LDA      #100
00210          STA      $8C
00220          LDD      #4
00230          JSR      $A954
00240          PULS     D,X,Y,U,PC
00250 KB1      RTS
00260          END
00270
00280
```

Program Listing 1

```
10000 W=1536:FORK=0TO37:READB$:P
OKEW+K,VAL("&H+B$):NEXT:EXECW:PO
KE492,201:RETURN
10010 DATA30,8D,00,09,BF,01,6B,8
6,7E,B7,01,6A,39,34,16,F1,01,EC,
27,02,35,96,34,60,86,64,97,8C,CC
,00,04,BD,A9,54,35,60,20,EE
```

Program Listing 2

Easy LINEINPUT In Standard Color Basic

Here's a short routine that simulates the LINEINPUT command on a standard Color Basic CoCo:

```
5000 IN$ = "":I = 733:EXEC41872
5010 J = PEEK(I):IFJ = 0THENRETURNELSEIN$ = IN$ + CHR$(J):
I = I + 1:IFI<982THEN5010ELSERETURN
```

The simplicity of this routine is due to the use of an undocumented ROM routine at $A390. (This routine might not work on newer CoCos because of this—eds.) These two lines input IN$ to a maximum length of 249 characters. If you enter a string of such length, however, there is a short delay after input is complete. No garbage collection will ever occur during input, though.

Be sure the program calling this subroutine has cleared enough string space. Quotes, commas, and colons are allowed within IN$. The subroutine uses I and J, common scratch variables. Input begins at the current cursor posi-

tion. You'll see no prompt or even a question mark. Pressing break when running this subroutine works like enter, so you must press break twice to stop the program.

Ronny Ong
Arlington, TX

Create a Cassette Menu with TIMER

You can use the TIMER function of Extended Basic to create a menu for use with the cassette. Pack your favorite programs in a long tape (like 60 minutes), and calculate the value of TIMER needed to get to each program by the fast forward button. Then you can use the MOTOR on and off function to get to each program quickly.

Jae Nam Noh
Orlando, FL

Using a Brand X Recorder With the CoCo

Although Radio Shack recommends using their cassette recorder as a data-storage device for the Color Computer, many people prefer to use a recorder that they already own. My experience with several inexpensive recorders indicates that usually only a minor modification is required to achieve reliable recording.

It appears that with many off-brands, the voltage signal sent from the computer overwhelms the recorder's automatic level-control circuit. This results in distortion of the signal sent to the tape. The solution to this problem lies in adding a resistor in the line connecting the output of the computer to the input of the cassette, reducing the signal and bringing it back to acceptable levels.

I have found that 150k-ohms works well with my recorder, but the value used is a function of the characteristics of the recorder and should be established through experimentation. You can actually hear the quality difference in the before and after by listening to the audio signal.

A convenient way to attach the resistor (1/8 watt) is to solder it between the center posts of a spare male/female phono-plug pair connecting the outside shield with a stiff wire. Once attached and debugged, the pair is held together on-end with a piece of heat-shrinkable tubing.

This minor modification combined with good data tapes has given me two years of CoCo computing free of tape-reliability problems using a Brand X recorder.

Thomas Szlucha
Fairport, NY

Better Keyboard Response

Here's a one liner (Program Listing 3) to solve your keyboard response problems. If you are using machine-language programs that originated with the older CoCos (1.1 ROM) and are now using them on the new 64K and CoCo 2 computers (1.2 ROM), you probably notice the poor keyboard response on some of these programs.

Radio Shack changed the vector addresses when using the POLCAT routine. This one liner checks for all the references to the old vector address and changes them to the new address in the 1.2 ROM.

CLOADM your machine-language program, but do not execute. Then type in the listing and run it. You don't even need to know the starting or ending address of your program. When you run the program you see "working" displayed on the screen for a few moments. When the OK prompt appears, go ahead and execute your program.

Guier Wright
Peter Paplaskas
HOT CoCo *Staff*

```
10 PRINT"WORKING":FORT=(PEEK(157
)*256+PEEK(158)) TO 32766:IF PEE
K(T)=161 AND PEEK(T+1)=193 THEN
POKE T+1,203:NEXT:ELSE NEXT
```

Program Listing 3

HOT CoCo *pays $25 for each Reader's-Forum submission used. In the case of duplicate submissions, selection is based on the earliest postmark.*

✓See List of Advertisers on page 130

Doctor ASCII

by Richard E. Esposito

Due to the unexpectedly large number of inquiries to Doctor ASCII, we must ask that you enclose a self-addressed, stamped envelope if you want an immediate reply. Please be patient, as the doctor must answer many questions.

Send your questions to Doctor ASCII, c/o HOT CoCo, *Pine St., Peterborough, NH 03458.*

Q. I have a 64K CoCo with Extended Basic. I wish to LLIST to the CGP-115 Color Graphic Printer with 32 characters per line, *not* 40. Is there a POKE that will do this? I've tried POKE115,32 with no results.

Daniel W. Flynn Jr.
Oscoda, MI

A. The explanation of the POKE115,## in the CoCo's documentation is misleading. The value at address 155 tells the CoCo that it should pause after ## characters have been sent to the printer to allow sufficient time for the print head to return to the left side of the page. It does not generate a carriage return and line feed after it has encountered a line of ## characters as one is led to believe. Since most printers come with their own built-in RAM buffer, this POKE is often ignored.

If you save a program in ASCII to tape or disk, Program Listing 1 will let you print it out in a 32-character format.

Q. Where can I get a screen-dump program for my C. Itoh Prowriter printer?

R. Jones
Middletown, PA

A. The Radio Shack one might work. Your Prowriter is made by TEC, the maker of the Radio Shack LP VIII. Since I do not have access to a Prowriter, I could not tell for sure. Custom Software Engineering, 807 Minuteman Causeway (D-2), Cocoa Beach, FL 32931, markets a screen-dump program for your printer at $9.95.

```
10 DEV=1: 'CHANGE TO -1 FOR TAPE
20 INPUT"asciiFILENAME";N$
30 OPEN"I",DEV,N$
40 LINE INPUT#DEV,I$
50 IF EOF(DEV)=-1 THEN CLOSE#DEV
: END
60 IF LEN(I$)<=32 THEN PRINT#-2,
I$ ELSE PRINT#-2,LEFT$(I$,32): I
$=MID$(I$,33): GOTO 60
70 GOTO 40
```

Program Listing 1. Print Routine for 32-Character Format on a CGP-115 Printer

Q. A few years ago, I purchased a 4K CoCo, and last December I upgraded it to 32K and Extended Basic.

When I got it back, the CoCo's mains had been reversed, and the antenna switch became live. Once I sorted this out, the CoCo worked, but all the colors were wrong. Also, when demonstrating this to the Radio Shack people, we found that some of their machines were wrong, too. Is there any way I can fix the problem myself?

Hedley Jones
Epsom, Surrey
England

A. Order the *Color Computer Technical Reference Manual*, RS number 26-3193, $14.95, and write to Motorola Semiconductors, 3501 Ed Bluestein Blvd., Austin, TX 78721, for specification sheets on CoCo's vital parts such as the MC6847, MC6821, MC1372, MC6883, and MC6809E.

Q. I made my own 64K upgrade on my E board CoCo several months ago using the standard procedure. When I use the program listed in the July 1983 issue of *HOT CoCo* to move into 64K, I do indeed move into 64K mode and can write above the ROMs by POKEs or by transferring machine-language routines to this area from low RAM. My problem is that I cannot overwrite my Basic ROMs and 64K commercial programs such as Super Color Writer, and your "40K Custom Color Basic" will not work correctly.

Tim Harris
Ames, IA

A. You have the symptoms of what Radio Shack calls a 32K E board machine. Your problem lies in one of two areas. First, in reference to "64K Modification" (*HOT CoCo*, July 1983, p. 44), retrace the steps starting with the first one in the center column at the bottom of p. 46. When doing this, note that the pins in Fig. 2 should have been numbered counterclockwise from the notch. If you have it wired correctly, then I suspect a partly bad 74LS02 or 74LS138. This modification uses NOR gates in these chips that are otherwise not used, and the fact that they are bad would not show up until this modification is made.

Q. After using my CCR-81 to record a program for my CoCo, it is disturbing to find a hidden bug. I often list the program, correct the error, and then attempt to CSAVE it. Doing this usually results in trouble, namely, an erasure of the program. Is there a technique to solve my problem? Is there a product that will let me alter my programs?

Craig McCormick
Fairfax, VA

A. The CCR-81 tape recorder, like all tape drives, is a sequential storage device. If you want to access the fifth file on a tape, you must go past the four that precede it. One way to alleviate the problems associated with tape storage on your CoCo is to space your files apart, keeping track of the reading of the tape counter at the beginning of each file. This way, when you change a program and make it longer to correct an error and you resave it starting at the same tape counter number, there is room on the tape for the additional code. If you write programs that use POKEs, EXECs, or USR routines, it is a good idea to CSAVE your program to tape before running it. An error in your program could wipe out Basic's RAM pointers, effectively erasing your program from memory.

The product that would help alleviate your problems most is a disk drive because it is much faster than tape and it is a random-access device. You are usually not concerned about where a program is physically located on a disk. It need not even be stored in contiguous granules because the operating system takes care of this housekeeping for you. If a new version of a program is larger than the old one, additional granules will be allocated to that file following a save.

Q. I've purchased a software package that lets me enter a TRS-80 Model I/III program into my CoCo using the joystick port. It works fine except that when I list the entered program, I get back a bunch of garbage. I was wondering if this was caused by the same reason the new MC-10's programs won't load into the CoCo: different tokens. If so, where can I get a list of tokens so that I can develop a conversion program as I have seen for the MC-10? Any suggestions would be greatly appreciated.

Bill Smith
Pasadena, TX

A. If your program works with ASCII files, then no conversion is necessary. If not, look at the early issues of *80 Micro*. A comprehensive index appears in the *1983 Special Anniversary Issue*. You might also get the information on tokens from a local TRS-80 user's group.

Q. I have a 32K F board CoCo with one disk drive. I have heard that I have the same board as the new 64K CoCo except for a ROM. Will my computer support the new operating system, OS-9? If so, will it use the full 64K?

Brian Spellman
Clinton, MD

A. You already have a 64K computer, and yes, it will support OS-9. You can also run FLEX and many other programs that will use the 64K with the standard Disk Color Basic or even cassette. The odds are that even your case is white like the new computer's under its silver paint.

Q. I purchased an adventure game called Bedlam from Radio Shack about four months ago and have never been able to escape. Can you help me? I have gotten the blue pill, window hook, and green key. Where does the green key go? What is the blue pill for, and how do you get out? Please tell me some clues. I'm going crazy.

Bob Adams
Westerly, RI

```
10 REM PROGRAM THE SAM TO START
THE SCREEN DISPLAY AT 1536 FOR T
APE OR 3584 WITH DISK SYSTEM
20 IF PEEK(49152)=68THEN W1=7 EL
SE W1=3
30 POKE 65478-((W1 AND 1)=1),0
40 POKE 65480-((W1 AND 2)=2),0
50 POKE 65482-((W1 AND 4)=4),0
60 POKE 65484-((W1 AND 8)=8),0
70 POKE 65486-((W1 AND 16)=16),0
80 POKE 65488-((W1 AND 32)=32),0
90 POKE 65490-((W1 AND 64)=64),0
100 A$=INKEY$:IFA$=""THEN100
110 IF A$="B" THEN W1=W1-1 ELSE
W1=W1+1
120 IF W1<0 THEN W1=127:SOUND10,
1 ELSE IF W1>127 THEN W1=0:SOU
ND10,1
130 GOTO 30
```

Program Listing 2. Adventure Peeking Routine

A. Program Listing 2 gives you a window on memory. With it, you can peer into the innards of your adventures and find all the key words and phrases that are presently eluding you.

If you have an adventure game such as Bedlam that EXECs in low memory, first PCLEAR8 or offset load the adventure so that when you load in my program it does not overwrite part of the adventure game. With this program running, the up- and down-arrow keys change the area of memory that corresponds to the display screen. Since the clues that you want will correspond to the ASCII codes for text, you can read them directly out of memory and even compile your own Bedlam dictionary if you wish.

Q. I'm considering upgrading my 16K Extended Basic Color Computer by purchasing the new Radio Shack 64K RAM upgrade and the keyboard upgrade. Assuming I did, how would the resulting 64K unit differ from the new 64K Extended Basic Color Computer 2? Would the OS-9 system and the Color Computer 2 disk 0 work as well with the upgraded unit as with the Color Computer 2 version?

James K. Boudreau
Marietta, GA

A. The new 64K machine is not a Color Computer 2. The Color Computer 2 is the machine that is marketed with 16K in the smaller case. As to comparing an upgraded machine with the new 64K one, the only functional difference would be that the new machine might have newer versions of the ROMs, namely Color Basic 1.2 and Extended Basic 1.1. OS-9, FLEX, and other 64K programs will work with either set of ROMs.

If you buy a disk, Radio Shack has created a real dilemma. If you buy one that has the new 1.1 Disk Basic ROM in

its controller cartridge, you can boot OS-9 with a single disk. With the older disk ROM, you need to run a program called * from a second disk first and then insert the OS-9 disk when prompted. The problem that arises with the new ROM is that many existing disk programs, including ones sold by Radio Shack, will not work with the new ROM. If you are in the market for new disk software, be sure to check that it is compatible with the ROM that you get.

Q. I've read the article "64K Modification," which appeared in the July issue of *HOT CoCo*, but I'm still confused. What is required to modify my 32K Extended Basic 1.0 machine, which is equipped with an F board?

James R. Vespi
Dolgeville, NY

A. Your computer will do everything that the new 64K CoCo will without modification! The odds are that it even has a white case under its silver paint. Radio Shack has never sold a 32K CoCo with an F board—just a 64K machine that they painted silver.

Q. In the November issue, you told a reader to check his 64K upgrade with a memory-testing program. Can you give us one?

Jack Thompson
Gaithersburg, MD

A. Program Listing 3 should suit your needs. To check to see if your D or E board was upgraded correctly to 64K, first run the program with the E Board Test. If this checks out, you followed the instructions correctly up until step 1 at the bottom of p. 46 ("64K Modification"). If the first part is okay, run the ROM Area test. If this test does not check out or your machine hangs up with this test, retrace your steps beginning with step 1 at the bottom of p. 46. Keep in mind that Fig. 2 on that page should be numbered counterclockwise from the notch.

Q. I've written many programs for my CoCo where I needed continuous pixel movement. For example:

```
10 CLS0: M=247: H=31: V=16
20 IF PEEK(341)=M THEN V=V-1
30 IF PEEK(342)=M THEN V=V+1
...
...
60 SET(H,V,8): GOTO 20
```

This was done on an earlier model CoCo. However, now they tell me that the chip was changed and that the keyboard rollover table (341-345) holds the ASCII code, or something like that. Can you help?

In Elmer's Arcade, *HOT CoCo*, November 1983, p. 14, the author uses IF PEEK(341 AND 8)=0 ..., IF PEEK(342 AND 8)=0 ..., IF PEEK(343 AND 8)=0 ..., and IF PEEK(344 AND 8)=0 .... Will this be my alternative? What does it mean?

Barry Hornstein
E. Rockaway, NY

A. The keyboard rollover table includes the addresses 338-345. If no key is down, all eight addresses contain the value 255. In reference to the following table, if a key is pressed (top row in table), the address corresponding to that key is replaced by a different value (right column). For example, if the R key is pressed, address 340 will contain the value 251 at that moment.

Address	338	339	340	341	342	343	344	345	Values
	@	A	B	C	D	E	F	G	254
K	H	I	J	K	L	M	N	O	253
E	P	Q	R	S	T	U	V	W	251
Y	0	1	2	3	4	5	6	7	239
S	8	9	:	;	,	-	.	/	223
	ENT	CLR							191

```
10 FOR I=32383 TO 32407
20 READ X
30 POKE I,X
40 NEXT I
50 EXEC32382
60 DATA 26,80,142,128,0,166,132,
183
70 DATA 225,223,167,128,0,183,25
5,222,140
80 DATA 225,0,38,241,183,255,223
,28
90 DATA 175,57
100 CLS
110 PRINT"E-BOARD TEST - E"
120 PRINT"WRITE TO ROM AREA - R"
130 PRINT"WHICH ONE?";
140 A$=INKEY$:IF A$="" THEN 140
150 PRINTA$
160 IF A$="R" THEN 240
170 IF PEEK(49152)=68 THE S=&HE0
00 ELSE S=&HC000
180 FOR I=S TO &HFEFF
190 POKE I,0: IF PEEK(I)<>0 THEN
PRINT "TEST FAILS AT $";HEX$(I):
GOTO110
200 POKE I,255:IF PEEK(I)<>255 T
HEN PRINT"TEST FAILS AT $"HEX$(I
):GOTO110
210 NEXTI
220 PRINT"E BOARD TEST CHECKS OK
": PRINT
230 GOTO 110
240 POKE&HA147,99
250 IF PEEK(&HA147)<>99 THEN PRI
NT"TEST FAILS":GOTO 110
260 PRINT"OS-9, FLEX, 64K CHECKS
OUT": PRINT
270 GOTO110
```

Program Listing 3. Memory Test for 64K Upgrade

In your reference to Elmer's Arcade, none of those PEEKs when ANDed with eight will be zero if you have the newer 1.1 Color Basic ROM (EXEC 41175 to find out). The purpose of the AND 8 is to check to see if the 8 bit is set. If true, the result is eight; if false, the result is zero. This can be verified with pencil and paper by converting the number to binary and then checking to see that the fourth bit from the right is one.

Q. I have a new MC-10 Color Computer with the 16K expansion module. I also have Radio Shack's book *TRS-80 Color Computer Programs*, RS number 62-2313. I can run many of the programs, but cannot run those that have PEEK in them somewhere. For example, "Dragrace," *HOT CoCo*, November 1983, p. 68, with PEEK(65280) and "Orange Trap," *HOT CoCo*, November 1983, p. 126, with PEEKs 341, 342, 343, and 344. Any suggestions?

Pastor Mark S. Camp
Baldwin, MO

A. In "Dragrace," the author was checking to see if the fire buttons were pressed on the joysticks. Normally, address 65280 contains the value 255. If the right fire button is pressed, it is changed to 254 and if the left is pressed, it is changed to 253. If both are pressed at the same time, it is changed to 252. The other addresses in "Orange Trap" refer to the keyboard rollover table, which is explained in the preceding Q and A.

Q. I've seen ads for disks called flippies. What is the difference between them and the normal ones?

Mae Freudiger
Brooklyn, NY

A. A normal disk, whether single or double sided, has only one index hole and one write-protect notch. The index hole is the small hole next to the large one in the center of the disk. You can use a cheap hole punch to turn your disks into flippies. Make a template or use an old disk with the magnetic media removed. Use it with a felt-tip marker to indicate where a second set of index holes and a second write-protect notch should be on your new flippy. If you mark the positions correctly, the finished product will be completely symmetric with respect to the holes and notches. (See Fig. 1.)

Carefully slip a piece of paper or light card stock between the magnetic media and the housing to avoid scratching the media, then use the hole punch to punch a hole in the housing only for a new index hole. *Do not punch a hole in the magnetic media!* Repeat this procedure on the flip side of the disk. Next, punch a hole completely through the disk in the position that you marked for a new write-protect notch. Punch it so that the new notch is half-moon shaped.

If you do this to your disks, you can use both sides, thus halving the cost and space needed to store programs. Generally speaking, a single-sided disk is one that has been certified on only one side. Most likely, the second side has not been tested. This is not a guarantee, but I have personally never come across a disk with a bad second side. Let me know how it works out for you. ■

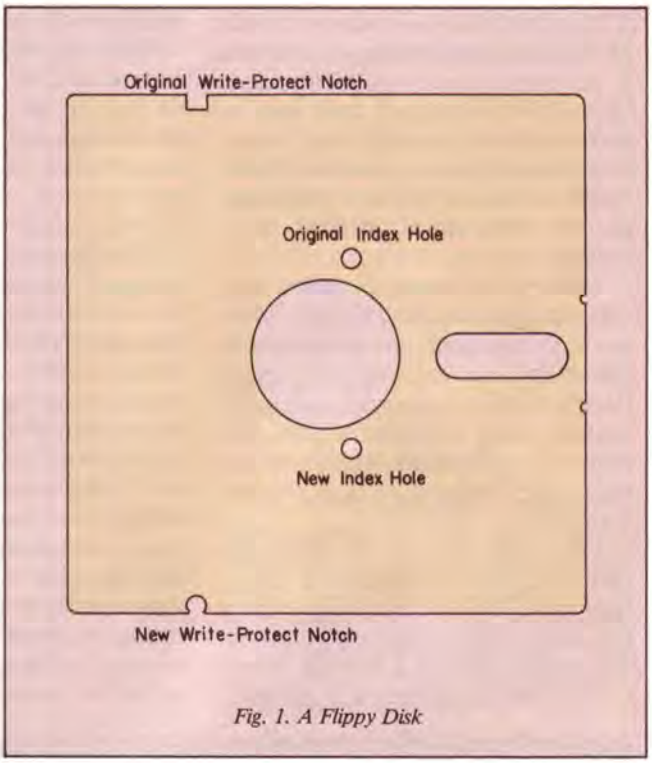


Fig. 1. A Flippy Disk

See List of Advertisers on page 130

Re:FLEX

PUBLIC DOMAIN MEANS FREE

by Scott Norman

Sometimes life holds some pleasant little surprises. I recently found some fellows writing terrific FLEX utilities practically in my own back yard. The routines work well and the price is right—as in free.

The gentlemen in question are Bruno Puglia, Leo Taylor, and Joe Mardo, and they have been at this for about three years. Their principal goal is promoting the cause of transportable FLEX software for all sorts of 6800 and 6809 systems; they aren't even Color Computer owners. The interests of the CoCo FLEX community are well represented in their work, however.

Two of their newest utilities are a supercharged CAT and COPY. These are fairly big (seven and 18 sectors, respectively), but each replaces several standard utility commands and incorporates novel features besides. The syntax is mainstream FLEX, so it's easy to get the hang of using the new commands.

Three disk files are furnished for each routine: an assembled binary command file, a commented source file, and a .DOC file that you can print out to give yourself a manual. (Some of their shorter utilities skip the latter and incorporate the documentation in the source-file header.)

These self-contained manuals are complete: CAT's runs to six pages, COPY's to nine. The TSC assembler is recommended for people who want to customize the source file for their own purposes.

I can only compare CAT and COPY (from now on, these names will always refer to the Taylor-Puglia commands) with the stock versions in my everyday FLEX—Frank Hogg Lab's version 5.0:3. The new material should work equally well with any other CoCo FLEX. CAT replaces not only the old CAT, but also the DIR and FILES commands.

CAT has eight options. You can change the default options (i.e., the ones in effect at power-up) by going into the source file, making the changes outlined in the manual (which amount to setting FCBs equal to zero or one as desired), and reassembling. Subsequent use of *any* option on the command line turns off all the defaults, so you have to enter every option you want explicitly to do anything nonstandard.

In case you forget which defaults are in effect for your system, the command CAT + brings up a help list with a "yes" for those that are enabled, or a "no" for those disabled.

CAT can use the standard syntax for quick-and-dirty jobs. For example, CAT 1 A .TXT (with the space) lists all the files on drive 1 whose names begin with A, *or* that have the .TXT extension, while CAT 1 A.TXT (no space) lists everything whose name begins with A and whose extension is .TXT—nothing unusual so far.

The fun begins when you invoke some of the options; a single letter designates each one:

- A—Alphabetize the catalog listing according to the first letter of the file name.
- D—Display each file's creation date.
- F—List each file's directory number.
- M—Generate a maximum (DIR-type) listing. This is equivalent to D, F, and S (see below) options, plus routines needed for interpreting the beginning and ending track and sector data and file protection codes.
- N—List nonexistent (deleted) files that might still reside on the disk. Of course, such files might not be intact. They might have been partially overwritten since deletion.
- P—Use the paging subroutine to generate a formatted listing (handy for disks with numerous small files).
- R—Repeat CAT when any key is pressed (except for E, which exits to FLEX). This is useful for cataloging many disks in succession, with the same set of options in effect.
- S—Display file sizes, in sectors.

The calling syntax is simple enough:

CAT + [Option letters]

Drive:1 Volume:SLN 3 Created:15-Nov-83

FILE#	NAME	TYPE	BEGIN	END	SIZE	DATE	PRT
1	COPY	.CMD	01-01	01-05	5	31-Jan-80	
2	CAT	.CMD	01-06	01-0C	7	3-Nov-83	
3	CAT116	.DOC	01-0D	05-01	57	3-Nov-83	
4	CAT116	.TXT	05-02	08-07	57	3-Nov-83	
5	CHECKD9	.CMD	08-08	08-09	2	14-Oct-83	
6	CHECKD9	.TXT	08-0A	09-10	24	14-Oct-83	
7	DATE	.CMD	09-11	0A-03	4	3-Nov-83	
8	DATE	.TXT	0A-04	0C-02	33	3-Nov-83	
9	DISKFIND	.CMD	0C-03	0C-08	6	3-Nov-83	
10	DISKFIND	.TXT	0C-09	0F-06	49	3-Nov-83	
13	DISKEDIT	.CMD	11-0C	12-02	8	3-Dec-82	
11	WORDSWAP	.CMD	0F-07	0F-0A	4	3-Nov-83	
12	WORDSWAP	.TXT	0F-0B	11-0B	35	3-Nov-83	

Files = 13 Biggy = 57 Total = 291/291 Free = 287

Fig. 1. An Example of an Alphabetized Maximum Listing.

The option letters can appear in any order, without spaces.

Here's an example of a full-bore catalog. The subject was one of my utility disks, containing the new CAT and some other Taylor-Puglia routines (the COPY is the standard utility, however). The command was CAT +AM, i.e., I called for an alphabetized maximum listing, as shown in Fig. 1.

The Begin and End data are in track/sector format. The files are organized according to the first letter of their names, while the file numbers are assigned according to location on the disk. Note the sequence in which the last three files are listed. File sizes are in grans, and there are no protection codes ("PRT" column) in effect for this particular disk.

The summary line at the bottom of the listing includes data on Biggy, the largest file on the disk. This would have been the same even if I had called for a selective directory in which the 57-sector files did not appear.

Total gives the total number of sectors in the current display, followed by the number occupied on the disk, and Free is just the number of empty sectors remaining.

CAT uses an adaptive display: The number of columns printed or displayed on the screen will vary, depending on the options selected. For example, when I tried a quick search for any deleted files with CAT +N, I received a listing four columns (four file names) wide, organized according to the file numbers.

CAT is both useful and easy to learn. COPY is even more of a workhorse, although it requires a little more effort to become familiar with its details. It is compatible with the syntax of the original COPY command, but, once again, it branches off with many new options (17).

The letters that enable the new option or options go before the drive numbers in the general syntax of the command:

COPY [Options], <Drive>,<Drive>
[,<Match List>]

You can specify any number of options in any order. Some of them are dangerous (in terms of possibly damaging a desired file), and these draw a request for a Y/N verification before the routine will proceed. The error trapping seems to work very effectively.

Here's a quick rundown of COPY's options:

- A—Copy in alphabetical order.
- C—Allow copying of corrupt (damaged) files.
- D—If the source and destination disks contain files with identical names, this guarantees that the destination will wind up with the newest version.
- E—Delete any existing destination disk file that matches a source file. Sometimes you may want to copy the older version of a file, and this does it.
- F—Copy by file number. This lets you use file numbers, rather than a match string list, as arguments of COPY. This is handy for building working copies of a system disk, where only FLEX.SYS and a few well-chosen utility commands are desired.
- K—Kill duplicate copy on the source disk. This is useful for getting rid of unwanted copies of files, but is potentially dangerous.
- L—List without copying. Disables the actual copy subroutine, allows you to preview what would have happened had it been allowed to run. For example, COPY KDL 1,0 previews which files would be deleted from drive 1 because a newer version exists on drive 0, but it doesn't actually delete them.
- M—Converts a FLEX serial file into a random file. You can use it with the R option below to recover a random file by track and sector.
- N—Copies files on the source disk that are not on the destination disk in any form. You can use it to add all new files to a back-up disk.
- O—Turns off all default options.
- P—Prompt before copying. Requests a yes/no answer before copying the next file. It's useful for scanning through a long source disk to build subsets of the files on a destination copy.
- R—Reads a file from a specified track and sector, without using the directory. You can use it to recover files if the directory has been destroyed, or to recover deleted files (assuming they have not been overwritten).
- S—Make a second copy of every file with .CPY extension.
- T—Overrides any protection on track 0 (where some systems store data files).
- U—Use current FLEX date, rather than source file date, for all copies.
- W—Wait for a keypress before copying. It removes the necessity to copy COPY.CMD onto the source disk. After COPY is loaded into RAM, the routine pauses and lets you remove the system disk and insert the source disk into the same drive.
- Z—Delete a file from the source disk after it has been copied. The result is that files are moved, not copied, from one disk to another.

There are many neat ways to use combinations of these options to automate complicated copying tasks. My favorite application is the production of dedicated, special-purpose working disks (like my customized DynaCalc package) that contain subsets of the utility command set.

COPY's self-contained documentation does a fine job of providing other examples—very professional stuff, indeed.

How can you get copies of CAT and COPY? The authors are primarily interested in promoting the cause of FLEX. Therefore, they have placed these routines in the public domain. Until the crush gets too great, you can get copies by sending a formatted disk, together with return postage and a large return envelope, to:

Bruno D. Puglia
27 Maitland Ave.
Randolph, MA 02368
617-961-3548 (weekends only)

To make things easier, members of user's groups should decide among themselves to send just one disk and make their own copies later. A few strategically placed disks can go a long way toward preserving the man's sanity, as well as giving him the time to write more good stuff.

And speaking of user's groups, one of Bruno's current enthusiasms is a national group for FLEX. I'd be happy to act as a clearinghouse for any thoughts you have on forming a FLEX group. Whether or not Re:FLEX would be the appropriate vehicle for any major exchanges of information remains to be seen.

Random Basic Comes to OS-9

My first piece of software that runs under OS-9 has arrived, and it isn't the expected Basic-09; rather, it's another version of Computerware's Random Basic (Box 668, 4403 Manchester Ave., Suite 102, Encinitas, CA 92024, 714-436-3512, $75).

As you might recall from the January column, I have a weak spot for the FLEX edition of Random Basic due to its large numerical range: 1.0 E – 99 to 9.999...E + 99. That comes in very handy for many problems in the physical sciences where very large and very small quantities are commonplace.

The OS-9 version, referred to on the disk label as Random Basic CC9, continues the tradition. In fact, it seems to be about as straight a translation of the earlier Random Basic as you might reasonably expect.

The ability to handle random, sequential, and ISAM (indexed sequential address mode) files is still there, as are user-defined functions, extended variable names to make programs more readable, and a host of other features.

There are a couple of welcome additions, though. For one thing, there are actually two versions of the language on the disk: Basic9 and Basic11. They boast nine-digit and 11-digit precision, respectively, using binary-coded decimal internal arithmetic. That accounts for the language's ability to handle such an extended range of numbers, although it also accounts for its lack of speed.

The new dialect also has commands that are specific to its operating system. For example, SHELL passes (forks) control to the OS-9 shell, letting you carry out any normal shell function while retaining Basic and any user program and variables in RAM. The clear/break combination returns you to Basic.

Although I have had little opportunity to exercise Basic CC9 thus far, I'll mention one new operator that ought to be a part of every Basic under the sun: SORT. You can guess what it does. The complete syntax is

SORT <Array #1> [,<Array #2>]

where the arrays (of either numerical or string type) are singly dimensioned.

The operator sorts Array #1 in ascending order. If you specify Array #2, its elements will be put into the same order as those of Array #1. This provides an easy method of setting up an index to preserve the original ordering of some array that is subjected to further manipulations.

The two versions of Basic reside in the CMDS directory of the Computerware disk. There are also a BAS directory containing a short demonstration program, and a couple of isolated files: an introductory document, a sample command file that shows you how to automatically load and start a program, and a new printer driver and descriptor.

As the last, but very recent, word, the newest upgrade of Random Basic (v1.6) has just arrived. There are 15 new commands for doing graphics, reading the joysticks, and sending information to any designated port. You can mix text and graphics on a single display if you use the high-res capability of Frank Hogg Labs' O-Pack.

While the graphics capability was there in OS-9 all along, it is now much easier to use—rather like Extended Color Basic. More on this in future columns. ■

Write Scott Norman c/o HOT CoCo, *Pine St., Peterborough, NH 03458.*

Coming Next Month

We want to help you make the wisest choices of peripherals, so our April Peripherals Issue contains a buyer's guide to help you with these important purchases.

The poet who said "April is the cruelest month" obviously didn't know what else April's *HOT CoCo* contains. The regulars are all with us: Dr. ASCII, Graphically Speaking, Product News, Elmer's Arcade (just for the fun of it), The Educated Guest, and reviews of The Color Accountant, Motion Picture Programming (animated graphics), and Simplex.

HOT CoCo features are a sure thing to help you use your new peripherals, or your not-so-new ones. Broken joysticks? Don't throw them out. Brian Alsop shows you how to fix them. Considering a modem? Michael Johnson is on-line about CoCo communications. Use your CoCo with the Epson FX-80 printer after reading the feature by Steve Eichman.

In April you'll also learn how to buy a disk drive, and how to build a transient-protection device. And for all you techies or would-be techies, we'll include an I/O quick-reference sheet.

Selecting and buying peripherals is a serious issue. So is April's *HOT CoCo*. But using them is fun, and April's *HOT CoCo* has plenty of that too. ■

Sugar Software

No Royalties?

$19.95 - Tape
16K ECB required

Software that you design and sell can be enhanced by using the Auto Run loader. Sugar Software will allow you to do this without requesting royalties.

AUTO RUN

- Generates a machine language loader to preceed your program on the tape.
- Starts up your Basic or ML programs automatically
- Locate your program anywhere in memory
- Displays a colorful title screen (which you create with the Auto Run graphics editor) while your pro gram loads
- Gives your program that "professional" touch
- Optional vocal, musical, or "sound effects"
- Does not take up any memory
- Used by many top Color Computer software houses

SUGAR SOFTWARE ✓144
2153 Leah Lane
Reynoldsburg, Ohio 43068
(614) 861-0565

MasterCard VISA

Add $1.00 per tape for postage and handling. Ohioans add 5.5% sales tax. COD orders are welcome. CIS orders EMAIL to 70405, 1374. Dealer inquiries invited.

A complete catalog of other sweet Sugar Software products is available.

GENERAL SYSTEMS CONSULTING ✓300

2312 Rolling Rock Drive
Conley, Georgia 30027

CASSETTE SOFTWARE
SINCLAIR ZX81
TIMEX SINCLAIR 1000
COMMODORE VIC20
TIMEX SINCLAIR 1500
TI 99/4A
TRS80 COLOR
COMMODORE 64

VISA MasterCard

DESIGNED TO HELP MONITOR YOUR FINANCES
16K MINIMUM FOR T/S 1000 & ZX81
EXTENDED OR NON EXTENDED FOR TI & TRS80

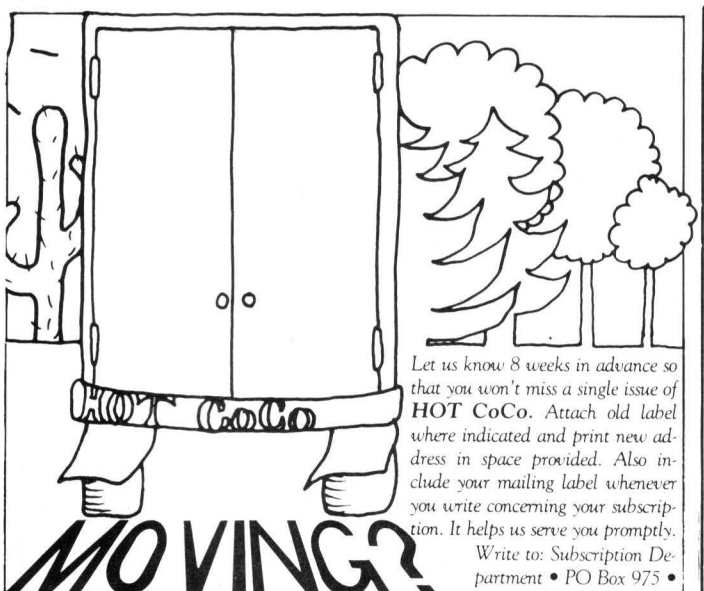
(404) 433-7143
(404) 243-7369

* At least 3K expansion ** At least 8K expansion	ZX81 T/S 1000 T/S 1500	VIC 20	TRS 80	TI 99/4A	COMM 64	YOUR PRICE
AMORTIZATIONS (LOANS)	14.95	15.95	16.95	17.95	18.95	
BAR CHARTS	15.95	**16.95	17.95	18.95	19.95	
ANNUITY EVALUATION	14.95	15.95	16.95	17.95	18.95	
FILE MANAGER	14.95	*15.95	16.95	17.95	18.95	
BANK STATEMENT BALANCER	14.95	15.95	16.95	17.95	18.95	
CHECKBOOK SIMULATOR	14.95	NA	NA	NA	NA	
DEPRECIATION STRAIGHT LINE	14.95	15.95	16.95	17.95	18.95	
DEPRECIATION DECLINE BALANCE	15.95	16.95	17.95	18.95	19.95	
DEPRECIATION ACRS	16.95	*17.95	18.95	19.95	20.95	
DIET PLAN	12.95	NA	NA	NA	NA	
HOME BUDGET	15.95	**16.95	17.95	18.95	19.95	
HOME INVENTORY	14.95	15.95	16.95	17.95	18.95	
HOME PAYABLES	14.95	NA	NA	NA	NA	
HOME EQUITY EVALUATION	14.95	15.95	16.95	17.95	18.95	
REAL ESTATE INVESTING	15.95	**16.95	17.95	18.95	19.95	
SAVINGS INVESTMENT ANALYSIS	15.95	**16.95	17.95	18.95	19.95	
IRS 1040 LONG FORM	29.95	**32.95	34.95	37.95	39.95	
IRS 1040A SHORT FORM & 1040EZ	24.95	**27.95	29.95	32.95	34.95	
INCOME TAX PROJECTIONS	16.95	**17.95	18.95	19.95	20.95	
IRA ANALYSIS	14.95	15.95	16.95	17.95	18.95	
DISK FILE CONCEPTS	NA	**24.95	NA	NA	29.95	

TOTAL YOUR PRICE	
1st CLASS POSTAGE HANDLING	3.00
GA. RESIDENTS 3% SALES TAX	
ADD 5.00 FOR DISK (COMMODORE ONLY)	
TOTAL PRICE	

NAME _____
ADDRESS _____
CITY STATE _____
CHARGE MY: ☐ VISA ☐ MC
SIGNATURE _____
HC-3-84

CARD# _____
EXPIRATION DATE _____



MOVING?

☐ *Extend my subscription one additional year for only $24.97.*
☐ *Payment Enclosed* ☐ *Bill Me*

Let us know 8 weeks in advance so that you won't miss a single issue of **HOT CoCo.** *Attach old label where indicated and print new address in space provided. Also include your mailing label whenever you write concerning your subscription. It helps us serve you promptly.*

Write to: Subscription Department • PO Box 975 • Farmingdale NY 11737.

Canada & Mexico $27.97/1 yr. only US funds drawn on US bank.
Foreign surface $44.97/1 yr. only US funds drawn on US bank.

affix old label

*name*_____
*address*_____
*city*_____ *state*_____ *zip*_____

new address

*name*_____
*address*_____
*city*_____ *state*_____ *zip*_____

HOT CoCo • PO Box 975 • Farmingdale NY 11737

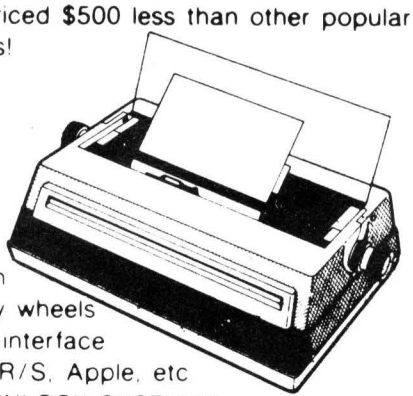
SCM

DAISY WHEEL
New Smith Corona TP-2

True letter quality printer for less than the cost of an office typewriter! Priced $500 less than other popular daisy wheel printers!

SALE PRICE:

$449



FEATURES:

- ★ Friction feed
- ★ 15 cps, 120 wpm
- ★ Changeable daisy wheels
- ★ Parallel or serial interface
- ★ Compatible with R/S, Apple, etc

SUNLOCK SYSTEMS
4217 Carolina Ave
Richmond, Va 23222 ✓456

ADDITIONAL PRINTER SPECIALS

Epson		Okidata		Gemini		C. Itoh	
RX80	$279	82A	$359	10	$279	8510AP	$369
80FT	$389	83A	$569	10S	$339	1150P	$629
FX80	$549	92	$429	15	$409	F10-40	$1099
FX100	$729	93	$699	15S	$469	F10-55	$1399

WE WILL MEET ANY ADVERTISED COST IN-STOCK

TO ORDER CALL TOLL FREE 800-368-9191
In Virginia call 804-321-9191
We accept MasterCard, Visa and CODs

✓See List of Advertisers on page 130

PRODUCT NEWS

edited by Cynthia Smith

Information used in the Product News section is supplied through manufacturers' press releases. *HOT CoCo* has not tested or reviewed these products and cannot guarantee any manufacturer's claim.

Math Addition For the CoCo

Here is another math program designed to make learning the concepts of addition and subtraction fun.

Math Joy: Addition and Subtraction has 3-inch bold numbers and high-resolution graphics with animation and sound. In the drills, it rewards correct answers and quickly corrects mistakes. The program allows a choice of one- or two-digit number addition and subtraction. Very young children will need adult help to use the program.

Math Joy requires 16K Extended Basic and is available on cassette for $11.95. Contact Microcom Software, P.O. Box 214, Fairport, NY 14450. 716-425-1824.

Reader Service ✓ 560

Photograph Your Display Screens

Now you can photographically record the images from CRT monitors for reference and presentations. Datacam I and Datacam 35 are graphics recording cameras designed for professional applications and use with personal and industrial computers.

Both cameras are portable, require no customizing for individual video screens, and use no complicated F-stop settings or exposures.

Datacam 35 comes in two monitor sizes (12″/13″ and 19″/21″), and records CRT images in a slide format. CRT graphics can be photographed on Polachrome 35mm film for speedy processing, or on 35mm color film.

Datacam I comes in the same two sizes and is useful for instant hardcopy photographic 3-by-4 prints for reference data, notebook presentations, or file information. It can be used with Kodak Instant Color film and comes with batteries and film. Prices of individual models vary from $500 to $1,000.

Photograph your display screens with Data Cam I and 35.

Contact Photographic Sciences Corporation, 770 Basket Road, P.O. Box 338, Webster, NY 14508. 800-828-6489 outside New York and 716-265-1600 (New York residents).

Reader Service ✓ 554

Letters and Numbers For Preschoolers

Software Specialists has introduced two new educational programs for children of preschool age. Both programs focus on an early learning approach that uses the computer as a fun tool to promote enthusiasm for learning.

In Early Letter Recognition, the child presses a key for one of the 26 letters of the alphabet for either a graphics rendition of the letter or an animated display using it.

Kid's Choice presents one of three displays at random, with each pressing of a particular number key, and has 30 displays in all. Routines associated with some of the number keys allow the child to participate in an activity based upon that number.

Kid's Choice requires 16K Extended Color Basic, and Early Letter Recognition is available for Color Basic and Extended Color Basic. Cost is $12.95 each, or $19.95 if you purchase both. Contact The Software Specialists, P.O. Box 2029, Princeton, NJ 08540. 609-443-6782.

Reader Service ✓ 559

New for Business

80 Custom Software has a new program designed especially for small businesses. This complete accounting system has up to 28 user-definable income and expense accounts, and up to 300 entries stored in memory at one time.

Records printed to the screen or optional printer include account totals, income and expense totals, individual entries, and a simple balance sheet. Individual entries can be listed by selected accounts or numerically in order of input. Business Manager supports monthly and yearly statements.

The program runs on a 32K Extended Basic tape or disk system and saves data to either tape or disk. Offered with complete instructions it sells for $24.95 tape and $29.95 disk from Reitz Computers & Electronics, 3170 W. Central Ave., Toledo, OH 43606. 1-800-242-COCO.

Reader Service ✓ 557

Extra Eyes And Tails For Your Dragon

Elkan Electronics has developed three new add-ons for use with the Dragon and Color Computer. The first, Dragon's Eye, is an on/off indicator that reminds you not to let your Dragon overheat. Installation time is 10 seconds, it requires no soldering, and does not invalidate your guarantee. Dragon's Eye is priced at £3.95.

Dragon's Tail is a joystick extender that makes it easier to plug your joystick into the Dragon, and saves wear and tear on your machine. It sells for £2.95.

The joystick Y-adapter called Dragon's Fork-tail enables you to use the joystick port for two purposes at once. Price is £3.95.

All three products are available from computer outlets or direct by order from Elkan Electronics, Freepost, 11 Bury New Road, Prestwich, Manchester, M25 6LZ, or from the new U.S. office at 1369 McCarter Highway, Newark, NJ 07104.

Reader Service ✓ 566

High-Scoring CoCo Utility

Teachers and test users will find Test-Aid useful. This program allows you to create a large bank of four-alternative, multiple-choice test questions; edit questions in the bank; select questions in the bank for inclusion on an exam; print a formatted copy of the exam; and generate different orders of items for an exam.

Test-Aid features menu-driven commands and formatted entry and display screens. The program handles questions up to 700 characters long, and allows you to store and load question banks from tape. Printed copy has automatic word-wrap at line endings and page breaks between separate test items.

Complete with instruction manual, the program comes on tape, requires 32K or 64K Extended Color Basic, and sells for $18 plus $1 shipping. Contact Infotools, 111 Country Club Lane, Oxford, OH 45056. 513-523-8473.

Reader Service ✓ 555

Plan For The Future

FICA-83 is a software program for the CoCo designed to calculate the approximate monthly pension check under the 1983 changes to the Social Security Act.

The computation of retirement benefits is a complex process and FICA-83 can eliminate much of this confusion. The program is also a valuable aid in retirement planning, and calculating Social Security pension under a variety of options.

You can use the program for a printout to the TV screen or to a hard copy with an 80-column line printer. Now available on cassette tape, FICA-83 comes complete with written instructions. It requires 16K Extended Color Basic,

and tape recorder. Price of the tape is $19.95 plus $2 shipping, from Parsons Software, 118 Woodshire Drive, Parkersburg, WV 26101.

Reader Service ✓ 553

Tax-Time Relief

This comprehensive tax preparation package saves time and anxiety for individuals and professional tax preparers. The program supports most tax forms and schedules now in use, is fully menu driven, and actually interviews you as you prepare the tax return.

Your data-entry errors can be immediately corrected by an edit mode similar to that of Color Computer Basic. The calculator mode supports all math functions on any line item that requires numeric data entry, and an audio warning sound alerts you to invalid math operations.

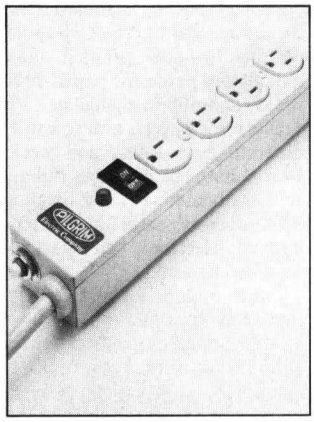
Printer outputs are made directly on government-approved forms and schedules using pin-feed or tractor-feed printers that support the CoCo serial-port interface.

The tax package comes with a comprehensive manual, and runs on 32K Extended Color Basic with one or two disk drives. The price is $149.95. Contact Micro Data Systems, 6 Edward Drive, Ashland, MA 01721.

Reader Service ✓ 551

Overload Protection

A new economical electrical outlet multiplier that has been introduced by Pilgrim Electric Company automatically shuts off the power when an overload occurs. The Model OS-4 has 4 NEMA 5-15R three-wire grounded receptacles. These are specifically intended for a CPU, printer, disk drive, and terminals. The product also features a built-in master on/off switch, red pilot light, and circuit breaker reset button. A heavy duty, 6-foot #14-3 SJT line cord is standard.



Pilgrim Electric's outlet multiplier prevents overload.

UL listed, the new Model OS-4 plugs into any 125 VAC, 15 or 20 Amp outlet and handles up to 1,875 watts. Priced at $28.95, they are now available from Pilgrim Electric Company, 29 Cain Drive, Plainview, NY 11803. 516-420-8989.

Reader Service ✓ 550

Free Communication Terminal Program

New for the Color Computer and the MC-10, Commterm allows both computers to access remote bulletin boards or timesharing services, or to act as terminals on-line to other computers.

Star-Kits Software Systems Corporation is offering Commterm free to anyone who sends a blank cassette and an SASE. Star-Kits gives users permission to copy the program or documentation for their own purposes. In return for making the program available, Star-Kits asks that Commterm users evaluate the program's worth and send what they consider a fair contribution to the company to encourage further development of such programs.

Contact Star-Kits at P.O. Box 209, Mt. Kisco, NY 10549. 914-241-0287.

Reader Service ✓ 562

CSPOOL

Micro Works has released CSPOOL, a print spooler for the CoCo and CoCo 2. CSPOOL lets you use your printer and computer concurrently, takes 26 bytes of Color Basic's memory, and gives 32K of print buffer.

It intercepts characters sent to the printer and stores them in the upper 32K of RAM. This allows a Color Basic program, that would normally wait for the printer, to return almost instantaneously with "OK." You can then run other programs.

CSPOOL is free with the purchase of a 64K RAM Upgrade Kit from The Micro Works, or you can purchase it separately on cassette or disk for $19.95. It requires 64K and is not for FLEX or OS-9.

✓See List of Advertisers on page 130

Contact The Micro Works, P.O. Box 1110, Del Mar, CA 92014. 619-942-2400.

Preview

Color Computer users can now use the Star-Kits Video Tape Sampler to preview programs before they buy them. The product is available in both VHS and Beta formats, and explains each of Star-Kits' software products for the CoCo and other 6809-based computers.

The Video Tape Sampler is priced at $20. After viewing it the user can erase and reuse the tape or return it to Star-Kits for full credit toward any program purchase.

Contact Star-Kits Software Systems Corporation, P.O. Box 209, Mt. Kisco, NY 10549. 914-241-0287.

Reader Service ✓ 561

Picture-Perfect Developer Program

PBK & Associates announces the release of BWDEV Black/White Developer, a 16K non-Extended program for the CoCo designed to assist photographers in film developing.

The program operates by asking a series of questions about the variables of film developing such as type of film, type of developer, normal or push processing, and developer temperature. It then calculates the variables and starts the timing process for each step of development with audible tones as reminders of agitation intervals.

BWDEV works with Tri-X, Plus-X, and Panatomic-X films and Microdol-X, D-76 and HC110 (dilution B) developers, including push processing of Tri-X to 1250EI.

It comes on tape and sells for $19.95 (add $2 for COD). Contact PBK & Associates, 5603 Linwood Court, Seabrook, MD 20706. 301-577-2930.

Reader Service ✓ 563

Low-Priced Printer

The DTC Style Writer, a daisy-wheel printer with hardware and software features, is being offered for $899.

Specifications and features include a standard 35K buffer memory for speedy throughput of approximately 20 pages. The use of the buffer allows the computer to fully load up the printer memory within seconds. Then you can use the computer for further applications while the DTC Style Writer finishes printing. With a multi-copy feature, repeat copies are possible without reloading the printer buffer memory from the computer. As an option, a 64K expanded buffer is available at $49.

Other features include: full bidirectional printing, automatic proportional spacing, standard Centronics parallel interface, graphics plotting, two-color printing, and a momentary pause for paper, print wheel, and ribbon changing.

The print wheel is available in 17 different type fonts, and works with a long-life cartridge ribbon. A user-oriented, self-test diagnostic routine evaluates the printer's internal electronic circuits and print mechanism.

Other options include a forms tractor for continuous paper feed, a bidirectional cut-sheet feeder, and 17 different type fonts to choose from. A variety of interconnecting cables enable the DTC Style Writer to work with most major personal computers in use today.

Contact Data Terminals and Communications, 590 Division St., Campbell, CA 95008. 408-378-1112.

Reader Service ✓ 552

Check-Mate Computing

Chess-D, a chess program for the Color Computer, uses methods of the current computer chess champion programs. It inspects 10,000 moves per second and at tournament level play (3 minutes per move) Chess-D can look ahead at least five plays. Players can set or change the lookahead level at any point during play from novice to expert level.

All chess moves are recognized or played as needed, including En Passant, Castling, and Promotion to any piece. All moves are entered through the keyboard using algebraic notation. A built-in opening book assures that the game starts correctly. A player can play as either black or white and can switch in the middle of the game.

The CoCo version uses high-res graphics and requires 64K RAM. The cassette sells for $39.95, and the disk for $49.95 with complete documentation from Computer Systems Distributors, P.O. Box 9769, Anaheim, CA 92802. 714-772-1390.

Reader Service ✓ 565

Access the World Of CP/M

CoCo Coupler 1 gives CP/M capability to your CoCo and is compatible with your disk controller and disk drives. Plug it between the computer and disk controller and it provides a platform for your disk controller cartridge. It needs no system modification.

To operate, menu select CP/M and the firmware loads the CP/M operating system. Select Color Basic, and your CoCo programs will run perfectly.

The CoCo Coupler 1 contains a 4 MHz 280A microprocessor and firmware for execution of CP/M programs. In addition to CP/M utilities, you receive the Wayne Technology disk copy and format utilities.

CoCo Coupler 1 requires 5.0 VDC @ 225 mA ± 10% supplied by the Color Computer. The disk format is compatible with Omikron/TRS-80 Model I, and display control emulates the Lear Siegler ADM-3A terminal.

It comes with a 90-day warranty, assembled and tested with CP/M 2.2 disk and manual. The price is $250. Contact Wayne Technology, P.O. Box 5196, Anaheim, CA 92804-1196. 714-772-5757.

Reader Service ✓ 566

DTC

DTC Style Writer

DATA TERMINALS & COMMUNICATIONS
590 Division Street
Campbell CA 95008
Phone (408) 378-1112 TLX2 (910) 590-2436

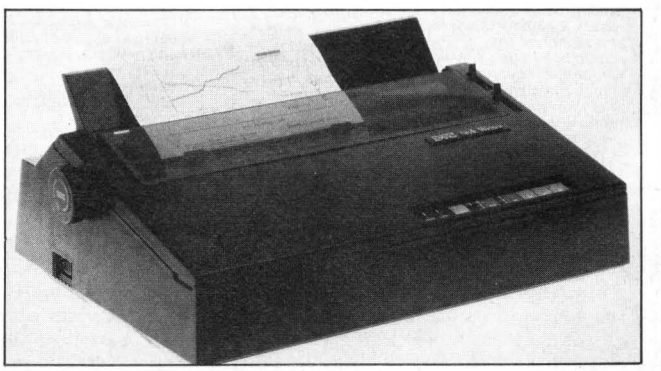
Style Writer: compatible with YOUR personal computer via the standard parallel (Centronics) interface, & comes with "the most complete word processing & graphics capabilities with letter quality printing and a wide variety of print styles" uses an advanced daisy wheel printer, a 35K buffer (stores up to 20 pages) and print speeds of over 130 words per minute. Many user convenience features are included such as self test, easily interchangeable print wheel cassette, readily available single & multistrike film as well as cloth ribbons for a variety of printing. Options include a forms tractor (adjustable) plus a great automatic single bin sheet feeder which provides increased operator efficiency. This feeder installs in seconds with no hardware/software modifications required. Plus it holds up to 150 sheets and can be reloaded during operation. All this and a lot more!

*** FEATURES ***

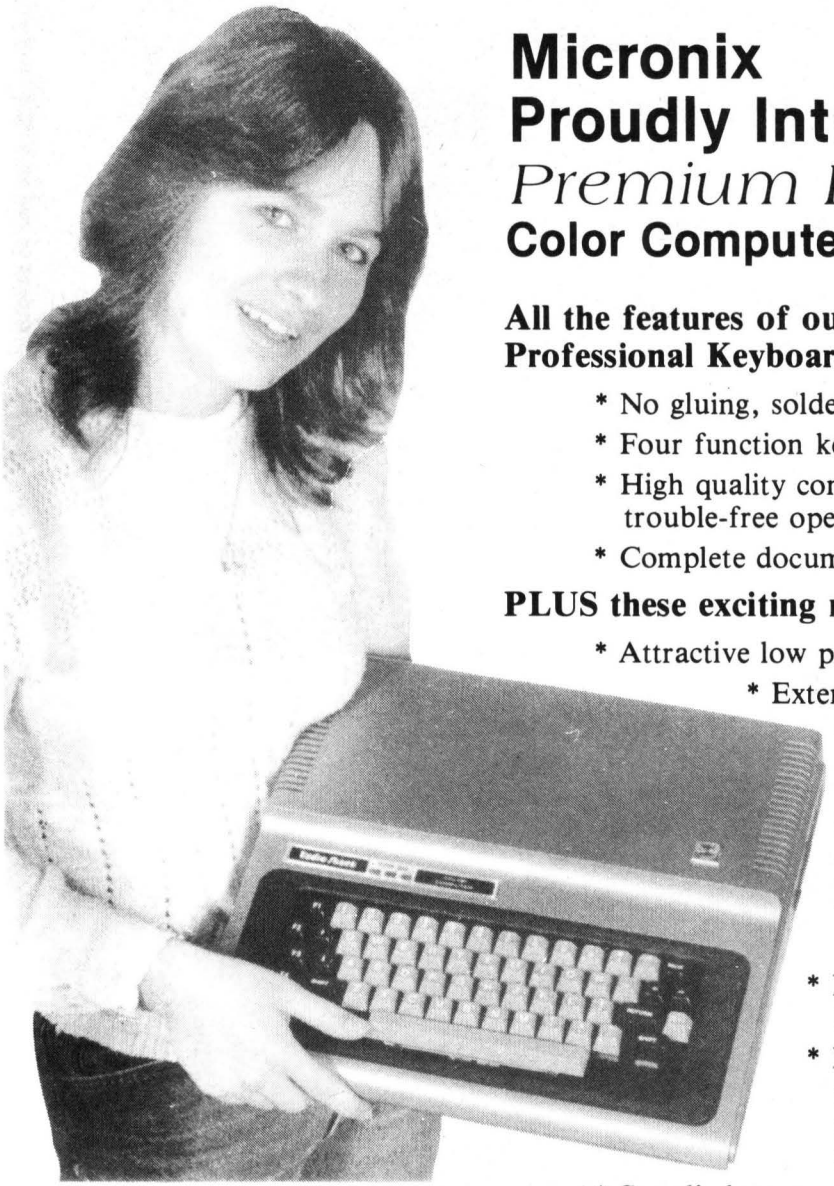
* Automatic Proportional Spacing
¢!"#$%'()*+,-./0123456789:;<=>?@`
ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_
+abcdefghijklmnopqrstuvwxyz{|}~`
* Variable pitch & Lines per inch
Variable pitch & Lines per inch
Variable Pitch & Lines per inch
Variable pitch & Lines per inch
* RED and BLACK printing
* SHADOW or DOUBLE print
* Automatic UNDERSCORE
* Sub_{SCRIPT} $SUPER^{script}$
* 35K Buffer (67K Optional)
* Bidirectional Printing
* Graphics

830930:V1.2

Print samples from the DTC daisy-wheel printer.



DTC Style Writer



Micronix Proudly Introduces Our New *Premium Keyboard* for your Color Computer

All the features of our popular Professional Keyboard:

* No gluing, soldering, or cutting—plugs right in
* Four function keys complete the matrix
* High quality construction assures years of trouble-free operation
* Complete documentation included

PLUS these exciting new features:

* Attractive low profile
* Extended Radio Shack layout
* Silk-smooth feel—uses ALPS keyswitches

Our Versakey software enhances the keyboard's utility

* Auto-repeat, n-key rollover and type-ahead
* F1 becomes DEFINE, F4 becomes CTRL
* May define up to 128 keys (including their SHIFT, CTRL, and SHIFT-CTRL combinations) as strings of up to 80 characters each.
* Supplied on cassette, may be copied to disk

"Have Josie ship yours today!"

The Premium Keyboard	$79.95
The Professional Keyboard . . . . (including software)	$59.95
Versakey software	$ 9.95

Both keyboards carry a 90-day limited warranty.

Please specify your computer's PC board type if known. Otherwise, specify the complete catalog number and serial number.

PLEASE NOTE ADDRESS AND PHONE NUMBER CHANGE

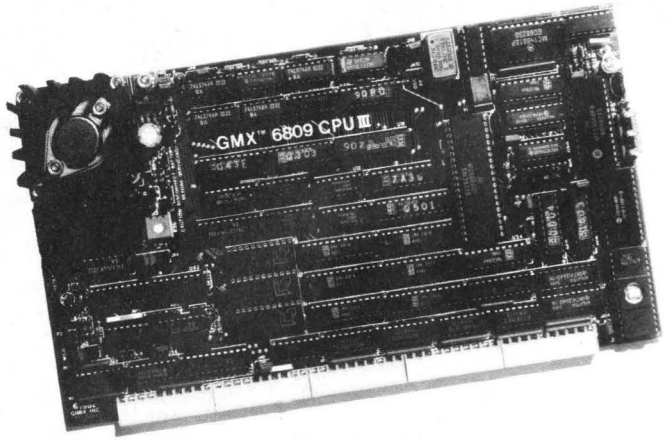
Micronix Systems Corporation ↙203

8147 Delmar Blvd.
St. Louis, MO 63130
(314) 721-7969

Terms: Prepaid check or money order, Mastercard or Visa.
Shipping Charges: U.S. $2.00, Canada $5.00, COD $3.50 (No COD's to Canada).

↙See List of Advertisers on page 130

GIMIX STATE OF THE ART 6809 SYSTEMS FOR THE SERIOUS USER.



GIMIX has 19MB or high performance 47MB Winchester Drive Systems and/or Floppy Disk Drive Systems.

For the user who appreciates the need for a bus structured system using STATIC RAM and powered by a ferro resonant constant voltage transformer.

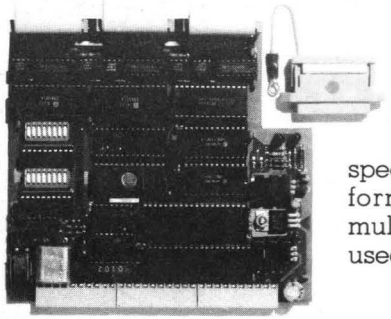
GIMIX has single user systems that can run both FLEX and OS-9 or Multi user systems for use with UniFLEX or OS-9.

GIMIX versions of OS9 and UniFLEX include maintenance and support by Microware (90 days) and TSC (1 year). Maintenance and support after this period are available at extra cost.

(NOTE: this support and maintenance is only for use with approved GIMIX hardware)

For the ultimate in performance, the Unique GMX 6809 CPUIII, using either OS-9-GMXIII or UniFLEX GMXIII (available shortly), gives protection to the system and other users from crashes caused by defective user programs. e.g. During program development, a programmer who crashes goes back to the shell or the debugger, while the other users are not even aware anything occurred.

The intelligent serial I/O processor boards significantly reduce system overhead by handling routine I/O functions, thereby freeing up the host CPU for running user programs. This speeds up system performance and allows multiple terminals to be used at 19.2K baud.



GIMIX 6809 systems support five predominant operating systems:

OS-9 GMX III,
OS-9 GMX II,
UniFLEX,
OS-9 GMX I,
FLEX

and a wide variety of languages and development software.

Whatever your application: software development, instrumentation, process control, educational, scientific or business; whether you need single or multi-user capabilities, GIMIX has hardware and the operating systems to get the job done reliably.

Please phone or write if you need further information.

GIMIX INC.
1337 WEST 37th PLACE • CHICAGO, ILLINOIS 60609 • (312) 927-5510 • TWX 910-221-4055

You've invested a lot of time and money into your computer . . . . .
It's time that investment paid off!

THE COLOR ACCOUNTANT

Introducing The Color Accountant (from The Programmer's Institute), the only complete personal financial package specifically designed for the TRS-80 Color Computer. This unique package includes:

1. **Complete Checkbook Maintenance**
2. **Chart of Accounts Maintenance**
3. **Income/Expense Accounts**
4. **Net Worth Statement**
5. **Payments/Appointments Calendar**
6. **Color Graph Design Package (graphs any files)**
7. **Check Search**
8. **Home Budget Analysis**
9. **Decision Maker**
10. **Mailing List**

After the initial setup, **THE COLOR ACCOUNTANT** requires less than an hour of data input each month.

The checkbook maintenance program is the key to the entire package. Once your checkbook is balanced, the checkbook summary file will automatically update the home budget analysis, net worth, and income/expense statements. You can then graph any file, record bills and appointments, make decisions, print a mailing list, analyze various accounts or stocks, and even calculate taxes.

All programs are menu-driven and allow add/change/delete. Each file and statement can be listed to screen or printer, and saved to cassette or diskette. **THE COLOR ACCOUNTANT** also comes with 40 pages of documentation that leads you step-by-step through the entire package. The TRS-80 COLOR Ext. Basic requires 16K for this package.

$74.95 Cassette **$79.95 Diskette**

Send $2 For Our New 64-Page Catalog (#11) Which Contains More Than 500 Software Products.
(Catalog is provided FREE with any order)

To Order:
Write or call Toll-Free (800) 431-2818
(N.Y.S. residents call 914-425-1535)

THE COMPUTRONICS INC.
50 NORTH PASCACK ROAD
SPRING VALLEY, N.Y. 10977

- add $3 for shipping in UPS areas
- add $4 for C.O.D. or non-UPS areas
- add $5 to CANADA or MEXICO
- add proper postage elsewhere

30-DAY MONEY BACK GUARANTEE

***** ALL PRICES AND SPECIFICATIONS SUBJECT TO CHANGE *****
DELIVERY SUBJECT TO AVAILABILITY

BUSINESS PAC 100

Now Available for the TRS-80 Color Computer!

Includes Inventory Control, Payroll, Accounts Receivable, Accounts Payable, Checkbook Maintenance, and more. Comes with 128-page user's manual.

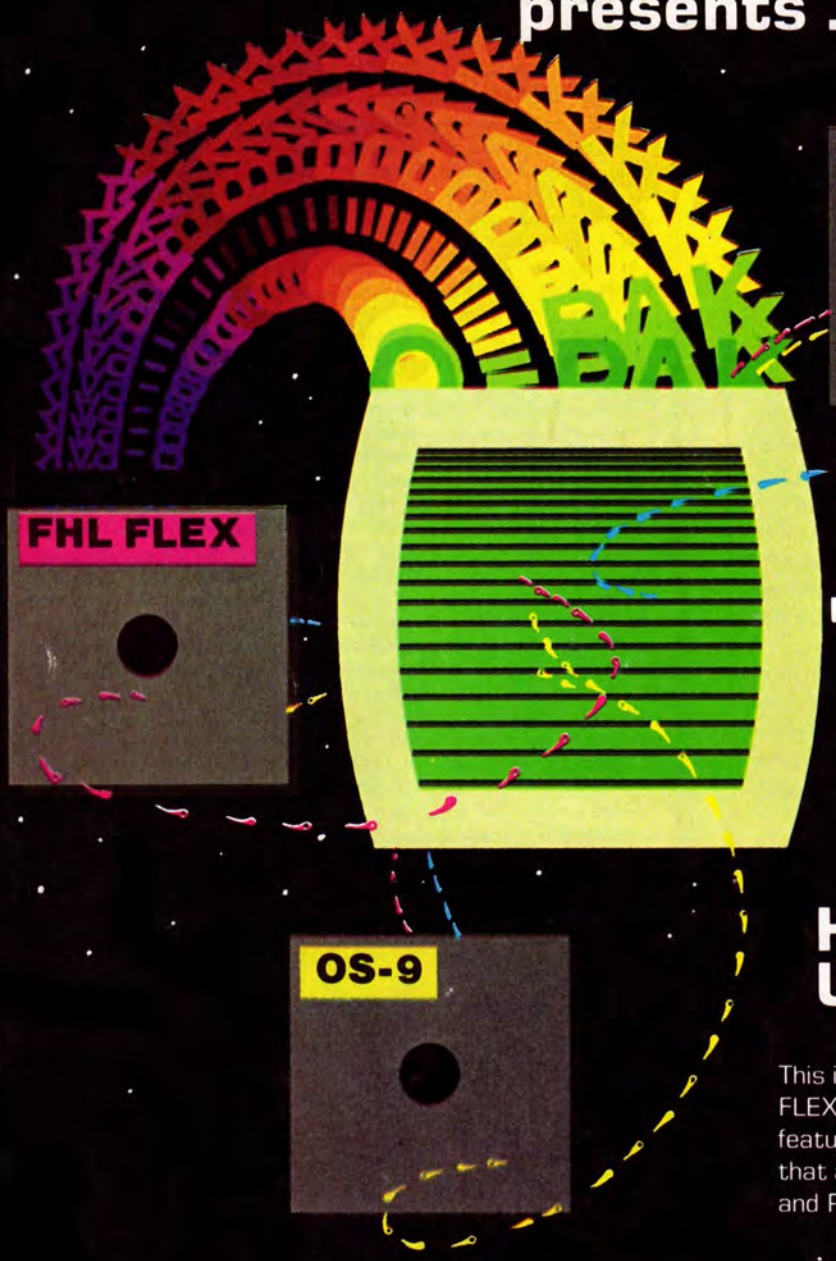
Interest apportionment by Rule of 78's • Annuity computation program • Time between dates • Day of year a particular date falls on • Interest rate on lease • Breakeven analysis • Straightline depreciation • Sum of the digits depreciation • Declining balance depreciation • Double declining balance depreciation • Cash flow vs. depreciation tables • Print checks with daily register • Checkbook maintenance program • Mortgage amortization table • Compute time needed for money to double, triple, etc. • Determine salvage value of an investment • Rate of return on investment with variable inflows • Rate of return on investment with constant inflows • Effective interest on a loan • Future value of an investment (compound interest) • Present value of a future amount • Amount of payment on a loan • Equal withdrawals from investment to leave zero over • Simple discount analysis • Equivalent & nonequivalent dated values for obligations • Present value of deferred annuities • Percent markup analysis for items • Sinking fund amortization program • Value of a bond • Depletion analysis • Black-Scholes options analysis • Expected return on stock via discounts dividends • Value of a warrant • Estimate of future earnings per share for company • Compute alpha and beta variables for stock • Portfolio selection model • Option writing computations • Value of a right • Expected value analysis • Bayesian decisions • Value of perfect information • Value of additional information • Derive utility function • Linear programming solution by simplex method • Transportation method for linear programming • Economic order quantity inventory model • Single server queueing model • Cost-volume-profit analysis • Conditional profit tables • Opportunity loss tables • Fixed quantity economic order quantity model • Cost-benefit waiting line analysis • Net cash-flow analysis for simple investment • Profitability index of a project • Weighted average cost of capital • True rate on loan with compensating balance required • True rate on discounted loan • Merger analysis computations • Financial ratios for a firm • Net present value of project • Laspeyres price index • Paasche price index • Construct seasonal quantity indices for company • Time series analysis linear trend • Time series analysis moving average trend • Future price estimation with inflation • Mailing list system • Letter writing system (links with mailing list system) • Sort lists of names • Shipping label maker • Name label maker • DOME business bookkeeping system • Compute week's total hours from timeclock info. • In-memory accounts payable system • Generates invoices on screen and print on printer • In-memory inventory control system • Computerized telephone directory • Time use analysis • Use of assignment algorithm for optimal job assignment • In-memory accounts receivable system • Compare 3 methods of repayment of loans • Compute gross pay required for given net • Compute selling price for given after-tax amount • Arbitrage computations • Sinking fund depreciation • Computerized UPS zone table • Type envelope with return address • Automobile expense analysis • Insurance policy file • In-memory payroll system • Dilution analysis • Loan amount a borrower can afford • Purchase price for rental property • Sale-leaseback analysis • Investor's rate of return on convertable bond

100 Ready-To-Run Business Programs

only $99.95

(Available on diskette only)
Also available for TRS-80 Models I, II and III, Apple, Atari, Osborne, Kaypro, all Commodore Computers and most CP/M systems.

The world's largest manufacturer of software presents . . .



O-PAK

O-PAK for OS-9

Hi Res Screen & Utilities Package

This is the same Hi-res screen that is used on FHL FLEX. Using the same control codes and the same features. The utilities include a three way copy utility that allows copying files between FLEX, OS-9 and Radio Shack DOS. For CoCo OS-9 - **$34.95**

FHL FRANK HOGG LABORATORY

THE REGENCY TOWER • SUITE 215 • 770 JAMES ST. • SYRACUSE, NY 13203
PHONE (315)474-7856 • TELEX 646740

262