

THE ORIGINAL
TRS-80TM COMPUTER

H&E COMPUTRONICSTM INC.

WORD PROCESSING
"SCRIPSIT"

NOW INCLUDES
BUSINESS
COMPUTING.TM



Cover Photo by Harry Peterson

H&E COMPUTRONICS INC.

50 N. PASCACK ROAD

SPRING VALLEY, NEW YORK 10977

*TRS-80TM IS A TRADEMARK OF TANDY CORPORATION

U.S. POSTAGE

STANDARD

BULK RATE

Permit #58

New City, N.Y. 10956

FORWARDING & RETURN POSTAGE GUARANTEED

PLEASE ANSWER THESE 5 QUESTIONS

1. Do you have information that must be kept organized and accessible?
2. Do you ever need to perform statistical analyses?
3. Would you like to have a tool that will allow you greater flexibility in managing your own or your company's money?
4. Could you use a mail list program that will be easy to use, maintain an unlimited number of names, allow you great sorting flexibility, and even interact with a word processor?
5. Do you own or have access to a TRS-80 microcomputer?

YES NO

<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	

If you answered "yes" to any of these questions, we can be of assistance to you. Our Maxi Series of applications programs are designed to give you maximum versatility with a minimum of hassle. These programs were created for business use, but you'll find yourself using them for personal applications as well.

Thorough support — Of course, each program comes with in-depth, user-oriented documentation, and is menu-driven to make it easy to use. When necessary, the Maxi programs are compatible with each other, and, whenever pertinent, are interactive with the major word processing and spreadsheet programs published by other manufacturers. Also, we maintain a telephone support line to provide you with any assistance you might require.

Maxi Manager by Dale Kubler

Maxi Manager is a remarkable data base manager. Its fast machine language sort complements its large data storage capacity. The sophistication of its data entry, management, and printing capabilities makes Maxi Manager a versatile tool for many applications.

The program now includes Maxi Utility, which allows you to rescue files on diskettes that have been damaged by excessive wear or misuse and lets you expand, add, or delete fields from an existing data base. Let the unmatched capabilities of Maxi Manager handle your data management!

Model I & Model III. Minimum 1 disk drive required . 012-0196 \$149.95

Maxi Mail by Dale Kubler

Maxi Mail is a powerful mail list management system that:

1. Is easy to use.
2. Has virtually unlimited storage capacity.
3. Interfaces with the major word processors to generate form letters and other text.
4. Prints mailing labels up to four-across in any format desired.
5. Has virtually unlimited coding capabilities with thirteen fields of information for each record.

Maxi Mail is the most sophisticated user-oriented mail list program available.

TRS-80 Model III only. 2 disk drives required 012-0148 \$99.95

Maxi Cras by Dale Kubler

(Check Register Accounting System)

Maxi Cras is a system that will computerize check writing, recording and analysis for business and personal finance. The system features 223 income and expense accounts, each of which will handle an unlimited number of transactions. Extensive register and report printing capabilities make Maxi CRAS an indispensable tool for managing money effectively.

Model I & Model III. Minimum 2 drives required 012-0145 \$99.95

Maxi Stat by David Walonick

Maxi Stat is the most useful statistical analysis package on the market today. It was developed to allow maximum flexibility in designing customized analysis. Maxi Stat handles the three main components of statistical analysis:

1. Complete menu-driven codebook creation and editing.
2. Menu-assisted data entry.
3. User-created control files to describe the statistical analyses to be performed and printed out on the variables of your choice.

Model I & Model III. Minimum 2 drives required 012-0153 \$199.95



THE
BUSINESS
DIVISION

Send \$1.00 for our 16 page booklet "Getting The Most From Your Micro" All 16 pages are packed with indepth explanations and printout samples from the Maxi Series of applications programs.

THE BUSINESS DIVISION

BOX 3435
LONGWOOD, FL 32750
(305) 830-8194



PUBLISHER
Howard Y. Gosman

BUSINESS MANAGER
Steven M. Kahan

EDITOR-IN-CHIEF
Hubert H. Howe, Jr., Ph.D.

BUSINESS EDITOR
Peter Shenkin, Ph.D.

MANAGING EDITOR
Martin Leffler

CONTRIBUTING EDITORS
Leo M. Conrad
Richard Kaplan
Spencer Koenig
Joseph Rosenman
Gordon Speer
A. A. Wicks
Steven M. Zimmerman, Ph.D.

ADVERTISING DIRECTOR
Kevin Rushalko

SALES MANAGERS
Valerie Furci
Rona Lowenfeld
Sheryl Prevot

DEALER SALES MANAGER
Janet Lasher

ART DIRECTOR
Edmund Khaleel

OFFICE MANAGER
Beatrice Kahn

SOFTWARE MANAGER
Darlene Bell

CUSTOMER SERVICE
Robert Williams

INVENTORY CONTROL
Michael Wiseltier

SHIPPING
Joan Gentry

PRODUCT DEVELOPMENT
Steven Kaplan
David Staub

PRODUCTION
Eileen Berman
Al Pizzo
Ruben B. Remigio
Louis Wetstein

MARKETING MANAGER
Andrew Hofer

PROGRAMMING MANAGERS
Roy Flynt
Jason Gapco

INVOICING MANAGER
Adele Damiano

SUBSCRIPTION FULFILLMENT
Janet Dillon
Karen Levine

AUGUST 1983

ISSUE NUMBER 60

CONTENTS

FEATURES

- 11 Hardware Review William D. Allen
Epson FX-80—First Look
- 14 Program Previews A. A. Wicks
Screen Artist II from The Smallsystem Center
- 25 Practical Business Programs S. M. Zimmerman and L. M. Conrad
Month #8: Miles and Move
- 27 Customer Response Time Calculator Dennis P. Avola
A model for the field service manager
- 30 It Started with the Abacus Frank Tymon
The age of computers is at hand
- 35 XSNORF Anthony T Scarpelli and Leigh T. Whittemore
The new, exciting word game for the pocket computer
- 37 The Computer Revolution: Who are the Beneficiaries? .. Name Withheld
Computers for the deaf/blind
- 39 Cobol Primer #3 H. C. Goodrich
Doing elementary arithmetic
- 42 Ask Richard Richard Kaplan
Final installment in this question/answer series
- 43 Metric Conversion Hubbard C. Goodrich
Convert all types of measurements with this program
- 45 Microfuture Sallie Stephenson
A micro fantasy
- 46 Everlastingly at it Mike Shadick
You're as Young as your Microprocessor!
- 48 Recipe Locator James Scalise
A system to help you locate recipes
- 54 Household Inventory William Byberg
Keep track of what you own
- 55 Decoding Passwords Spencer Koenig
Discover the passwords on your diskettes
- 57 CASSIO—An End to the Cassette Blues John Wilson
Save all your data at 1500 baud.
- 59 Add a Fancy Horizontal Scroll Dan Keen and Dave Dischert
Give your programs this exciting and dynamic feature
- 60 Perpetual Calendar Gail D. Landis
Compute any calendar for a 278-year period

REGULAR DEPARTMENTS

- 2 Bits and Pieces Howard Y. Gosman
Publisher's Remarks
- 4 The Crystal Ball
News and rumors of interest to TRS-80 owners
- 9 Letters to the Editor
Readers tell us what's on their minds
- 16 Beginner's Corner Spencer Koenig
Assemblers—More than One Way to Skin a CPU!
- 21 Pocket Computer Corner S. M. Zimmerman and L. M. Conrad
Finding the Effective After Tax Yield on Investments
- 62 Computronics Classified
- 68 Advertising Directory

Entire contents copyright © 1983 by H & E Computronics, Inc. All rights reserved. Printed in the United States of America.

All correspondence should be addressed to: The Editor, H & E Computronics, Inc., 50 North Pascack Road, Spring Valley, NY 10977. Unaccepted manuscripts will be returned if accompanied by sufficient first class postage. H & E Computronics will not be responsible for the return of unsolicited manuscripts, cassettes, floppy diskettes, program listings, etc. not submitted with a self-addressed, stamped envelope. Opinions expressed by the authors are not necessarily those of H & E Computronics, Inc.

Material appearing in the *H & E COMPUTRONICS MAGAZINE* may be reprinted without permission by school and college publications, personal computing club newsletters, and nonprofit publications. Only original material may be reprinted; that is, you may not reprint a reprint. Each reprint must carry the following notice on the first page in 7-point or larger type:

Copyright © 1983 by *H & E Computronics, Inc.*, 50 North Pascack Road, Spring Valley, NY 10977.

Please send us two copies of any publication that carries reprinted material.

ADVERTISING RATES

Contact Advertising Director for rate card. Special discounts available for multiple insertions.

Kevin Rushalko
(603) 547-2970

For information about receiving copies of *COPUTRONICS* in quantity contact:

U.S. and Canadian Distributor

H & E Computronics, Inc.
50 North Pascack Road
Spring Valley, New York 10977
Attention: Steven M. Kahan
Tel.: (914) 425-1535

International Distributor

Worldwide Media Service, Inc.
386 Park Avenue South
New York, New York 10016
Attention: Sandra A. Joseph
Cable: WORLDMEDIA
Telex: 620430 (WUI)
Tel.: (212) 686-1520

BITS AND PIECES

Howard Y. Gosman

OUR LAST ISSUE

This is the final issue of the *Computronics Monthly News Magazine* that you will receive from H & E Computronics, Inc.

The publishers of 80-US have acquired the *Computronics Magazine* and will fulfill all *Computronics* subscriptions starting with issue #61.

As many of our subscribers know, the *Computronics Magazine* is (as we say on the cover) the original magazine for TRS-80 owners. *Computronics* was the first publication devoted to TRS-80 owners (yes, even before *80-Micro*).

Originally H & E Computronics, Inc., started operations solely as the publisher of the *Computronics Magazine*. As the market for software and peripherals for the TRS-80 started to blossom, so did the number of publications devoted to the TRS-80 (do you remember *Softside*, the *MicroComputer Newsletter*, or *CIS*?). At that time, no one believed that the *Computronics Magazine* (or any other publication for the TRS-80) would survive. We did survive — CIS

didn't. *Softside* (with its three spin-off publications) didn't survive as a TRS-80 publication, although they successfully converted to a general-audience computer magazine. *Micro-Computer Newsletter*, along with about two dozen similar publications, all went underground, never to be heard from again. *Computronics* flourished (as all of the competition busted) . . . then came *80-Micro*. Unlike H & E Computronics, Inc. (a small, privately-owned corporation), Wayne Green, the publisher of *80-Micro*, was also the creator of several other magazines (such as *Byte*). Needless to say, *80-Micro* soon dominated the TRS-80 publication market.

Despite the competition from *80-Micro*, the *Computronics Magazine* continued to have an extremely loyal following and our subscriptions steadily climbed. Our goal has always been to offer TRS-80 owners an alternative to *80-Micro*.

Just after the start of *80-Micro*, another alternative magazine for

continued on page 6

The H & E COMPUTRONICS MONTHLY NEWS MAGAZINE is published by H & E Computronics, Inc., 50 North Pascack Road, Spring Valley, New York 10977. *The H & E COMPUTRONICS MONTHLY NEWS MAGAZINE* is not sponsored, nor in any way officially sanctioned by Radio Shack, a division of Tandy Corporation.

The purpose of the *H & E COMPUTRONICS MONTHLY NEWS MAGAZINE* is to provide and exchange information related to the care, use, and application of the TRS-80™ computer systems. H & E COMPUTRONICS, Inc. does not take any financial responsibility for errors in published materials. Users are advised to check and edit vital programs carefully.

The H & E COMPUTRONICS MONTHLY NEWS MAGAZINE encourages comments, questions, and suggestions. H & E COMPUTRONICS will pay contributors for articles and programs published in the magazine.

The H & E COMPUTRONICS MONTHLY NEWS MAGAZINE is typeset by Photonics, Ltd., 188 Highwood Ave., Tenafly, NJ 07670, and is printed by Kay Offset Printing Service, Inc., 154 Grand Street, New York, NY 10013.

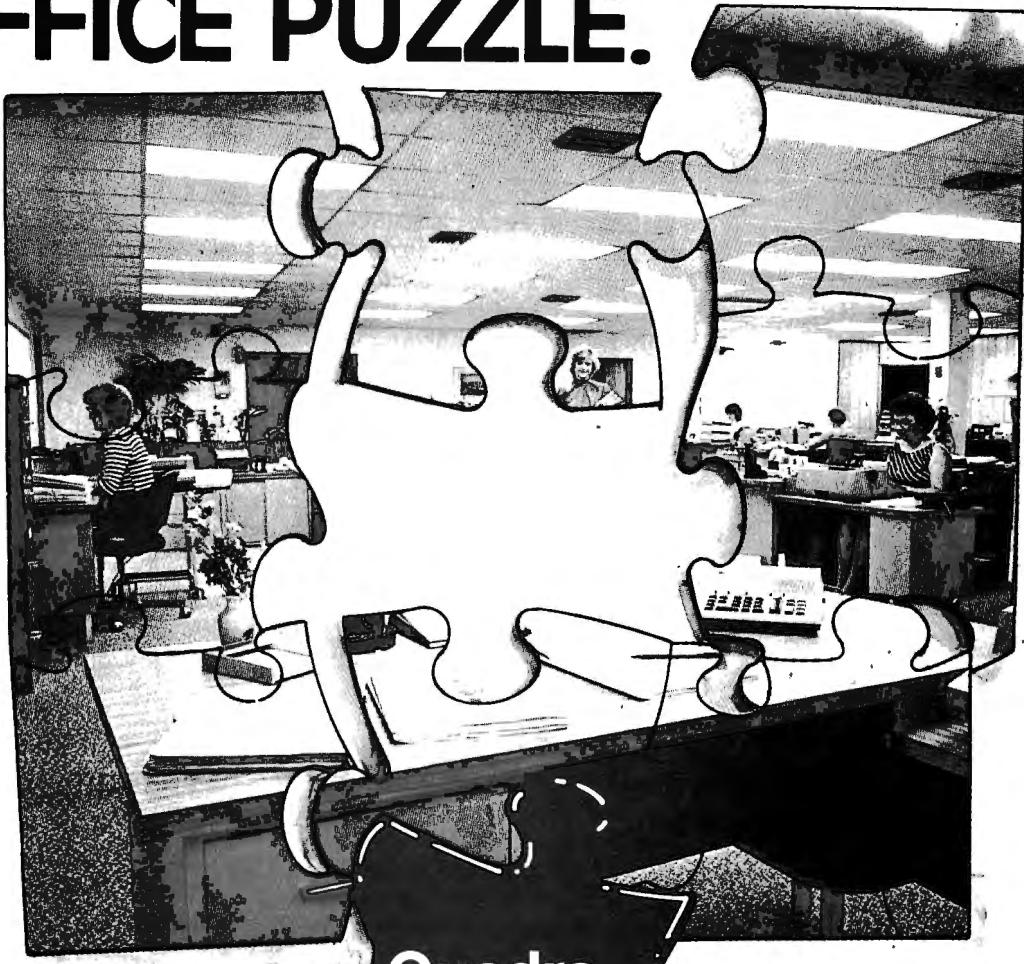
SUBSCRIPTION RATES

\$24 per year SURFACE MAIL	U.S. Only
\$36 per year FIRST CLASS MAIL	U.S.
\$36 per year AIR MAIL	Canada and Mexico
\$48 per year AIR MAIL	Outside U.S., Canada and Mexico
\$3 per copy Single Copies	U.S., Canada and Mexico
\$4 per copy Single Copies	Outside U.S., Canada and Mexico

Foreign subscriptions and sales should be remitted in U.S. funds drawn on a U.S. bank.

YOUR SUBSCRIPTION HAS EXPIRED IF . . . THE NUMBER ABOVE YOUR NAME AFTER THE DASH ON YOUR MAILING LABEL IS 60 (OR LESS). THE NUMBER FOLLOWING THE DASH TELLS YOU THE LAST ISSUE THAT YOU WILL RECEIVE. For example, if your subscription number is 16429-60, your subscription expires with this issue (#60).

MULTIPLEXING, THE MISSING PIECE TO A BUSY OFFICE PUZZLE.



A Quadra-MAC (Multiplexor) can expand the capabilities, capacity, and speed of your office services with the minimum expenditure. The Quadra-MAC allows a company to hook up to four TRS-80 Model I and/or III computers to one hard drive. This immediately gives all of those computers the benefits of having their own hard drive but saves the company thousands of dollars because one hard drive can be shared by up to four computers instead of having to buy a hard drive for each unit.

The Quadra-MAC also allows each computer to use as much processor time as needed without affecting the other users because only the hard disk drive is being shared not the microprocessor.

In addition, each computer can run totally independently of the other computers without being affected by either hardware or software failure of another computer.

Quadra-MAC

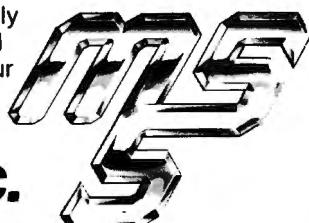
The Quadra-MAC also offers the following features:

- Up to 100 foot cables (allows up to 200 foot radius between two terminals)
- Includes a DOSPLUS Operating System
- Compatible with most hard drive systems
- Full File Locking with Multiplexing Features (prevents the file from being altered by more than one operator at the same time)
- No connector or buffer box cost

The Quadra-MAC is available for only \$995.00 which includes the complete DOSPLUS Operating Systems. The cable costs only \$2.50 per foot.

If MSS products are not currently carried by your local dealer, call Micro Systems Software for your closest dealer.

Dealer Inquiries Welcome.



MICRO-SYSTEMS SOFTWARE, INC.

4301-18 Oak Circle, Boca Raton, Florida 33431, Telephone: (305) 983-3390 Toll Free 1-800-327-8724

THE CRYSTAL BALL

News and Rumors of Interest to TRS-80 Owners

HIGHER DISK CAPACITIES

Minifloppy diskettes have come a long way since the first single density 5.25-inch drives appeared. Storage technology has steadily increased the amount of data that can be stored on a minidisk. Now a California manufacturer is packing more data onto minidisks than ever before.

The Amlyn company has unveiled a minifloppy diskette drive that will store more than two megabytes of formatted data on a standard 5.25-inch diskette. Today's standard minifloppy drives have a track density of 48 tracks per inch, while the new drives will write and read 170 tracks per inch. This has been made possible by a new innovation—inside the drive is a scale printed on mylar plastic. 1700 lines per inch have been photographically printed on the mylar, and the drive uses the scale as a reference to determine where the read/write head is positioned over the diskette surface. One problem with diskettes is that changes in humidity can distort the plastic of the disk, causing unreliable reading. However, the mylar reference scale reacts to humidity in exactly the same way, compensating for any change in the diskette. In addition, the drive uses a special reference track at the outside edge of the diskette, which allows the drive to measure any elliptical distortion of the other

tracks. Then the head moves along the same elliptical orbit as the tracks on the disk, using a special stepper motor that is capable of very small "microsteps."

Another relatively new technology in diskette storage is the vertical recording format. Kodak is now making a new type of floppy disk, called ISOMAX, which uses magnetic particles of a different shape than the needle-shaped particles used on standard diskettes. The ISOMAX particles are roughly egg-shaped and can be packed more closely together. The particles can also be magnetized equally well in any direction, which makes the new disks the first that can be used in either horizontal or vertical recording format (vertical format yields far higher capacities than the standard horizontal). The higher-density recording format plus packing of more tracks on each diskette may soon lead to 5.25-inch floppy diskettes with storage capacities of 10 megabytes or more! The only catch right now is that a standard disk drive cannot use ISOMAX diskettes. An ISOMAX drive will need smaller gap between the poles of its read/write head than the gap on a standard drive.

There are many new types of floppy drives appearing as manufacturers try to come up with the best possible design. One Japanese firm has created the smallest floppy diskette yet: only 2.5 inches in diameter. It can reportedly store 64K on a single, continuous spiraling track.

Hard disk drives are also making great strides in greater capacities on smaller disks. At the National Computer Conference, the Maxtor company of San Jose showed a 5.25-inch Winchester hard disk with a 380 megabyte capacity!

Shugart is apparently working on the first optical disc drive for microcomputers. A Shugart spokesman at the NCC show acknowledged that the company is working on such a drive, to be announced formally by the end of the year. This drive would be able to store one gigabyte (a billion bytes) of data on just one side of a 12-inch disc. Storage and retrieval of data would be much faster with an optical disc than a hard disk drive or floppy, because the reading and writing is

done with a laser beam. The system's one drawback is that the disc is not erasable—once written, data would stay on the disc permanently. This should present no problem though, since a billion bytes is more storage than most applications could ever conceivably need, and the discs themselves are expected to be quite cheap—perhaps less than \$15 each. The optical drive will not be a peripheral for the average home user, though—at least not at first. The Shugart drive is expected to cost about \$10,000.

ATARI'S 1200 XL DISCONTINUED?

There are strong rumors around that Atari will soon withdraw their new 1200 XL computer from the market, just a few months after its appearance. This was to be Atari's first serious computer, to be accepted even by businessmen as a professional computer. But Atari has had problems since the beginning with the 1200 XL, as many people felt that it did not deliver enough in power or sophistication to warrant the almost-\$1000 price tag. They couldn't hold up that price long, as retailers undercut each other's prices to the point that you could find the computer for under \$750 almost anywhere, and in some places for as little as \$500. Now ATARI is releasing four new XL models, the 600 XL, 800 XL, 1400 XL and 1450 XL with prices starting around \$199. None of the new machines will be as expensive as the \$749 retail price that Atari has attempted to keep on the 1200 XL. Many retailers and at least one software publisher feel that Atari is definitely going to discontinue the 1200 XL even though Atari says that the computer "will remain in its existing distribution channels" after the new XL models appear.

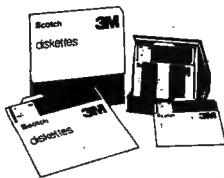
WIRELESS NETWORKING

A new method of hooking up a local area computer network was shown at the NCC show. Nelma Data Corporation of Ontario is developing a transceiver that allows computer-to-computer communication using a standard radio frequency signal. RF/PAC will work on any terminal or computer with an RS232

continued on page 6

Scotch® Diskettes

record reliability



at the lowest price!

Call our Modem Hotline (anytime) - 619-268-4488
for exclusive monthly specials. Our free catalog
contains more than 600 fantastic values.

ABC Data Products

(formerly ABM)

8868 CLAIREMONT MESA BLVD.
SAN DIEGO, CALIFORNIA 92123

ORDERS ONLY ITT TELEX INFORMATION
800-854-1555 4992217 619-268-3537

BeaLin Corporation

OFFERS YOU...

TRAX-SW

80/100 (Revised Version)

Now you can retain your *TRS-80 Block graphics capability while adding all of the features of **Graftrax-plus. Such as...

- Underline Mode
- Subject/Superscript
- Line Drawing Graphics

TRAX-SW	\$99.95
GRAFTRAX-PLUS (MX 80/100 Specify)...	\$69.95
GRAFTRAX 80	\$65.00
GRAFTRAX 80 (Modified for MX 100)	\$89.95

(Not available for serial MX-80 Printers.)

Plug in board supplied with or without **Graftrax-Plus.

LNW MODEL II 125 K/4MHZ OPERATION \$1995.00

Supports Model I systems and CPM. 5-8 Double density controller supports Hi-Res B&W and Color
TRS-80 Model I compatible; supplied with a NEC JB 1201 Hi-Res green phosphor monitor,
DOSPLUS 3.4 and LNW Basic (*Special prices for package purchases - call for quotation*)

LNW EXPANSION SYSTEM \$369.95

TEAC THIN LINE DISK DRIVES 40TK/SS/DD \$260.00

(one year warranty/price includes case and power supply)

PRINTER SWITCHES \$129.95

Manual/Software Selectable-Mod I, III...\$119.95 Mod II, XII, & XVI..

Switch between two printers, use your computer to produce draft copies on one printer.
Then make final copies on letter quality printer without switching cables.

(**MODEL II, XII, & XVI** requires **NO cable modification.**)

VERBATIM DATA LIFE DISKETTS (Box of 10) \$26.95

PAGE-IT \$29.95

A disk utility that prints basic programs in easily readable format.

- COMPATIBLE WITH MOST OPERATING SYSTEMS
- HIGHLIGHTS REM STATEMENTS
- INDENTS CODE — LINE NUMBERS STAND OUT
- UNPACKS BASIC CODE
- DOUBLE SPACE OPTION
- SUPPORTS SPECIAL OPTIONS OF 15 PRINTERS

HAYES SMARTMODEM \$229.00
1200 BAUD VERSION \$599.00

(Call or write for information and prices.)

USE OUR DATA LINE TO BROWSE THROUGH THE MANY ITEMS WE HAVE AVAILABLE. YOU MAY PLACE ORDERS OR HAVE SPECIFIC QUESTIONS ANSWERED. DATA LINE FORMAT IS 300 BAUD, 7 BIT, NO PARITY. SATISFY ALL OF YOUR COMPUTING NEEDS, COMPLETE SYSTEMS, DISK DRIVES, PRINTERS, EPSON RIBBONS & REPACKS, PAPER PRODUCTS, PLUS MUCH MORE. CALL OR WRITE FOR FREE CATALOG.

Software shipping and handling — \$3.50

PHONE VOICE (301) 490-2744 MODEM (301) 730-2229

*Trademark of Tandy Corp.

**Trademark of Epson

BeaLin Corp. 9335 Old Scaggsville Rd. Laurel, Md. 20707

VISA OR MASTERCARD ACCEPTED — DEALERS INQUIRIES WELCOME



WHY BUY A JOY STICK TWICE?!

IF YOU HAVE AN ATARI COMPATIBLE JOY STICK, YOU CAN USE IT ON YOUR TRS-80 WITH OUR KIT

JOY STICK KIT*

Without Joy Stick Model I/III \$15.95
With Joy-Stick Model I/III \$26.95

— Be Your Own SYSOP !!!! —

Bullet-80 Bulletin Board Model I/III
Version 8.0 \$150.00

Full line of H&E Computronics Software. 13% off any individual Versa Business Package
13% off All game software (from all manufacturers) if you mention this ad

Please call for information about ANY products. We have in stock a FULL LINE of Software from ALL Major Houses.

After-Market Computer Gallery**
P.O. Box 993 (Mail Order)
1 Franklin St. (Retail Outlet)
Danbury, CT 06810



Voice Line - 203 743-1299
Bullet-80 Computer Line - 203 744-4644
(300/1200 Baud)

PRICES DO NOT INCLUDE SHIPPING & HANDLING.

*Internal Installation Required. No trace cutting or electronics involved.

**A Division of Computer Services of Danbury

ARRANGER

100% Machine Language Disk Index Program for the TRS-80 Model I & III.

Automatically recognizes ALL major DOS's!

The Arranger is a master index system that automatically records the names of your programs, what disks those programs are on and type of DOS. Features include —

- Automatic single and double density recognition.
 - Accepts LDOS, DOS+, TRSDOS, DBLDOS, NEWDOS/80, MULTIDOS . . .
 - Works interchangeably with Model III, I double density.
 - Capacity of 250 disks, 44 filenames/disk
 - Quickly locates any amount of free granules
 - Finds a program in less than 30 seconds!
 - Alphabetizes 1500 filenames in 40 secs.!
 - Option to sort by any extension (/BAS, /CMD, ???)
 - Easily updates diskettes previously added with only 2 keystrokes.
 - Backup function built in.
 - Uses 1 to 4 drives, 35, 40 or 80 tracks.
 - Radio Shack double compatible
- Requires 32k / 1 disk minimum

JUST \$29.95

FREE SHIPPING

SATISFACTION GUARANTEED

Specify: TRS-80 Model number
(If you've added double density to your Model I, please indicate)

TRIPLE-D SOFTWARE

P.O. Box 642C
Layton, Utah 84041
(801) 546-2833

PERSONAL CHECK
VISA OR
MASTERCARD

THE CRYSTALL BALL

continued from page 4

interface and will allow communication within an office at distances as far as 200 feet. A small box with an antenna is placed on top of each computer, and the transmitted signal is encoded so as to be received only by the unit that it is addressed to. If it works, this system could eliminate the cabling headaches facing businesses that want to network their equipment. ■

BITS AND PIECES

continued from page 2

TRS-80 users was started... 80-US. 80-US is an excellent magazine devoted solely to TRS-80 owners, and has been in direct competition with the *Computronics Magazine* since its inception. Both 80-US and the *Computronics Magazine* have been offering TRS-80 users a multitude of information relating to the TRS-80 — information not published anywhere else.

So, by mutual agreement, 80-US has acquired the *Computronics Magazine*. We chose 80-US because it is an exceptionally well done publication and is truly dedicated to TRS-80 owners.

The decision to sell to 80-US was made because of several factors. The *Computronics Magazine* accounted for 100% of our income in 1979... but only 6% of our income during the first six months of 1983. The remaining 94% of our income comes from software sales. Recognizing the increasing competition among publications (80-US, 80-Micro, and others) H & E Computronics, Inc. started to aggressively sell software through our mail order catalog. Three years ago we translated our first software package to the APPLE computer. Now, our best selling software packages (including the VersaBusiness Series and the Business Pac 100) work on the TRS-80, Apple, Kaypro, Osborne, Xerox, IBM, Televideo, TI Professional Computer, Morrow Micro-Decision, Heath and just about all other microcomputers. 100% of our sales in 1979 were to TRS-80 owners. This year, only 25% of our customers were TRS-80 owners (about 35% of our customers are Apple owners). So, H & E Computronics, Inc., has aggressively expanded into other markets in addition to the TRS-80 market.

H & E Computronics, Inc. has become the largest microcomputer mail order sales company in the world. Our present catalog (#12) is on the way and is being mailed to a quarter of a million microcomputer owners. The next catalog, #13, will be mailed to a half million. By catalog #14, we expect to be mailing one million catalogs to microcomputer owners on a quarterly basis.

The printing costs related to the magazine have been horrendous, and the costs of acquiring new subscribers and advertisers made the *Computronics Magazine* a break-even operation. So, we made a good business decision. H & E Computronics, Inc., sold the *Computronics Magazine* to 80-US. The agreement is of mutual benefit: we will be able to free up our staff to devote additional time to expansion within the software market, and 80-US will gain a significant addition to its readership (which is especially pleasant for their advertisers).

What about our loyal subscribers? Our subscribers will start receiving 80-US as of September 1983. We believe that 80-US is an exceptional publication, very similar to *Computronics* in content. Any *Computronics Magazine* subscribers who also receive 80-US will have additional issues added to their 80-US subscription.

All *Computronics* subscribers will receive our latest catalog (#12) and will continue to receive future catalogs. The Mod-II/12/16 Newsletter will continue to be published by H & E Computronics, Inc.

MORE MAGAZINE CHANGES

80-US is now being distributed on newsstands, and the magazine will get a face lift and a new name. 80-US will become *Basic Computing*.

80-Micro and all of the other Wayne Green publications for computers have been sold to Computer World, Inc. (publishers of *InfoWorld*, *ComputerWorld*, *PC World* and other publications).

IMPORTANT NOTICE

FOR ALL PAST PURCHASERS OF OUR VERSABUSINESS SERIES, BUSINESS PAC 100, OR MASTER PAC 100: It is extremely important

continued on page 8



New Release

Now supports Mailing Lists, Form Letters,
"ZAP-PROCESSING", and 18 more printer drivers.

ZORLOF II

The Magnificent WORD PROCESSING SYSTEM

For the TRS-80 Model I and III

- Supports over 50 different popular printers including OKIDATA Microline 80, 82A, 83A, 84A, Qume, Centronics 737, 739, Radio Shack Line Printer IV, VI, Daisy Wheel II, EPSON MX-80, MX-100, Graftrax, Graftrax Plus, Gemini-10, Gemini-15, NEC PC-8023A-C, Spinwriter 5510, 5515, 5520, 5525, C. Itoh Prowriter 8510, Starwriter FP-1500, F-10, Tec 8500R, Smith-Corona TP-1, Brother HR-1, COMREX Com-Riter CR-1, IDS Microprism 480, and Diablo 630.
- Supports proportional space right-margin justifying on Centronics 737, 739, Radio Shack Line Printer IV, Daisy Wheel II, Graftrax Plus, NEC PC-8023A-C, Spinwriter 5510, 5515, 5520, 5525, C. Itoh Prowriter 8510, Starwriter FP-1500, F-10, and Diablo 630.
- Powerful Mailing List and Mail-Merge capabilities for personalizing standard legal documents and Form Letters, handling infinite number of data records per run, infinite number of data fields per data record, and data fields as large as up to 1000 characters each.
- Brand new feature called "ZAP-PROCESSING", allows you to display and edit any type of data or program file in "ZAP" (byte-hexadecimal) format.
- Any character or symbol your printer can print, even dot graphics, can be used in mid-line printing with the Special Character feature.
- Written in fast Z80 machine language with type-ahead key-stroke buffering for speed typing.
- Single key-stroke control of all editing functions for ease of use.
- Continuous on-screen display of word count, line count, and free memory count.
- Superscripts, subscripts, underlined, bolded, expanded and condensed type styles - combine and intermix within a line.
- Automatically justifies and word-wraps on the screen as you type.
- Search, Replace, and Global Search and Replace.
- Odd and even page user-definable headers, footers, and page number lines, with automatic page numbering.
- User-definable linespacing, sheet size, top, bottom, left, and right margins.
- Move blocks of text and copy blocks of text from disk, to disk, and within the text.
- Examine disk directory on any disk and kill files while editing.
- Powerful full-screen editing features for EDTASM and BASIC files, including automatic renumbering of lines.
- Built in function to dump contents of screen to printer.
- Print-previewing formats text, inserts headers, automatically numbers pages, etc. on the screen without printing it on paper.
- Page by page pausing capability for sheet fed printers.
- Supports both parallel and serial printers.
- Printer control code access.
- Works with NEWDOS, NEWDOS80, TRSDOS, MULTIDOS, LDOS, and DOSPLUS - Single or Double Density.
- Compatible with most all available spelling checker programs.

GUARANTEE

Many word processing systems claim theirs are the best, but few would dare guarantee them. Not us! We are confident that ZORLOF is the most useful word processing system on the market for under \$200. If you don't agree, return it within 30 days for a full refund.

Add \$2.00 shipping & handling. Florida residents add 5% sales tax. Checks require 3 weeks to clear banks.



SEE YOUR LOCAL
DEALER OR CALL

(305) 259-9397



ANITEK SOFTWARE PRODUCTS □ P.O. BOX 1136 □ MELBOURNE, FL. 32935 □ (305) 259-9397



The Banner Machine®

- For the TRS-80 I & III with 32K tape or 48K disk
- For use on the Epson MX-80 with Graftax
- Uses dot graphics instead of TRS-80 block graphics
- Menu-driven program
- Operation similar to a word processor
- Makes signs up to 10" tall by any length
- 10 sizes of letters from 3/4"-8" high
- Mono or proportional spacing
- Automatic centering; Right and left justifying
- Makes borders of variable width up to 3/4"

Order The Banner Machine®—\$49.95 from

Virginia
Micro
Systems

Virginia Micro Systems
13646 Jeff Davis Highway
Woodbridge, Virginia 22191

Phone (703) 491-6502



TRS-80 MODEL I T.M.* GOLDPLUG - 80

Eliminate disk re-boots and data loss due to poor contact problems at card edge connectors. The GOLD PLUG - 80 solders to the board card edge. Use your existing cables.

CPU/keyboard to expansion interface \$18.95
Expansion interface to disk, printer, RS232, screen printer (specify) \$9.95 ea
Full set, six connectors . . . \$54.95



EAP COMPANY
P.O. Box 14, Keller, TX 76248
(817) 498-4242
*TRS-80 is a trademark of Tandy Corp.

BITS AND PIECES

continued from page 6

that you register your software package. We have placed all registered owners in a database so that we may send owners of these software packages information related to the latest updates currently available for each software package.

NEW COLOR COMPUTER

Radio Shack has made many good marketing decisions this year with regard to new product entries. The TRS-80 Model 12, the Model 4, and the portable Model 100 are all real winners, each bound to be very successful in its market. The Model 100 is especially notable—it's a state-of-the-art portable computer that is presently the most advanced one available anywhere in the world at anywhere near the price. Now, however, it looks like they've made at least one marketing mistake this year. The new TRS-80 Micro Color Computer (called the MC-10) is here, but we have great doubts that it will be a good addition to the TRS-80 line. At \$120, this computer might have looked like good competition for other inexpensive computers a few months ago, but that was before Commodore and Texas Instruments dropped the prices on their VIC-20 and TI 99/4A to around \$100. If Radio Shack had just packaged the MC-10 differently, it might have had a better chance of success—this is a case where looks can make a big difference. The new TRS-80 is very small—a 2 pound plastic box that looks more like a large calculator or a child's toy than a computer. The "typewriter style" keyboard is not full size and you would not be able to type efficiently on it. For your \$120 you only get 4K of memory, which is really not enough even for a beginner—they should have made it 16K. The color graphics might be a lot of fun to experiment with, but without more memory, even the machine's graphic abilities will be limited. Storage is on cassette tape, and it doesn't look like there will ever be a disk drive for this one.

SUCCESSFUL HOOKUP

We finally got around to connecting our Model 100 portable computer to a desktop machine,

and it's just as easy as Radio Shack says.

We ran a null modem cable (an RS232 cable with several pins reversed) between the Model 100 and our Model 12. The opening for the RS232 port on the Model 100 is a little too small—you may have to either remove the plug housing at that end of the cable or use an extender to fit the plug into the RS232 slot (the Model 100's port has exactly the same problem).

Using Small Business Systems Group's ST80-III smart terminal program on the Model 12 and the Model 100's own TELCOM program, we were successfully transferring files back and forth less than fifteen minutes after setting up the machines and starting to read the TELCOM documentation.

Text files created with the Model 100's TEXT program can be used with the ELECTRIC PENCIL word processing program on the Model 12. The reverse also worked—ELECTRIC PENCIL files could be edited by the TEXT program on the Model 100. Next step, BASIC programs. A simple program was written on the Model 100, run, and then sent to the Model 12. The program also ran fine on the Model 12. Data files stored in RAM on the Model 100 are essentially text files, and they can be transferred just as easily.

What we have in mind, besides portable word processing, is to create a set of remote data entry programs for the Model 100 to be used in conjunction with our own in-house software which runs on a Model II. A BASIC program on the Model 100 will prompt the operator for a series of variables, say, for check register entries, and then sequentially write the data into a RAM file. This will allow the operator to do some data entry work at home rather than come into work on the weekend. On Monday morning, the file can be transferred into the Model II, where another BASIC program reads the file and processes the entries. This kind of programming is very simple to work out, and will make lap-size portables like the Model 100 very attractive.

PHONE BILL ANALYZER

Golden Braid Software has released LONG DISTANCE ANALYZER, a program that gives a

detailed picture of your company's phone expenses. The program organizes phone bills, identifying parties called, producing totals and reports, and analyzing geographic patterns. This program can be used to bill phone costs to your clients, cut abuse and waste by employees, evaluate special services (like WATS), print an alphabetical directory, and do cost accounting by your categories. You can accumulate monthly bills for long term analysis. The TRS-80 version requires 48K, TRSDOS and two disk drives (a printer is helpful, but not necessary). The program is priced at \$195. ■

LETTERS TO THE EDITOR

Were Facts Correct in "Born Again Computery?"

I have been a subscriber to *Computronics* for over two years, and I have assumed that your articles were factual. However, a recent experience does not support this assumption.

The January 1983 issue, page 55, carried an article by Michael Shadick called "Born Again Computery" about a pastor and a congregation using a TRS-80 computer as a religious teaching aid. I am interested in doing this with my Model III, so I wrote a letter to the pastor whose name was given and who was quoted several times in the article.

I am grateful to Rev. Roger Lundeen, pastor of the Hope Lutheran Church in Surrey, N. D., who was courteous to take the time and effort to reply. According to Rev. Lundeen, there is no Baptist church in Surrey, he does not have any kind of computer, nor does he know of any church in the area that uses a computer. Of course, the pastor's name given in the article was incorrect, and it is amazing that my letter was delivered. For once, I find myself thankful to the U. S. Postal System.

My question is: what is the source of the information presented in the article? Is it from the imagination of Mr. Shadick, or did he get the facts confused? Not only am I interested just to know the truth, but also I still would like to contact this person (or any other) who is using a computer in this manner.

Charles Battle
1616 Fairfield Drive
Plano, TX 75074

More Facts about "Born Again Computery"

I was surprised when I discovered that my name had been used extensively in an article that appeared in the January 1983 issue of *Computronics*. The article was written by Michael Shadick and appeared on page 55. The article appeared without my knowledge or consent. In fact, my first knowledge of the article came when someone sent me a clipping of it two weeks ago.

The article uses my name and quotes me extensively. Of all of the quotes, there is only one that I might have said: "I discovered, that it could help me in my ministry." The other quotes are outright fabrications.

Although I use my computer in my ministry, I have never had time to provide for its use in the area of Bible based games. The report of a Christian version of "Dungeons and Dragons" called "Keys to the Kingdom" is quite preposterous. My major use of the computer is in the areas of word processing and information management. I have used it for some programmed instruction experiments as well.

Mike Shadick attended a house church that I pastored in St. Paul, Minnesota before I was called to Timber Lake. He used his acquaintance with me, and my name, to lend credibility and legitimacy to his imaginative article about computer operations in imaginary Surrey ND. It is unfortunate that Mike used a factual name in his fictional account.

I have communicated with Mike, and discovered that his motivation was to provide me contacts with others who are also attempting to integrate the personal computer into the parish setting. His judgment in choosing this means of accomplishing his motivation was faulty.

It is my opinion that the way I am portrayed in this article could be damaging to my ministry here and damaging to my credibility in my attempts to provide input to my

★ FREE SHIPPING ★
Within Continental 48 States



MORE MAXI'S
MANAGER W/Utility (B.O.) . \$119.95
MAXI UTILITY \$44.95
MAXI CRAS Mod I/III \$84.95
MAXI MAIL Mod III \$84.95
MAXI STAT Mod I/III \$164.95

LAZYWRITER Mdl I/III..... \$149.95
NEWSRIPT 7.0 w/labels ... \$119.95
LDOS Ver. 5.1 Mod I or III \$109.95
DOSPLUS 3.5S/3.5D/3.5III.... \$119.95
MULTIDOS 1.6 SD/DD/III.... \$89.95
GEAP w/DotWriter 1.5..... \$84.95
SUPERUTILITY + Ver. 3.0... \$59.95
DATA-WRITER Mdl I/III.... \$124.95

LNW SPECIALS
LNW 80 Mdl II \$1,795.00
Expansion II 349.95
5/8 Doubler w/Dosplus 3.4D .. 199.95

RIBBONS

ZIP BOX RELOADS	1/2 Dz.	Dz.
Epson MX 70/80-20 Yds.....	24.00	42.00
Epson MX 100-30 Yds	30.00	52.00
NEC/Prowriter.....	21.00	36.00

Centronics 730/737/739/779 or
LP-I/II/IV-16Yds..... 18.00 32.00
All ZIP BOXES are individually sealed black nylon and require no rewinding. Epson Reloads also available in red, blue, brown, green & purple. Any mix allowed.

CARTRIDGES	Each	Dozen
Epson MX70/80	7.00	70.00
Epson MX100	12.00	125.00
Prowriter 8510 & NEC 8023A	7.50	80.00
RS LP I/IV	6.50	70.00
RS LP VI/VIII	6.50	65.00
RS DSY WH II or DW P 410	6.50	70.00
RS DSY WH II - Nylon	6.50	70.00
MICRLNE 80/82/A/83/A/92	N/A	30.00
MICRLNE 84 1/2x40 yds	5.50	60.00
Diablo Hytype II - Multi Strike	6.50	65.00
Qume -Multi Strike	5.00	50.00
NEC Spin - Hi Yld - Multi Strike	7.00	70.00
Centronic 703/04/53	11.00	120.00

Minimum order 3 cartridges - any mix. For smaller quantities add \$1.50 per order. All our reloads and cartridges are manufactured by one of the oldest and most reputable ribbon Mfg's. in the country.

* * * * * **QUALITY GUARANTEED** * * * * *

**SEE OUR EXPANDED ADS IN
80 MICROCOMPUTING
SEND FOR YOUR FREE CATALOG.**

ORDERING INFORMATION

No credit cards at these low prices. Add \$2.00 on all COD orders. Certified Ck/MO/COD shipped immediately. Please allow 2 weeks for personal checks. For extra fast service phone in your COD order. Free shipping within Continental 48 states via UPS ground. For Canada, Hawaii, Alaska, applicable shipping and insurance charges apply. Prices subject to change without notice. New York State residents please add appropriate sales tax.

The items listed above are a cross-section of our product line. We carry the full line of most companies listed in the ad, plus much more. **SEND FOR YOUR FREE CATALOG.**

**146-03 25th Road, Dept. C
Flushing, New York 11354**

Mon.-Thurs. (212) 445-7124 Fri. & Sat.
10 A.M.-9 P.M. 5 P.M.-5 P.M.

continued on page 10



FLEXIBLE DISKS

WHY PAY MORE?

Prices Per Box (of 10 Pieces)

	VERBATIM		MAXELL	MEMOREX	3M SCOTCH
	OPTIMA	DATALIFE			
5 1/4" SS, DD	#XL614-01 \$46.00	#MD-525-01 \$25.00	#MD-1M \$30.00	#3481 \$25.00	#744D-0- \$25.00
5 1/4" DS, DD	#XL624-01 \$59.00	#MD550-01 \$34.00	#MD2-DM \$41.00	#3491 \$34.00	#745-0 \$36.00
5 1/4" SS, QUAD	—	#MD577-01 \$33.00	#MD1-DDM \$40.00	#3504 \$37.00	#746-0 \$35.00
5 1/4" DS, QUAD	—	#MD557-01 \$42.00	#MD2-DDM \$49.00	#3501 \$47.00	#747-0 \$47.00
8" SS, DD	#XL34-6100 \$47.00	#FD34-8000 \$33.00	#FD1-1200 \$40.00	#3090 \$34.00	#741-0 \$33.00
8" DS, DD	#XL34-6201 \$58.00	#DD34-4001 \$40.00	#FD2-1200 \$45.00	#3102 \$40.00	#743-0 \$42.00

VERBATIM 8" or 5 1/4" HEAD CLEANING KIT \$10.00
 REPLACEMENT 8" or 5 1/4" CLEANING DISKS (10 pieces) \$16.00
 SCOTCH 8" or 5 1/4" HEAD CLEANING KIT \$32.00
 DISK MINDER 5 1/4" PLASTIC Holds 75 DISKS with 5 separators ... \$20.00

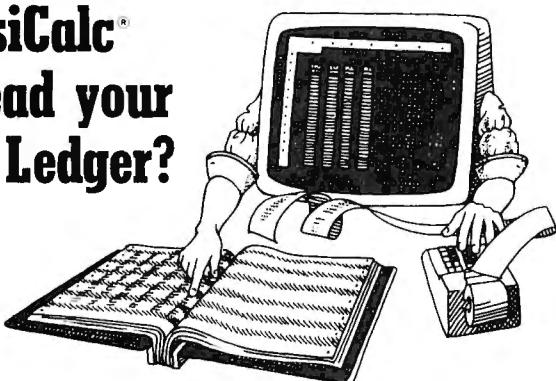
Send (2) 20¢ stamps for list of all available diskettes!!

S. D. C. & S. Co., Inc.

85-07½ Jamaica Avenue, Woodhaven, NY 11421
 (212) 849-8600

ORDERING INSTRUCTIONS: Money Orders, Bank Checks shipped same day.
 Personal checks must clear (15 days). Add \$3.00 shipping and handling per order.

**What if
 your VisiCalc®
 could read your
 General Ledger?**



It can with VIS\Bridge/GL™ from Solutions, Inc.

You can make even better corporate decisions by using the power of VisiCalc on data from your own **General Ledger**. □ Solutions' new Bridge product loads information from the Tandy General Ledger package directly into VisiCalc — Automatically! □ Make projections with VisiCalc without re-entering data already in your General Ledger. □ Project current financial data into the future. □ Compare your projections to actual results. □ Store monthly results to spot trends quickly.

Requires any version of VisiCalc, 2 disk drives and the Radio Shack General Ledger (26-4501). Available for TRS-80® Model II/12/16. VIS\Bridge/GL, \$195 + \$4 shipping & handling within U.S. To order phone (802) 229-0368 or write Solutions, Inc., 13 State St., Box 989, Montpelier, VT 05602. Mastercard and Visa. Dealer inquiries welcomed. Also available for Model I, III and II are: VIS\Bridge/REPORT PLUS™ \$95; VIS\Bridge/SORT™ \$89; VIS\Bridge/DJ™ \$295.

All VIS\Bridge products are trademarks of Solutions Inc. VisiCalc® is a trademark of VisiCorp. TRS-80® and Radio Shack are trademarks of Tandy Corp.

LETTERS TO THE EDITOR

continued from page 9

denomination as they too seek to apply computer technology to ministry. For these reasons, I have to insist that the article be retracted.

Mike is a creative and imaginative writer, and I understand that he contributes regularly to your publication. It is my hope that this incident will not prejudice you against his future submissions. I do hope that his future contributions in the area of "Born Again Computery" will be clearly marked as fiction and use fictitious names.

Rev. David A. Lundholm
 Box 408
 Timber Lake, SD 57656

Confessions of a Computronics Columnist

As I see it, my sin with regard to Pastor David Lundholm was not in the words I wrote, but rather in what I did with them.

To wit: before submitting the article to you, I should have first cleared it with Pastor Dave. Had I done so, he could have suggested changes and/or rewritten it completely, or rejected it.

I am deeply saddened that my failure to clear the article with him has resulted in his consternation.

Interestingly enough, this letter is being printed with the able assistance of a TRS-80 Model I, disk drives, and printer, all of which were once Pastor Dave's.

Mike Shadick
 Cedar Square West, Apt. E-414
 1515 South Fourth Street
 Minneapolis, MN 55454

Editor's Reply

We regret very much having been an unwitting party in the perpetration of Mr. Shadick's fictional account. For the reasons enumerated above, we hereby retract the article.

Non-Praise for the Abacus

Ref: Abacus, Computronics June 1983 page 5. Please, no. Don't. It would be interesting, but that's not

continued on page 13

HARDWARE REVIEW

William D. Allen

Epson FX-80 — First Look

Copyright® 1983 William D. Allen

The FX-80, Epson's new printer, has recently appeared on the market. Fortunately for me, the place where I used to work got one of the first FX-80's delivered. All the computers there were Apple II's, so I was not able to test the FX-80 with some of the software I would have liked to use, but I was able to check out a lot of the features. I was also able to get a few comments about the FX-80 from Jerry Goodwin of Pioneer Software. If you are interested in the FX-80 in particular, or if you are thinking of buying a new printer, read on.

SELECTING A PRINTER

First let me explain how I look at printers. I write for a living, so getting a new printer is a serious matter. Not only will it cost me money when I buy it, a mistake could cause me real business problems. Almost a year ago, after doing a lot of research, I decided that I would get an Epson MX-80 F/T. More accurately, I had decided to get the MX-80 F/T just as soon as I could justify the cost. Before I could do that, however, the new Epson FX-80 appeared, and I was lucky enough to get to use one of the first ones delivered. Like most people would, I started to think I should buy the new FX-80 instead of the older MX-80.

When I finally did buy a new printer, however, I did not buy either the proven, older MX-80 F/T or the flashy, new FX-80. What follows is a quick look at the new FX-80 along with the story of my buying a printer, ironically one that I had already decided not to get.

MX-80

I had two main reasons for making my original choice, the MX-80 F/T. Those reasons were the reliability of the unit and the fact that it was well supported by good third-party software such as MAXPRINT. Because there are several other printers that seem to be as reliable as the Epson, the deciding factor was the software support. In future articles, I plan to review some of that software.

Since some of you may not be familiar with the Epson printers, let me first say a little more about the Epson MX-80. It is one of the great success stories in the microcomputer world. Not only has it sold well, it has worked well. And Epson kept improving it: Graftax, Graftax Plus, and a friction feed model named the F/T (Friction/Tractor). Although I never purchased one, fortunately I was able to use several, both the MX-80 and the MX-80 F/T.

The plain MX-80 has only a tractor feed. I originally decided to get the F/T (Friction/Tractor) option because I wanted the friction feed for business reasons. With it I could use the printer with letterhead, invoices, etc. Since there are some relatively inexpensive friction-feed adapters for the MX-80 on the market, I thought of buying the cheaper MX-80 rather than the more expensive F/T model. Then I could add friction feed to the printer later by buying one of the kits.

Before I did that, however, I talked to someone who has used most of the friction feed kits for the MX-80, Ed Umlor of E. & G. Umlor. You may have seen Ed's name because he does some hardware reviews for Byte. His normal line of work, however, is selling and servicing TRS-80 hardware. Some of his business customers have bought different models of these kits, and Ed has yet to see one that he feels works well.

The only real drawback to the Epson was the price of the ribbons. I decided I could take care of that by reloading my own. It is messy but cheap. More thorough research showed it to be unnecessary. One of the local Apple dealers, Agriplex Computers, sells Epson ribbons so cheaply that it is not worth reloading them.

DAISY WHEELS

Since I was choosing a printer for a business that consists mostly of word processing, you may wonder why I did not want to get a daisy wheel printer. Contrary to what a lot of "experts" have said, very few people who do word processing

should buy a daisy wheel printer. They are slower and less reliable than dot matrix printers. The only real reason for buying a daisy wheel printer is if you are in a business where image is important, such as law or advertising.

FX-80

As I mentioned, before I bought an MX-80, the FX-80 came onto the market. According to the local dealer, only the MX-80 is being discontinued; the MX-80 F/T will continue to be offered. (Later I asked a spokesman for Epson America about that. He would neither confirm or deny it.)

The new FX-80 has many nice features that are real improvements over the MX-80. The first thing I noticed was the manual. The MX-80 Graftax Plus manual was written by Dr. David Lien, who is well known for writing the first TRS-80 manual as well as other books. I am a great fan of Dr. Lien, but I preferred the straightforward approach of the manual that came with the FX. It is in sharp contrast to the tutorial approach of the MX Graftax Plus manual that comes with the current MX-80's.

For first time users, however, the manual I liked so much would not be very good. Epson seems to have realized that too. By the time you read this, the FX-80 will be delivered with a different manual. This new one is more in the style of Dr. Lien. It is, however, by someone else, Mr. David A. Kater.

The new manual arrived from Epson only hours before this article had to go into the mail, so I was not able to review it properly. There might be some serious mistakes in the manual that I did not see, but I am favorably impressed with it. In fact, it appears to be the best printer manual I have seen. For one thing, the author seems to have anticipated many common mistakes and warns you against making them. A good example is the information on how to construct and use the alternate character set. It seems clear, but unfortunately I was not able to actually

try the instructions. The only thing I saw that was not well done was the illustration showing why you would want to use unidirectional printing. The purpose is to get columns to line up evenly, but the example did not show that.

If you bought one of the first FX-80's and have one of the old manuals, you can get a copy of the new one through your local dealer. According to Epson, all you need is proof that you purchased an FX-80 with the old manual. It is easy to tell the difference. The original manual had a predominately white cover; the new manual's cover is grey.

The next thing I noticed about the FX-80 was the convenience of being able to use the internal switches. (These are called DIP switches because they are small switches that look like the most common variety of integrated circuit, often called a DIP, which stands for "Dual Inline Package.") Inside the MX-80 there are also two DIP switches that control a lot of the printer's features. The one I use most often controls whether or not the zero is printed with a slash. Unfortunately, the switches can only be used after putting the printer totally out of action. To get to the switches inside the MX, you must remove the paper, unplug the printer, turn the printer over, remove four screws, turn the printer back over, and then take the case apart. To use the MX again, you must repeat the procedure in reverse. The FX, on the other hand, has these switches conveniently located beneath an easily removable panel where they are convenient to use.

The third thing I noticed, however, was something I did not like. Even with a new ribbon, I was unable to get the print quality of the MX. The characters of the FX are nicely formed, but they do not print out with the solid black look of the MX. The FX-80 has a command to slow the print speed down. I tried the FX in that mode, but the print quality did not change. The noise is also supposed to be lessened at the slower speed, but I did not notice any difference in the level of the sound. Only the pitch of the noise seemed to change. I have talked to several other people about these problems including someone from Epson, and they had all noticed the same things.

There is one other drawback to the FX, its nonadjustable tractor. The

tractor on the MX can be adjusted for width so that various kinds of tractor feed labels, checks, et cetera, can be used. However, there is an adaptor for the FX that gives it a variable width tractor feed.

The rest of the differences between the MX and the FX are improvements. The MX-80 has character sets and graphic capabilities in a ROM. There have been several versions of the MX-80 ROM. The latest is Graftrax Plus. The FX-80 has that kind of stuff in a ROM too, but it has even more than Graftrax Plus. According to Epson, the new FX ROM is upward compatible with software written for Graftrax Plus ROM. The FX also has 2K of RAM that can be used as either an input buffer or to store an additional character set.

Also, the MX prints at 80 characters per second (cps); the FX prints twice as fast. There is a catch to this, of course. The price has gone up. The increase over the MX-80 F/T is about twenty percent, not that much if you can make good use of the improvements.

Mechanically, the FX is also very nice. It is easy to thread the paper into the tractor feed. I find it much easier than the MX. With the FX, the paper can be torn off within about an inch of the print head. Unless you must print on the very top of the sheet of paper, you can make good use of this feature. It seems like ten percent of the paper I put through the MX is wasted just getting a printout. I am always pushing the FF (Form Feed) button to get a printout up to where I can tear it off.

That brings up another feature of the FX. When you press the FF switch on the MX, you have to wait until the form feed is complete before you can put the printer back on line. With the FX, the two switches can be pressed one right after the other. The FX has a built in latch.

The FX-80 is supposed to be able to make use of software written to use the features of Graftrax Plus, but I was not able to test this. Consequently I called Jerry Goodwin, better known as Pioneer Software. Jerry said that he cannot speak for others, but all his MX-80 software works with the FX-80.

Jerry Goodwin is one of the most knowledgeable people I have interviewed while writing articles about microcomputers. Consequently, I also asked him what he thought about

the FX-80. Jerry said he thought the three best improvements over the MX-80 were the ability to download a character set, the reverse linefeed, and the proportional printing mode.

While I was trying to decide which printer to get, something unexpected happened. Although I have been writing things ever since I worked as a newsman almost twenty years ago, I have never really sold that much. Suddenly several magazine editors started asking me for articles and reviews. My printer was broken, and the printer I had borrowed for my writing was about to leave town with its owner. My own printer could be repaired, but there was no guarantee it would work reliably. The time had finally come for me to buy a new printer.

The FX-80 looked very attractive because of its ability to hold a custom character set in its 2K of RAM. I do enough scientific writing that it would be really great to have things like Greek letters and integral signs easily available. On the other hand, there is no support software out for the FX yet. It is too new. I was not sure how easy it would be to use the features of the FX if I had to do the software myself. (I attribute my lack of ulcers to the fact that I program only when unable to find any other solution to a problem.)

The MX-80 F/T was about \$100 cheaper than the FX-80. It was well supported by software like MAX-PRINT that friends of mine were using and liked well. It would also handle single sheets of business letterhead, something I felt I really needed since I had my own business.

In the end, however, I bought neither the MX-80 F/T nor the FX-80. Instead I bought a plain MX-80 without the F/T option. As sometimes happens, luck intervened. During a previous interview with Jerry Goodwin, already mentioned as the head of Pioneer Software, I told him what I thought I needed in a printer. Jerry knows a lot more about printers than I do. He was nice enough to point out, ever so politely, that my reasoning about what I needed in a printer was as fouled up as a bureaucrat's planning. In a few minutes he convinced me that I could do everything I needed to do just as well with the plain MX-80 and a little bit of common sense. (More about that in a later article.)

Then came the end to my original dilemma. MX-80's were being closed out for \$360 by the local dealer. I

compared that to the MX-80 F/T selling for \$510 and the FX-80 costing \$610. I decided MX-80 was too good to pass up at that price, so I bought one. Now all I need to do is figure out how to rationalize — I mean justify — the purchase of an FX-80.

William D. Allen
POB 5155
North College Station
Lubbock, TX 79417-5155 ■

LETTERS TO THE EDITOR

continued from page 10

why I, and others, subscribe (read "pay").

I have had the "100" for about a month. What dazzles me is the blinding speed of the find function. Love it.

G. Mueden -15D
(Metro TUG)
310 West 106 St.
New York, NY 10025

Did "Base 2" Fold?

Your May 1983 issue has a 2-1/4 page article beginning on page 48 titled "Home Run with Base 2."

Please correct me if I am wrong; however, as a former "Base 2" owner, it is my understanding that the company folded a couple of years ago. While one company in "Silicon Valley" was handling some repairs on this printer, I heard that they were running out of replacement parts and, by this date, may have stopped service all together.

It WAS a good little machine, but I believe you will find the company has not been in production for a couple of years.

I feel that the TRS-80 Model 4 was a big disappointment compared to what most of us had been expecting. I am looking forward to your unabashed comments in a future issue.

Joe Ottenstein
P.O. Box 1765
Ormond Beach, FL 32075-1765

We regret that the article in question had been submitted almost two years before it was published. We were not aware that the company had folded.

Errors in "Fast Conversion between Hex and Decimal"

Several errors occurred in the print-

ing of my article "Fast Conversion between Hex and Decimal" that was published in the May 1983 issue. Two mistakes are in the text, and they are not serious enough to reprint, but the program listings contain certain errors which should cause the program to fail.

In the second paragraph, the word "invariable" should be "invariably."

Line 100, the third line in the second column reads "INT(A/16)". It should READ "INT(A/16)+48".

Line 200, the last program line before the section "Hex to Decimal Conversions," reads "(AD-(AD*256))". It should read "(AD-(A*256))".

The word "current" in the fourth paragraph of the second page of the article (page 30) should read "current digit." The sentence should read, "Before we add the current digit to the result, the previous result must first be multiplied by 16."

Line 110 in the last program listing in the article, at the bottom of page 30, reads "IF AD>1". It should read "IF AD>=1".

I hope that these errors did not cause readers any problems.

Alan R. Lowe
18 Bosworth Street
Hammersley 6022
Western Australia

Enhancement to Text Editor

I have found that a small change adds a nice feature to the free Text Editor program on the Computronics subscription cassette: when CLOADing a program from tape, each line is printed to the video display. That is because I added line 2125 PRINT A\$(L). I like this feature partly because I find it fascinating to see a tape recorder cause a written program to print out. I worked with RATT in the Navy, but this is different. For disk systems, also add line 2175 PRINT A\$(L).

I worked at Radio Shack as a repairman for 5 months. That in itself is a fascinating tale, believe me!

Your program makes it a pleasure rather than drudgery to write, and I feel like saying thanks for all the effort you must have put into it.

Pat E. McGee
3530 West 123
Cleveland, OH 44111 ■

RIBBONS & SUPPLIES

SATISFACTION GUARANTEED
FREE SHIPPING in US and CANADA
NO MINIMUM ORDER
FREE CATALOG

DISCOUNTS apply to ANY COMBINATION of products. Deduct 10% for 6 to 23 items, 15% for 24 to 99, and 20% for 100 or more items. SCHOOLS may take an additional 5% discount.

MASTERCARD & VISA orders, send card number and expiration date. Checks accepted with no delay.

FREE SUPPLIES CATALOG includes instructions for loading refill loops. Most orders shipped within 24 hours. Florida residents add 5% sales tax.

REPLACEMENT FABRIC RIBBONS FOR	Ribbon in Cartridge	Refill Loop
Anadex 9500	13.00	3.95
Anadex Word Scribe	13.00	3.95
Axiom GP-80M	6.50	2.93
Axiom IMP I & II	13.50	2.93
Base 2	13.50	2.93
C-Itoh Prowriter	5.50	2.43
C-Itoh F-10	5.50	1.68
Diablo Hytype I	5.50	1.59
Diablo Hytype II	5.50	1.68
DEC LA34	5.50	2.03
Epson MX-70, MX-80	6.50	2.93
Epson MX-100	9.75	4.45
IBM PC (MX-80)	6.50	2.93
IBM PC (MX-100)	9.75	4.45
MPI 88G, 99G, 150	13.50	2.93
NEC 3500	TBA	2.43
NEC Spinwriter	5.50	2.43
NEC PC 8023A-01	5.50	2.43
Okidata Microline 80, 82, 83 (spool)	2.50	
R.S. DW II (1449)	TBA	2.08
R.S. DMP 200 (1483)	TBA	2.93
R.S. DMP 400, (1418)	5.50	1.68
R.S. DMP 500 (1482)	TBA	2.93
R.S. DWP 410 (1449)	TBA	2.08
R.S. LP6, LP8 (1418)	5.50	1.68
R.S. LP3, LP5 (1414)	5.50	2.43

BASF DISKETTES, 5 1/4"

with hub ring, LIFETIME WARRANTY

STOCK#	SIDE/DEN	SECTORS	PRICE/BOX
54974	SS/DD	SOFT	29.20
54976	SS/DD	10	29.20
54978	SS/DD	16	29.20
54980	DS/DD	SOFT	40.50
54982	DS/DD	10	40.50
54984	DS/DD	16	40.50
54986	SS/QUAD	SOFT	42.20
54988	SS/QUAD	10	42.20
54990	SS/QUAD	16	42.20
54992	DS/QUAD	SOFT	54.00
54994	DS/QUAD	10	54.00
54996	DS/QUAD	16	54.00

DATA SYSTEMS

(305) 788-2145

BOX 99 • FERN PARK, FL 32730

PROGRAM PREVIEWS

A. A. Wicks

SCREEN ARTIST II

Several months ago, I reviewed a program on these pages called, "Infoscan." If you read that review, you may recall that I was particularly impressed with the first-rate graphics that were presented in the program.

The producers of Infoscan have another program—"Screen Artist II" for the Model I and Model III that will allow you to create the same type of graphics that were found in Infoscan. The graphics thus produced are not limited to displays that might go with a program being written, such as a border (this is one of the easiest), but may be used for animation, demonstration, educational visualizations, and so on.

Briefly, Screen Artist II functions similarly to a word processor, which, in a sense, it is. You may, in addition to calling upon the graphics arsenal of the TRS-80, use characters as well, and even create your own or foreign characters. The characters, letters and numerals, may be produced integrally with the graphics, hence may be particularly useful for graphics that have word or letter call-outs. In addition, the program will generate a short BASIC language program that may be imbedded in your own programs. This, in turn, will allow you to call the graphics you have created, and these will operate in the high speed of the machine language that the graphics use. The response from BASIC will be in less than 1/100th of a second.

The program is issued on one double-sided disk, one side is Model III TRSDOS 1.3 compatible, the other side TRSDOS 2.3 and NEWDOS80 compatible. Both have 32K and 48K versions. The 32K and 48K versions are the same, with the exception that the 48K version will generate large block letters at a single keystroke for each letter, and will operate in a mode known as "Quick Draw," which will be described in detail later.

Screen Artist II allows you to store on disk the programs you create, from which they are recalled for changes if necessary, as well as for operation.

OPERATIONAL MODES

The program operates within the scope of three different "modes," and you may switch from one mode to another by a single keystroke. Within each mode there are many other functions available, using a one- or two-keystroke operation.

The first mode, Alphanumeric, is what would normally occur at the keyboard when you are not using Screen Artist II. However, the letters and numbers when used with Screen Artist II in operation may be entered anywhere on the screen. By using the arrow keys, the blinking cursor may be placed anywhere, and the required key is then pressed. If a word is being entered it is not necessary to again move the cursor, as it will move to the next character position as in normal operation.

The second mode, Graphics, is probably the heart of the program, and at least it is the highly visible component. Graphics operation with the TRS-80 is customarily a tedious process if done using character strings. Using Screen Artist II, all the hard, slow work is replaced by two simple maneuvers. If you refer to the TRS-80 manual (Level II for Model I), you will notice there a Table of

graphics codes, which also shows the appearance of the graphics character. Assuming that you have defined your planned screen using these characters, you merely enter the code (two-digit, the first "1" is omitted), at the selected cursor position, and the graphics character will instantly appear at that location.

If you have done a little planning in advance, it is obvious that this is going to be a very rapid and painless way to create a graphics display.

The second way to deposit the graphic characters at the cursor position perhaps allows you to be a little more creative as you go along, and does not necessarily require a great deal of pre-planning. Assuming that the keys Q, W, A, S, Z, and X form a pattern identical to the pixels in a six-pixel graphics block, then pressing any one of these keys results in the corresponding pixel being displayed. Therefore, you merely press each key as required until the character you require is complete. It should be mentioned here that the 16th line of the screen is always used as a status line, reporting information regarding codes, etc. In this instance, as you build a graphics character there is a continuous report displayed showing the relative number of the key being depressed (1-6), the graphics character, and the ASCII code of the character presently displayed. Also, the character being built is displayed at low intensity, flickering rapidly, to allow you to see it quickly if your screen is congested with previously created material.

The third mode of Screen Artist II is probably the most complex in terms of operational capability. It is called the "Special Command Mode," and provides more than 25 commands for a wide variety of functions; these are preceded by a one-key control key action. Some of the most interesting special commands and their descriptions follow—there are far too many to describe them all in detail.

One of these is the Block Letter mode. This large block character display was previously mentioned as being available only for the 48K configuration. To give you some idea of the size of these characters, only four lines may be entered per screen. The characters are sharp, clear, and impressive.

Another command will fill the screen with a designated character—which you are prompted to enter from a status line request. Still another will suppress the cursor and the status line, and display the full 16 lines of the screen. This is useful if you would like to see what the screen will look like when displayed from your BASIC program.

There is a command that reverses the video display. The documentation calls this "inverting," which is a misnomer. Performing this (for instance, from your program on a cyclic basis), can be visually attention-getting, you may be sure. Yet another command allows you to "open up" a display either from the right or left of the cursor, or from the cursor position up and down. This will permit the insertion of material in an existing screen, after which the material is closed up.

Rapid positioning of the cursor to the center of the current line occurs using another command. And an

almost instant rectangular border may be placed around the entire screen with one of the single key commands. You are not restricted to the outer fringes of the screen for this function either—by placing the cursor at the upper left corner of the border you wish to create, the border will be constructed from that position. You may also present borders within borders, in the same manner. However, if the cursor is placed beyond line 7 or beyond position 32 on the line no action will take place.

One unusual and interesting command will place you in what is described as "Quick Draw" sub-mode. This too, will only operate in the 48k version of Screen Artist II. With it, you may doodle on the screen to your heart's content—drawing lines that follow the cursor as you move it using the arrow keys. You may erase these lines, or skip over lines you have already drawn. While using this mode, you may also clear the screen entirely, reverse the screen, change the speed of movement of the cursor, and have displayed the coordinate position of the cursor (X, Y, as used in the SET, RESET and POINT commands of BASIC). The manual describes this function as being one that will entertain children of all ages, and I must agree with their deliberate lack of restriction on the age grouping.

The preceding functions are only some of the many available in the Command mode. Others will white-out the screen, save the screen to disk, clear the screen partially or completely, scroll the screen, and shift the screen right and left. There are also some auxiliary commands that use the numerals 1 through 4. The first "1" will store a screen in a "hold area" or buffer, and there are three sub-functions within this command. One duplicates the screen in the buffer, without affecting the current video screen. This is somewhat akin to filing data that you may decide you want later, at the same time as you modify existing material. In this case, it is the graphic display that you may modify. The next sub-command allows you to retrieve the screen stored in the buffer, and the third sub-command permits superimposing the buffered screen on to the current screen. This, in effect then, is an overlay, and can be useful in combining two separate graphics screens.

The remaining three digits—2, 3, and 4, operate as commands that permit using additional buffer zones for storage, exactly as described for the digit "1." If desired, all four holding zones may be used simultaneously. This, to me, appears to be an excellent way to create animation, as well as to overlay graphics.

SAVING, PRINTING, MERGING

At this point, you would store your graphics on disk, and the program hand-holds you through the entire process from naming your file to printing on the video monitor or printer.

The option of using the printer must be defined at one of the points in the program where it is offered, at which time "LPRINT" statements are automatically included. Unfortunately for owners of printers that will not (repeat:"not"!), print TRS-80 graphics, they cannot take advantage of this very fine aspect of Screen Artist II. Therefore, for the most part, those without this capability will be limited to displaying their graphics on the monitor only—something they should be aware of, when considering purchase of this program.

Screen Artist II permits you to change existing screens,

and this is called from the menu. Essentially, the graphics procedures are the same as for creating a new screen display.

It is extremely simple to merge a Screen Artist II graphics program that you have made, into your existing BASIC program. In fact, it is no different than the normal TRSDOS "Merge" procedure. When completed, your BASIC program includes the Screen Artist II-generated BASIC statements. There is a warning here, however; your program lines and the Screen Artist II program lines cannot be in conflict; you determine this at the time you are preparing the graphics. A GOSUB from your BASIC program will call the Screen Artist II graphics subprogram. Needless to say, you may use this (and other) GOSUBs as many times from your program as desired. When the GOSUB is encountered as the program executes, the screen will be displayed instantly on the video monitor, or, if specified, printed on the printer. I used this quite effectively for a program I had written, wherein a map of the world is frequently called for display during the operation of the program. It should be noted that you will save the merged program under a new file name (but it could just as well be the same name as your BASIC program once the operation has been proved).

Regarding the preceding merge operation, it is a comfortable feeling to have an operating program, complete with the graphics you have incorporated, safely tucked away as a disk file. No need to go back to Screen Artist II the next time you use your BASIC program—all that is needed is within your merged program—you just "run" and the graphics appear when required in all of their simplicity or intricacy as you have prepared them.

Incidentally, you will still be able to edit your BASIC program at any time, should you need to; however, you may not tamper with the Screen Artist II lines that you will see when you list the program. This would destroy the graphics (and possibly other parts of the program).

MANUAL

The manual for Screen Artist II is quite a bulky document, comprising 25 pages of single-side printed 8 1/2 by 11 inch sheets, stapled top-left. This number of pages produces a criticism from me—it need not be this large, and is really over-documented and frequently redundant in content, if not in exact wording. In some places in the manual, information you need in order to proceed is not where and when you expect it to be, but off in some preceding section. This gives you the uneasy feeling that you have probably read the information you need, but have only a vague idea that it is "someplace." This is compounded by the fact that there is no Table of Contents and no Index. I found that I was re-reading much material unnecessarily in seeking out some detail in order to proceed.

The net result can be called "poor organization." The manual certainly does not suffer from a lack of information—it's all there, and in exquisite detail. The largest section of the manual, "Instructions for Use," is 13 pages long. Information (if not all) of the non-repetitive material from the first Section (dually called "User Guide to Key Functions," and "Tutorial," depending on the page you are on), could easily be merged and shortened within this same section.

continued on page 33

BEGINNER'S CORNER

Spencer Koenig

Assemblers — More than One Way to Skin a CPU!

Well, it's that time again. One of the things that I really like about working with computers is that there's always something new just around the corner. It feels just like Christmas all over again.

I've just gotten my hands on a new little word processor that's a dandy. At this point it seems every bit as good as Superscripsit and as easy to use as Pencil 2.0. The more I use it, the more I like TYPITALL. What I really like about TYPITALL are its' help screens that you can modify to create your own "user-definable" on-line documentation. There's no need for large books you never want to have to look at. I recommend it to you.

Well, this month we continue with our exploration into the world of compilers and interpreters and talk a little about the mysterious world of assemblers. As I began to do research into the subject, I found out things aren't so clear cut. Everyone seems to add their own bells and whistles, but the basics are pretty much the same, as you'll see.

First, for those of you you have missed my previous group of articles covering compilers and interpreters, here's a brief recapitulation (one of my favorite all time words).

An interpreter takes a program written in a high level language like BASIC and looks up a routine for each word encountered. If the word or command is known, then the routine associated with it is executed. No matter how many times the word is encountered, it is always looked up and executed. That's the reason why interpreters take so long to "RUN" a program.

Compilers, on the other hand take a high level language source and comb through it from one to several times, organizing specific routines that are to be repeated, at the same time checking for errors. The output is in the form of another program (not an action like interpreters) that is directly usable by the processor—i.e. a machine language program.

There are, however, compilers that output intermediate programs that are themselves interpreted at a much faster rate than high level BASIC. An example of this kind of program is the PASCAL P-CODE system. This kind of compiler is said to create programs that are transportable to other machines that have their own version of the P-CODE interpreter.

Now, we get to the 64 dollar question. What is an assembler? An assembler program is a program that takes a low level language (like assembler) and converts it to machine code, directly usable by the machine for which it's intended.

Here comes the part that begins to open the door. What do we mean by "high level" and "low level" language? First, let's make an assumption. Let's assume that we use a high level language when we speak. Most people do, but there are always exceptions to every rule. When we use "an expression" or an idiom, you find that the individual words cannot be directly translated and maintain the same meaning (if you've taken high school French, then you know what I mean), but those you're speaking to understand what you are say-

ing. Though you may use one or more words, there may be several meanings. A single instruction may have several steps, and everyone listening to you would automatically know what those steps are.

A similar thing can be said for computer languages. In a high level language, one word can be translated into many steps that the computer can use. If we were to examine the interpreter in ROM, we'd see that a simple word like CLS executes a multi-line program in machine language (if you use a disassembler you can see that for yourself).

In contrast, we have the idea of a LOW level language. Low level means that you are getting closer to the level of language that the computer can use directly. Assembly language is the lowest level language that we can use with any ease and still deal with characters and words that have meaning.

An assembler uses mnemonics or shorthand symbols to represent the actions that they perform. One of the easiest to interpret is the command JMP, which is the mnemonic for jump. In BASIC we use the command GOTO to accomplish virtually the same thing.

The assembly language is said to be the lowest level language next to machine code because it is on a one to one relationship with machine code. That is to say, an instruction like JMP is converted to a number that corresponds to a specific single machine code instruction. So, we have one mnemonic and get one machine instruction.

Incidentally, you could write an assembler in BASIC that would use PEEKS and POKEs. If you could make a list for every instruction possible on the Z80, you could have the program output a list of numeric code that could be poked into the computer like the more fancy assemblers around. The problem is that there are hundreds of instructions, and such a program in BASIC would be huge. There are some assemblers in machine code that are also rather large due to the many options around. That's why there are many kinds of assemblers to get the job done.

Why do people use assemblers and want to write in assembly code? That is a good question, considering that there are many good alternatives, such as compilers. Well, the fact is that most compilers are not that precise about how they translate the high level code to low level code. To paraphrase and destroy the expression, there's a certain amount of efficiency lost in the translation.

There are times when this loss can be a critical factor in getting a program into a machine. Also, there are just some people who enjoy having complete control over the computer and taking complete responsibility for the success or failure of a program (I want to do it myyyyy way).

Writing assembly code means never having to say "the compiler did it." Assemblers also are used to "LINK" together various programs and routines into a larger single program. That's something that is a little dif-

continued on page 44

BUSINESS COMPUTING™

THE PUBLICATION FOR
SERIOUS SMALL BUSINESS
COMPUTER USERS

PUBLISHED BY

COMPUTRONICS
MATHEMATICAL APPLICATIONS SERVICE

50 N. PASCACK ROAD
SPRING VALLEY, NEW YORK 10977
(914) 425-1535

PUBLISHER: Howard Y. Gosman

EDITOR: Andrew Hofer

The purpose of BUSINESS COMPUTING is to provide and exchange information of interest to owners of under-\$10,000 computers. BUSINESS COMPUTING depends on reader participation for much of the information in each issue. Please send questions, comments, stories, press releases and any other information of interest to: The Editor, BUSINESS COMPUTING, 50 North Pascack Road, Spring Valley, New York 10977.

*** NEED HELP RELATED TO CP/M? ***

Digital Research, the creators of (and highest authorities on) the CP/M operating system, will cooperate with The Source Telecomputing Corp. to create a new on-line information service for CP/M-related questions (accessible through The Source.)

After dealing with CP/M users for many years, Digital Research's technical support center has realized that 80 percent of the questions asked them can be found on a list of only 100 basic questions -- a perfect set up for an on-line system. In addition to specific answers to the "100 most popular questions," users will get references for related articles and hardware and software reviews. The service will not be interactive at first, but may become so if expansion of the service seems desirable. The system's information will be updated weekly to give users the latest possible developments.

*** PROGRAM ANALYZES PHONE BILLS ***

LONG DISTANCE ANALYZER is a menu-driven business program that streamlines accounting for long distance telephone costs. It can save you time and money by organizing your phone bills, identifying

ADVISORY EDITORS: Dr. Peter Shenkin

Richard Kaplan

Beatrice Kahn

Steven Kahan

parties called, producing totals and reports, and analyzing geographic patterns.

The application itself is an important one for many different types of business. When a company's phone bill is more than \$5000 per month, the business may be able to afford special hardware to track its telephone expenses, but there are vast numbers of businesses or departments using microcomputers that do not have bills that large. They typically rely on antiquated manual methods of verifying and accounting for long distance expenses. These methods are extremely time-consuming and are generally ineffective.

You can use this program to cut abuse and waste, bill phone costs to your clients, recover phone company billing errors, evaluate special services (like WATS), print an alphabetical directory, and do cost accounting by your categories. You can accumulate monthly bills for long-term analysis.

The LONG DISTANCE ANALYZER is available for the IBM PC, and TRS-80 Models II, III, 4, 12 and 16. The IBM version requires 64K, 1.0 or 1.1 DOS, and one floppy drive. The TRS-80 versions require at least 48K, TRSDOS, and two disk drives. A printer is very helpful, but not necessary. This program has been field tested for more than two years in the TRS-80 line. It has been purchased by a wide variety of businesses, including construction companies, foundations, law firms, accountants, catalog companies, communications consultants, educational institutions, governmental agencies and trucking companies. The newest version contains several refinements suggested by users and many additional error protection devices and user-friendly features.

LONG DISTANCE ANALYZER is priced at \$195 and can be ordered from Golden Braid Software, Attn: George Reardon, 1450 Ranchero Drive, Sarasota, FL 33582; (813) 371-0388.

*** ANOTHER IBM WORKALIKE ***

Toshiba has jumped into the 16-bit computer market with the introduction of its TS300 Personal/Business Computer. The new computer uses MS-DOS 2.0 and is capable of reading IBM MS-DOS diskettes.

Equipped with one diskette drive, the standard TS300 will sell for \$2495 (which includes monochrome display). A two-drive system will cost \$3190, and an upgrade to color monitor will cost \$700.

Supplied with 192K of memory, the TS300 can be expanded to 512K. The disk drives are quad-density, and can are capable of storing 640K each, but only 320K or 192K format IBM diskettes can be read.

TBASIC-16, Toshiba's 16-bit BASIC Compiler, is included with the MS-DOS operating system (both were written by Microsoft). For \$295 more, you can get CP/M-86 with a CBASIC compiler.

The displays are particularly notable: both have a very high graphics resolution of 650 by 500 pixels, and the color monitor can display eight colors simultaneously (the IBM PC can only display four).

*** SHARP INTRODUCES A VERY GOOD PORTABLE COMPUTER ***

Sharp Electronics Corp. has introduced a computer that is really the most interesting portable computer to appear yet. Hot on the heels of Radio Shack's innovative Model 100, the new Sharp machine (called the PC-5000) steals the spotlight in the market for lap-sized portables.

Several features of Sharp's PC-5000 make this machine particularly notable. First, the computer's 8-line by 80-character LCD display is the largest yet seen on a portable computer -- it's twice the size of the Model 100 display (and there's an even bigger display coming from Sharp soon -- see below).

Another big feature is the optional 80 character thermal/impact dot matrix printer that installs in the back of the unit so that it looks built-in. The printer can be used with heat-sensitive thermal paper, or can give correspondence quality printouts on regular bond paper, using a carbon film ribbon. The ability to carry a full-size printer around with you will be very important for salespeople, reporters, and others who need a computer to accompany them on the road.

The standard PC-5000 is equipped with 128K of RAM (expandable to 256K), built in ROM (read-only memory) software for word processing and telecommunications, a



a rechargeable battery and an AC power adapter.

A version of the MS-DOS operating system will be available for the PC-5000, and a line of applications software will be available in bubble-memory plug-in modules. Some of the first to appear will be a spreadsheet analysis program, database system and an executive planning program.

Double-sided double-density diskette drives can be attached to the computer, and (presumably under MS-DOS) the diskettes will be data-compatible with those used in the IBM Personal Computer. A cassette interface can also be used for storage on tape.

This computer is a very impressive addition to the lap-sized computer field, and is proof that these machines are to be taken very seriously. Although we've used the TRS-80 Model 100 for comparison with the new Sharp computer, Radio Shack should have nothing to worry about, because they are not really competing for the same markets. Although the PC-5000 is undeniably more powerful than the Model 100, it also has a \$2500 price tag -- more than three times the price of Radio Shack's machine.

*** SPEAKING OF LARGE LCD DISPLAYS . . . ***

Sharp is readying the largest commercially-available LCD display yet. This one will display 80 characters by 16 lines, twice as much text as the screen on their PC-5000 (see above) and four times as much as the TRS-80 Model 100 (which had the largest screen until the PC-5000 came along).



This panel is only 18mm thick, and can display graphics with a resolution of 480 by 128 dots. Another new feature of this display is the new formula for the liquid crystals used. On most LCD displays, there is only a narrow range of angles from which you can view the screen. Change the viewing angle too far, and the screen's contents seem to just disappear. The Model 100 provided a control to change the viewing angle, and presumably the PC-5000 is controlled the same way. But, because of the new formula (a company secret), the new screen can be viewed across a 25 degree range of angles. Another new look: the display is reportedly blue on a white field! Not the dark grey on light grey that we're so used to.

This brings these displays up to about a half-page of text at one time. That's getting very good, and we'll be seeing full-page displays very soon -- like one year or less from now.

*** MITSUBISHI'S SEVEN-COLOR GRAPHICS PRINTER ***

A new printer from Mitsubishi will be capable of printing high-resolution graphics in seven colors. Especially interesting is the fact that this is a thermal printer, which uses a four-color transfer ribbon cartridge to produce seven colors. Resolution is said to be near-letter quality -- with 1440 dots printed per line. The printer is quiet, and produces copy at 45 characters per second. The price is expected to be about \$1500, in serial and parallel interface versions. Look for this printer around October or November of this year.

*** REAL ESTATE OVERLAYS FOR MULTIPLEX ***

MULTIPLAN, Microsoft's award-winning spreadsheet program, has been a big part of the "spreadsheet revolution" started by VISICALC. Now a lot of businesses have popped up selling the programmed "overlays" or "templates" used by these programs. The templates are used to set up the spreadsheet for a particular application, and purchasing a ready-made template allows the user to run that application without having to learn how to program the template.

RealData, Inc., is offering an package of six templates for MULTIPLAN that set up the system for real estate property analysis and related calculations. This package produces a complete sensitivity analysis for income producing properties. Using the overlays with MULTIPLAN, you can generate a 10-year analysis of cash flows and sale proceeds with automatic computation of depreciation (including capital improvements); annual debt service, annual interest and payoff amounts for three mortgages; capitalization rates, net operating income, before and after-tax cash flows, costs of sale, cash-on-cash return and internal rate of return; projected selling price by capitalization or inflation rate; adjusted basis, capital gain, excess depreciation, recapture and federal income tax calculations. You can also produce complete loan amortization schedules, IRR calculations, and comprehensive annual income-and-expense statements.

These templates are available in most disk formats (for CP/M-based systems) for \$120 plus \$5 shipping & handling. Contact RealData, Inc., P.O. Box 691, Southport, CT 06490; (203) 255-2732.

*** THE DAISYWRITER 2000 PRINTER ***

This is not just another daisywheel printer! This one packs in more features for a lower price than any we've seen yet. Made by Computers International, this printer will work with any computer, imitate virtually any other daisywheel printer, is truly multilingual, and has its own printer buffer. This is a very smart printer -- it has four internal CPU's.

By flicking a few switches on the DAISYWHEEL 2000, you can make it work according to the signal protocols of the DIABLO 630, the Qume SPRINT 9/11, NEC's 3500, 5500, or 7700, the Centronics 737 or the IBM PC printer (Epson MX-80). Don't worry about physically connecting the printer to your computer--it can directly connect to standard Centronics parallel, RS-232, IEEE-488, or 20-milliamper current loop interfaces. Normally, you would have to set the correct baud rate on the printer, to match the computer's transmission rate. Not on this printer -- you just send a carriage return to the printer and it automatically recognizes the baud rate and sets itself to the same rate! Connecting a daisywheel printer to a microcomputer and getting it to work is often a trying experience, but with this printer there would be no such problems.

By using a set of changable printwheels and flipping a few more switches, you can make the printer output text in sixteen different languages, with complete character sets for each. The printing is truly proportionally-spaced, with character-width tables built-in. But you're not stuck with those tables, since you can create your own under software control. The printwheels also incorporate a new development. They're cartridge printwheels, and they'll undoubtedly be much more resistant to damage than ordinary printwheels.

The best part is the built-in 48K printer buffer. All printers will need these soon to stay competitive. This buffer allows your computer to send many pages to the printer at very high speed, and the printer's buffer stores them to be printed out as fast as the print mechanism can handle it. In the meantime, your computer is free to continue working on another program. And your office won't get disrupted by the noise. This printer is very well soundproofed.

There is even a graphics print mode, which provides absolute or vector plots for creating graphs and charts. This mode is compatible with Diablo's HYPLOT.

Essentially, this printer combines just about every good feature of any daisywheel printer into a single machine. It's priced at \$1495 with an optional \$50 cable and \$150 tractor feed mechanism. That price would be pretty good just for an ordinary daisywheel printer -- for this one it's a fantastic bargain.

*** UPGRADED CLASS SCHEDULING PROGRAM ***

CMA Micro Computer has announced the release of a new version of its popular Class Scheduling Program. The advanced version offers new editing procedures for altering groups of courses in the system's master schedule.

The Class Scheduling Program allows schools with up to 2,400 students and as many as 999 courses and sections of courses to analyze the master schedules and prepare individual student schedules. The system allows for the automatic entry of required courses and the fast entry of any optional request and alternates. The system will provide course tallies and conflict matrices for all requests.

The system will schedule individual students and report anyone not scheduled in a requested course. Non-scheduled students can be given new requests and rescheduled until all students have been scheduled. Schedules can be printed for all students and rosters prepared for all courses offered. Adds and drops can be easily handled with forced scheduling or by re-scheduling of individual students using the program's automatic scheduling elements.

The system is designed to work with CMA's Grading programs and Attendance Bookkeeping system. The system requires a 48K Apple with Applesoft, two disk drives and an 80 column printer. Versions for the IBM Personal Computer are under development.

Additional information and demonstration versions are available from CMA dealers or directly from CMA Micro Computer, 55722 Santa Fe Trail, Yucca Valley, CA 92284; (619) 365-9718.

*** KAYPRO II UPGRADE KIT ***

There has been considerable concern among dealers of KayPro II computer that they may be swamped with requests for trade-ins when the new KayPro 10 hard disk-based system becomes available.

This has prompted Non-Linear Systems (the KayPro's manufacturer) to make plans to sell an upgrade kit that will allow owners to convert their KayPro II's into KayPro 10's, adding a hard disk drive, graphics capabilities, and additional KayPro 10 software.

The KayPro 10 will be selling for \$2795, compared to the \$1795 price tag on the KayPro II. Thus, they will sell the upgrade for \$1000 -- the difference in price between the two machines.

*** TALKING SPREADSHEET ***

The first spreadsheet calculator program that you can talk to will be available by the time you read this column. SuperSoft is adding voice recognition equipment to its ScratchPad financial planning program.

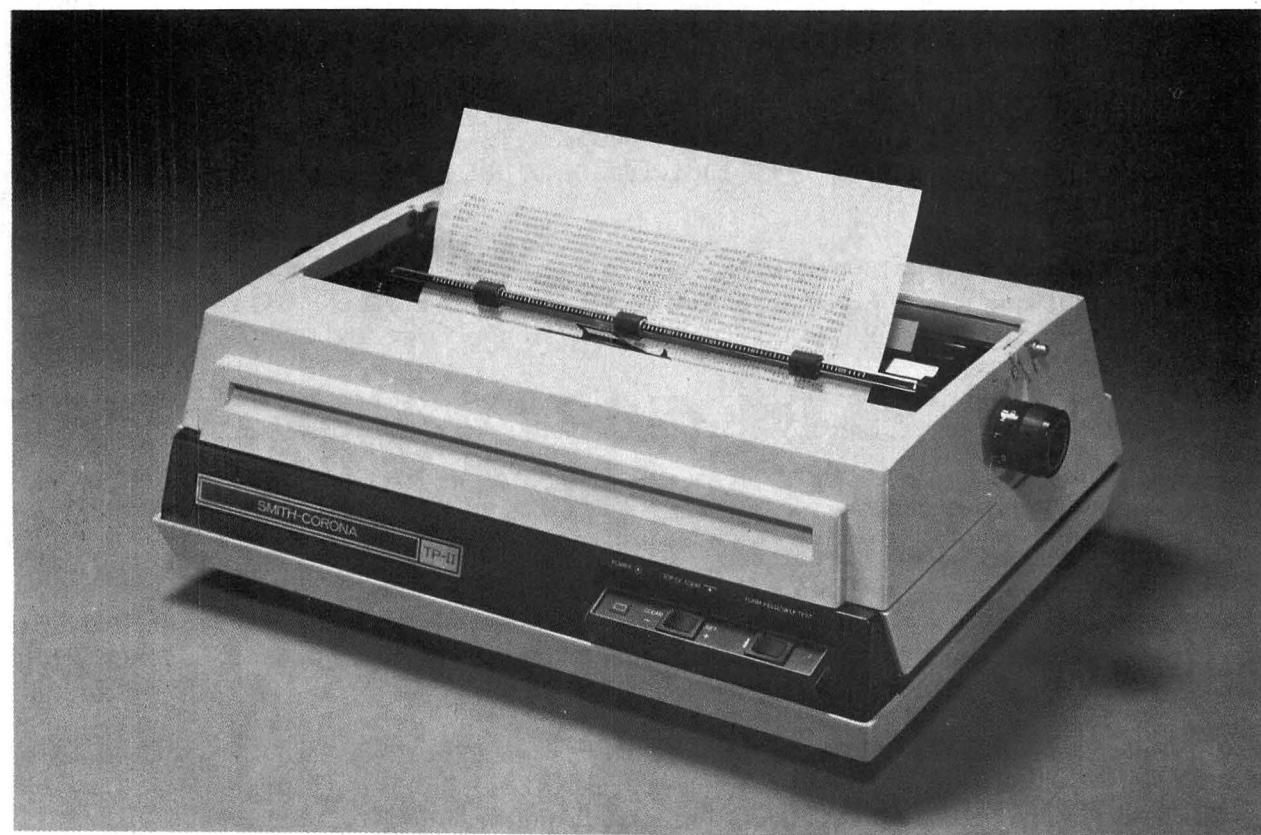
According to SuperSoft, anything you can do with the keyboard can now be done with voice commands. The program, with its new VoiceDrive, can interpret 97 words. You can load files, arrange columns, and do just about everything with the program by just talking to your computer. Once ScratchPad is programmed, you can sit back and perform all of your various analyses without touching the computer.

To program the system, the user repeats each of the 97 command words three times, to be stored in a voice file. When you run the system, you load your own voice file first, so the computer will understand you. This means, of course, that any number of people can use the same program, regardless of differences in accent or tone of voice.

The program will sell for \$495, and will be available at first only for the IBM PC, using a voice recognition card from Tecmar, Inc. (the leading manufacturer of IBM-compatible peripheral equipment). The program will be available "bundled" with the Tecmar board for about \$995. It will probably also be possible to upgrade current systems using ScratchPad to include the voice recognition features.

*** NEW PRINTERS FROM SMITH-CORONA ***

Smith-Corona has introduced a couple of new machines -- a new version of their daisy wheel printer, called the TP-II, and a memory typewriter that can plug into a computer, allowing it to be used either as a letter-quality printer or a memory typewriter.



The TP-II is a dual interface letter quality printer that is compatible with virtually all personal, home and small business computers. The TP-II offers a number of significant improvements over their original TP-I. The TP-II features:

Both RS232 serial and standard Centronics parallel interface ports in the printer.

A new 10/12 pitch ASCII 93-character printwheel with reverse slash, brackets, approximate sign (tilde), vertical line, up-arrow, greater-than and less-than symbols -- all the characters needed to print out computer program listings.

Capability of handling X-on/X-off and ETX/ACK along with hardware handshaking protocols to assure maximum computer compatibility.

Self test switch to automatically print out a test pattern as a rapid check on the printer's operating condition.

Easier setting of switches for baud rate, parity and character length, and for a foreign-language wheel. And the user can quickly reverse the polarity of the busy signal and control the carriage return and line feed responses of the printer.

Automatic underscore enables the user to underline words or sentences automatically.

Automatic carriage return.

Programmable margin and tab features allow the user to set and release the margins and tab positions under program control.

A built-in 256-character buffer.

Low-cost, easy to change printwheels and ribbon cassettes.

* Designed so the user can quickly and easily install an optional tractor-feed mechanism for continuous fanfold paper.

"Through advanced engineering, Smith-Corona has produced a significantly quieter text printer . . .," said Mort Gutman, Director of Printer Marketing. "The noise level is kept to a minimum."

Smith-Corona will support users, with an "800" number for any operating and servicing questions. In addition, to provide fast and efficient servicing of its printers, Smith-Corona will soon more than double the size of its network of Printer Service Centers to over 50, operating in locations nationwide.

The TP-II will sell for \$895 and will be available in August. The optional tractor-feed attachment is priced at \$149.



Smith-Corona has also introduced several new memory typewriters, one of which can be connected to a computer for use as a letter quality printer. The Memory Correct III Messenger portable electronic typewriter uses an optional "Messenger Module" to plug into

almost any computer.

The new typewriter is equipped with a wide range of features usually found only on far more expensive electronic typewriters. It uses daisy printwheels, presently offering 10 different typestyles. "With all of the conveniences offered in the new Memory Correct portable electronic typewriters, the keyboards remain simple, uncluttered and easy to operate," said Theresa Conetta, Smith-Corona's Director of Custom/Citation Marketing.

To transform the typewriter into a computer printer, the user just plugs it into Smith-Corona's Messenger Module interface and from there into the computer. The module is equipped with both RS-232 and Centronics parallel interfaces. Other features of the Memory Correct III Messenger include:

Full-line correction, for neatly removing up to 157 characters with one touch.

Multiple pitch, allowing a choice of 10, 12 or 15 characters per inch.

Automatic relocation to original position after corrections.

Automatic carriage return for non-stop typing.

Automatic centering of headings and other text.

Automatic underlining.

Memory speed "cushion" (buffer) to eliminate overstrikes, dashes and voids.

Reverse tab to facilitate intricate statistical typing.

The Memory Correct III Messenger will sell for \$599, with an additional suggested price of \$170 for the Messenger Module.

*** NEW VISICALC TEMPLATE FOR DOCTORS ***

CMA Micro Computer has announced the release of a new spreadsheet template for the medical profession, designed to run on the Apple III computer. The application, called MediCard, allows the small medical practice to handle their complete private billing and AMA universal claim form preparation using the program and their own copy of the Apple III Information Analyst package.

The package allows the user to quickly enter their practice billing address and other necessary information to customize the package to their practice. Patient files can be created for individual patients or families. As transactions occur they can be entered onto the system for end-of-the-month billing, immediate time of service statements or claim form preparation.

Patient records can be quickly recalled for bill posting or on-screen review of their payment status. The system allows for the use of such industry standard codes as RVS or ICD-A. Fees which are not billable under insurance plans can be quickly flagged and billed immediately to the patient's private account.

The Apple III MediCard system requires an Apple III with at least 128K and an 80-column printer. The system is optionally compatible with the Profile or Corvus systems hard disk. The package is available for \$249.95 from authorized CMA dealers. Versions are also available for the Apple II or IBM Personal Computer running VisiCalc. Users of SuperCalc and Multiplan should contact their dealers for special versions.

Additional information is available from CMA Micro Computer, 55722 Santa Fe Trail, Yucca Valley, CA 92284; (619) 365-9718.

PRACTICAL BUSINESS PROGRAMS

Steven M. Zimmerman, Ph.D. and Leo M. Conrad

Month #8: MILES and MOVE

Copyright® Zimmerman and Conrad 1983

This is the last in our series of articles on our general ledger package. We will review two programs, MILES for aiding those business persons with the need to keep track of gasoline mileage for tax and costs purposes and MOVE a utility program to aid individuals in moving files from disk to disk.

A list of the publication schedule for the other parts of our general ledger program follows:

No	Month	Program
1.	Jan. 83	GLMENU (Controls use of routines).
2.	Feb. 83	NAME & START (Initializes chart of accounts and firm's name).
3.	Mar. 83	TRANSACTION (Inputs monthly transactions).
4.	Apr. 83	CPA (Produces Balance sheet & Income and Expense statements, Profit & Loss statement).
5.	May 83	UPDATE & YEAR (Changes chart of accounts and performs year end closing tasks).
6.	June 83	TRIAL (Produces trial balance).
7.	July 83	BALANCE (Balances check book).
8.	Aug. 83	MILES & MOVE(Calculates gas mileage and moves files between disks).

The operation of MILES will be detailed, followed by a review of the operation of MOVE.

RUNNING MILES

This program is designed to make it easy to calculate the cost of operating a car for a given month. The assumption is you are charging your company a fixed amount per mile for using your car.

The first thing you see after calling the program from GLMENU is the following:

THIS PROGRAM IS DESIGNED TO CALCULATE GASOLINE MILAGE COST
STARTING MILES 9999 TO END?

You may input miles as 232 or 52232 etc. As long as you input starting miles and ending miles in a similar manner you will be alright. Even if you err and input ending miles for starting miles etc. you will be alright since the program adds the absolute value of the difference between the starting and ending miles. The next question is:

ENDING MILES?

After giving an answer to the above, the computer will print out the difference between the two numbers.

What Is:

- Lightning Fast
- Powerful
- Awe-Inspiring

What It Does:

- Loads source text in seconds.
- Utilizes 75% of disk space vs. EDTASM format.
- Creates object code for 794 instructions (EDTASM 696).
- Creates object code 2½ to 3½ times faster.

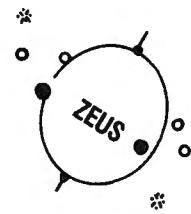
Why Do You Need It:

We think it is the **Most** powerful Editor Assembler in the Universe!

CEC Introduces

ZEUS

Editor Assembler



**INTRODUCTORY
Price \$79.95**

which includes an operating system that will BOOT and run in the Model I, III & 4!

COSMOPOLITAN ELECTRONICS CORPORATION
5700 PLYMOUTH ROAD • ANN ARBOR, MI 48105

Technical Line: (313) 668-6660 Toll-Free Orders: 800 392-3785

—DEALER INQUIRES WELCOME—

Add \$3.00 shipping & handling.
Foreign orders add \$10.00.
Michigan residents add
4% sales tax. COD add \$1.50.
Personal checks take two weeks
to clear. VISA & MasterCard
accepted.

When you have completed your data, input answer 9999 to the starting miles question. You may do this at any time and then return to adding more values if you wish.

After inputing 9999 you will get the following results:

SUM= a number

MILES MENU

- A ADD MORE VALUES
- C CHANGE VALUES
- M CALCULATE MILEAGE
- P PRINT RESULTS ON CRT
- LP PRINT RESULTS ON LINE PRINTER
- R RETURN TO GLMENU?

If you feel you have not made an error, either print the results on the CRT or printer before trying to make changes. On either device the transaction number, the starting miles, the ending miles, the difference and the sum will be printed out if requested. After printing out the input you can go to C above and answer the following question:

NUMBER TO BE CHANGED, NEW STARTING MILES,
NEW ENDING MILES?

After you have completed the changes desired you may wish to print out your results. Note we did not make any provision to save this file of information. A printout of the above is the only record available after you have returned to GLMENU.

RUNNING MOVE

It is sometimes necessary to move files from one disk to another. A good way to do this is with the COPY command under the TRSDOS operating system or other operating systems. There are occasions when it is desirable to move files without leaving BASIC. MOVE was written for these cases. Some of you may prefer to use MOVE to COPY.

After specifying M in GLMENU you will see on the screen the following:

THIS PROGRAM MOVES FILES TO NEW DISKS
MAXIMUM NUMBER OF TRANSACTIONS?

We used 250 as the answer to this question.

THIS PROGRAM DOES NOT DESTROY OLD FILES IT
ONLY CREATES NEW FILES *** USE KILL TO REMOVE
UNWANTED FILES

MENU

- OB MOVE OLD BALANCE/P&L
- R RETURN TO GLMENU
- TR MOVE TRANSACTION FILES?

The process of moving balance sheet files and transaction files, as far as you are concerned, looks very similar. We will only review the procedure for moving balance sheet and profit & loss files. The answer we made to the above question was OB.

NAME OF OLD BALANCE SHEET FILE?

We used as our example in the START and UPDATE routines an old balance sheet file called AUGUST2. We will now copy AUGUST2 into an additional file called JUNK. In practice we often copy DEP (a transaction file) and rename it by the current's month name so as to avoid the task of retyping our depreciation expense transactions.

DISK?

It is possible to have two different files on two different disks and to select one of these files by specifying the disk number. We also allow you to specify the disk on which the new file will be written.

SETUP NEW DISK DRIVES ENTER TO CONTINUE?

At this point you are given time to switch disks in any of your disk drives. Please note the disks in drive 0 must have a system on them. Those in any other disk units may or may not have a system.

NAME TO BE USED FOR OLD BALANCE?

As already noted we will use the word JUNK for the answer to the above question. Since we do not wish to save this file the word JUNK tells us it is just something we created for temporary use and it may be destroyed at any time.

DISK?

We placed this file on disk drive 0. At this point you can hear the disk drives working and the information being recorded.

As in the case of most of our programs, when an activity is complete you will be returned to the main MENU of the program. MOVE works in this manner. Lets assume we now return to the GLMENU and our running of MOVE is complete.

EXAMINING MILES

MILES starts in line 10 with the task of clearing the screen. In line 20 the limits on the program are set by the dimension statement. We used 200. In line 30 the title is printed on the screen.

The program begins the calculation of mileage in line 40. The job is completed in line 60. In line 70 the sum of the miles used is printed and in lines 80-100 the main action menu is printed.

The format of the output is determined in line 110. Lines 120-140 route the program depending on the selection in the menu. Lines 210-230 is an error correction routine. Lines 240-270 produce hardcopy. Line 240 is for those with operating systems with line counters.

Lines 280-290 are for adding the cost per mile calculations to the program.

EXAMINING MOVE

Room for string variables are reserved with the CLEAR 1000 statement in line 10. The screen is cleared in line 20 and the title printed out. Input data is used to establish the dimensions in line 30.

Line 40-50 print the function of the program and the action menu. Lines 60-80 direct the program flow depending on the decision in the action menu.

Lines 90-180 move transaction files and lines 190-50 move balance sheet files.

PROGRAM LISTING: MILES

```
10 CLS: SU=0: REM "MILES"
20 DIM SM(200),EM(200),D(200)
30 PRINT "THIS PROGRAM IS DESIGNED TO CALCULATE GASOLINE
MILEAGE COSTS":I=0
40 I=I+1: INPUT "STARTING MILES 9999 TO END";SM(I): IF SM(I)=
```

continued on page 34

RESPONSE: CUSTOMER RESPONSE TIME CALCULATOR

Dennis P. Avola

ABSTRACT

No one likes to wait; yet waiting is a part of everyone's daily life. Consequently many people have developed tolerance levels for waiting. Our customers are no different; however, once the customer's tolerance level is exceeded, his behavior can become counter-productive to your organization.

Consider a customer waiting for a service technician to repair a life-support system or a computer network that is non-operative. With each hour of system downtime costing the customer thousands of dollars; there is little tolerance for waiting. The penalty for having the customer wait "too long" can range from slow invoice payment to de-installation of system hardware; while responding "too quickly" becomes an expensive proposition.

The key is to balance the cost associated with waiting "too long" against the cost associated with the prevention of waiting in order to maximize profits. For the field service manager to achieve this balancing act, he must understand the dynamics of customer response times- (the elapsed time between a customer's service call and the technician's arrival at the site).

The dynamics of customer response times as depicted in exhibit A clearly illustrates the impact of the technician's productivity rate and the mean time to service (travel and labor included) a customer's equipment. As the mean time to service an account increases, customers wait longer for service provided the technician's productivity rate does not change. However, an increase in the technician's productivity rate will reduce the customer's waiting time for service significantly. The optimum strategy for the field service manager is to reduce the mean time to service and increase the technician's productivity rate.

This paper introduces a computer model designed to calculate the customer's waiting time (response time) based on inputs from the field service manager.

INTRODUCTION

"RESPONSE" is a waiting time computer model for the field service manager. "Response" calculates the elapsed hours between the time a customer places a service call and the time the service technician arrives at the customer's site. Response time calculations are performed for each record on file. The user may specify records as service territories, branch offices, districts, or regions. A maximum of 300 records may be stored on file (48K RAM capacity).

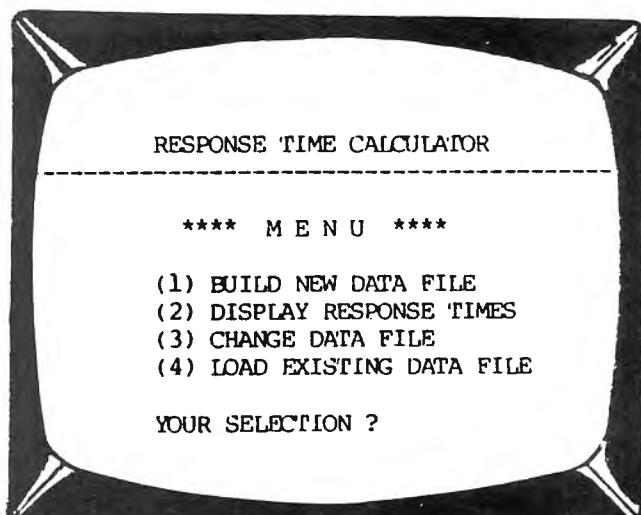
MENU

(see top of next column)

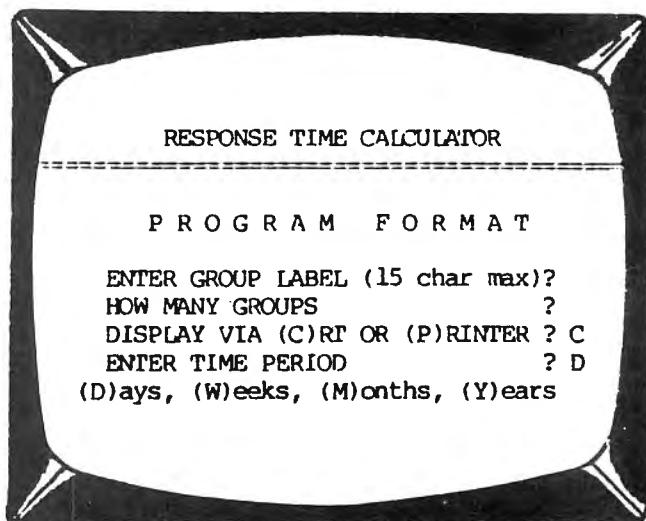
(1) Build a new data file

This menu option prompts the user to first format the data file. As depicted in screen #2, the user specifies the record label (15 characters max), the number of records, display via CRT or line printer, and the reported time frequency of customer service calls (daily, monthly, weekly, yearly).

(See next column)



Screen #1



Screen #2

The data input routine, as depicted in screen #3, illustrates the input requirements for eight service territories. The user is prompted to enter data from left to right on the screen. Changes are allowed after the user completes each input routine screen. Data may be saved on disk after the user completes input routine.

(See next page for Screen #3.)

(2) Display response times

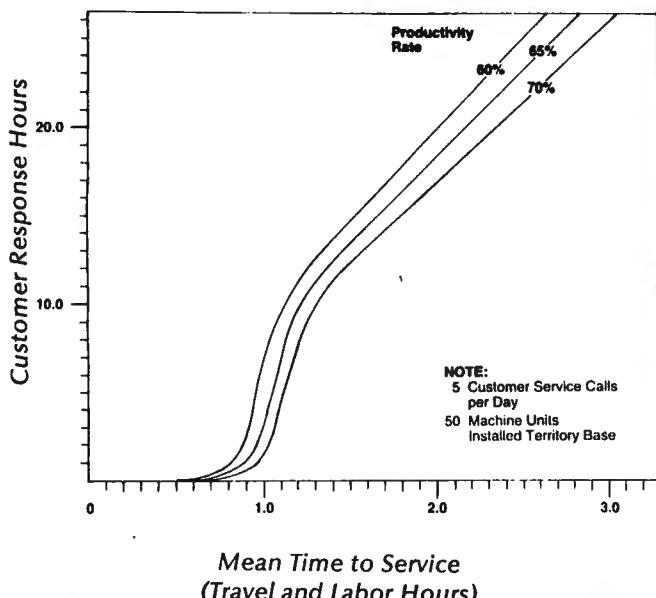
This menu option displays the average response times (in hours) to customer service calls as depicted in screen #4. The planned number of service calls required to achieve the calculated response times is also displayed by record.

(See next page.)

(3) Change data file

This menu option allows the user to change, and edit

IMPACT OF MEAN TIME TO SERVICE AND TECHNICIAN'S PRODUCTIVITY ON CUSTOMER SERVICE RESPONSE HOURS



CHANGE ROUTINE					
TERRITORY	EQUIPMENT INSTALLED BASE	CUSTOMER SVC CALLS / DAY	MEAN TIME TO SERVICE (hours)	SERVICE PRODUCTIVITY LEVEL	
1 MOUNTAIN	50	5	1	65	
2 STREAM	65	5	1	70	
3 RIVER	60	6	1	65	
4 EAST SIDE	100	8	1.2	70	
5 WEST SIDE	95	7	1.5	70	
6 SOUTH SIDE	80	9	1	65	
7 NORTH SIDE	90	9	1	60	
8 HILL TOP	85	7	1.2	65	
ANY CHANGES IN DATA?					

Screen #5

data entered during the input routine.

(Screen #5)

(4) Load existing data file

This menu option prompts the user to load an existing dat file from disk.

PROGRAM LISTING

```

10 REM RESPONSE TIME CALCULATOR
20 REM DENNIS AVOLA D December 4, 1982
50 CLEAR 1000:DIM A$(100),A(4,100)
60 F=1:X=1:W=1
70 U$="# %           % #####     #####.#     #####.#"
72 U1$="#" %           % #####     #####.#     #####.#"
80 REM MENU
82 CLS:PRINTTAB(7)" RESPONSE TIME CALCULATOR"
85 PRINT STRING$(63,"-")
87 PRINT:PRINTTAB(21)***** M E N U *****"
90 PRINT:PRINT TAB(19)"(1) BUILD NEW DATA FILE"
92 PRINT TAB(19)"(2) DISPLAY RESPONSE TIMES"
94 PRINT TAB(19)"(3) CHANGE DATA FILE"
98 PRINT TAB(19)"(4) LOAD EXISTING DATA FILE"
100 PRINT:PRINT TAB(19)"YOUR SELECTION ";:INPUT Z:IF Z<1 OR
Z>4THEN 82
101 ON Z GOTO 105,305,700,850
105 CLS:PRINTTAB(87)" RESPONSE TIME CALCULATOR"
110 PRINT STRING$(63,"=")
120 PRINT:PRINTTAB(18)"PROGRAM FORMAT"
130 PRINT:PRINT TAB(13)"ENTER GROUP LABEL (15 char max)":;
INPUT L$
132 L$=LEFT$(L$,15)
135 PRINT TAB(13)"HOW MANY ";L$;"($)...";:PRINT@428,::INPUT L
138 PRINT TAB(13)"DISPLAY VIA (C)RT OR (P)RINTER ";:INPUT Y$:
IF Y$<>"C" AND Y$<>"P" THEN 138
140 PRINT TAB(13)"ENTER TIME PERIOD":PRINT TAB(11)"(D)ays,
(W)eeks, (M)onths, (Y)ears":PRINT@556,::INPUT T$:
145 IF T$="D" THEN T$="DAY":GOTO 200
148 IF T$="W" THEN T$="WEEK":GOTO 200
150 IF T$="M" THEN T$="MONTH":GOTO 200

```

Screen #3

RESPONSE TIME CALCULATOR				
DISPLAY ROUTINE				
TERRITORY	EQUIPMENT INSTALLED BASE	CUSTOMER SVC CALLS / DAY	LOADING SVC CALLS / DAY	RESPONSE TIME (hours)
1 MOUNTAIN	50	5.0	5.2	3.2
2 STREAM	65	5.0	5.5	1.5
3 RIVER	60	5.0	5.4	3.6
4 EAST SIDE	100	8.0	4.7	20.9
5 WEST SIDE	95	7.0	3.1	24.9
6 SOUTH SIDE	80	9.0	5.2	14.9
7 NORTH SIDE	90	8.0	4.3	18.2
8 HILL TOP	85	7.0	4.3	19.0

Screen #4

```

155 IF T$="Y" THEN T$="YEAR":GOTO 200
157 GOTO 140
200 CLS:PRINTTAB(7)" R E S P O N S E   T I M E   C A L C U
L A T O R"
210 PRINT STRING$(25,"-");"INPUT ROUTINE";:PRINT
STRING$(25,"-")
212 GOSUB 215:IF X=1 THEN 240 ELSE 250
215 PRINTTAB(17)"EQUIPMENT"; TAB(28)"CUSTOMER";TAB(39)"MEAN
TIME";TAB(53)"SERVICE"
220 PRINT TAB(17)"INSTALLED";TAB(28)"SVC CALLS";TAB(39)"TO
SERVICE";TAB(51)"PRODUCTIVITY"
225 PRINTTAB(0) "#";TAB(3)L$;TAB(19)"BASE";TAB(28)"/ " ; T$;
TAB(39)" (hours)";TAB(53)" LEVEL"
230 PRINT STRING$(63,"-")
235 RETURN
240 IF L<8 THEN J=L ELSE J=8
250 P=@:FOR I=F TO J
255 P=P+1
260 PRINT@((384+((P-1)*64)),I:; PRINT@((388+((P-1)*64))-1),;:
INPUT A$(I)
265 PRINT@((402+((P-1)*64))-1),;:INPUT A(1,I)
270 PRINT@((413+((P-1)*64))-1),;:INPUT A(2,I)
275 PRINT@((425+((P-1)*64))-1),;:INPUT A(3,I)
280 PRINT@((438+((P-1)*64))-1),;:INPUT A(4,I)
290 NEXT I
300 GOTO 505
301 PRINT@896,STRING$(63," "):PRINT@896,"SAVE DATA FILE ON
DISK (Y/N)":;INPUT DF$:IF DF$<>"N"ANDDF$<>"Y" THEN 301
302 IF DF$="Y" THEN 900 ELSE 82
304 GOTO 82
305 IF Y$="P" THEN 360
310 CLS:PRINT TAB(9)"R E S P O N S E   T I M E   C A L C U
L A T O R":PRINT STRING$(24,"-");"DISPLAY ROUTINE"::
PRINT STRING$(24,"-")
320 PRINT TAB(20)"EQUIPMENT";TAB(31)"CUSTOMER";TAB(42)""
PLANNED ";TAB(53)"RESPONSE"
330 PRINT TAB(20)"INSTALLED";TAB(31)"SVC CALLS";TAB(42)"SVC
CALLS";TAB(55)"TIME"
340 PRINT"#";TAB(3)L$;TAB(22)"BASE";TAB(31)"/ " ;T$;TAB(42)"/
";T$;TAB(54)"(hours)"
350 PRINT STRING$(63," "): GOTO 400
360 CLS:LPRINTTAB(18)"R E S P O N S E   T I M E   C A L C U
L A T O R"
365 LPRINT STRING$(80,"-")
370 LPRINT TAB(20)"EQUIPMENT";TAB(31)"CUSTOMER";TAB(42);
" PLANNED ";TAB(53)"RESPONSE";TAB(65)"SERVICE"
375 LPRINT TAB(20)"INSTALLED";TAB(31)"SVC CALLS";TAB(42)"SVC
CALLS";TAB(55)"TIME";TAB(63)"PRODUCTIVITY"
380 LPRINTTAB(4)"#";TAB(7)L$;TAB(23)"BASE";TAB(31)"/ " ;T$;
TAB(42)"/ " ;T$;TAB(54)"(hours)";TAB(65)"LEVEL (%)"
390 LPRINT STRING$(80,"-")
400 FOR I = 1 TO L
410 S=1:GOSUB 1000
420 IF Y$="P" THEN 470
430 PRINT USING U$;I;A$(I);A(1,I);A(2,I);MU;WQ
440 GOTO 480
470 LPRINT USING U1$;I;A$(I);A(1,I);A(2,I);MU;WQ;A(4,I)
480 NEXT I
490 FOR Z=1TO1000:NEXTZ:GOTO 82
500 CLS
505 CLS:PRINT TAB(20)"C H A N G E   R O U T I N E"
510 PRINT STRING$(63,"=")
520 GOSUB 215
525 FOR K= F TO J
530 PRINT K;TAB(4);A$(K);TAB(19);A(1,K);TAB(30);A(2,K);
TAB(42);A(3,K);TAB(55);A(4,K)

```

```

540 NEXT K
550 PRINT@896,"ANY CHANGES (Y/N)":;INPUT G$:IF G$<>"Y"
AND G$<>"N" THEN 550
560 IF G$="N" THEN 599
565 PRINT@896,"CHANGE ";L$;" # ";:INPUT E
570 PRINT@896,"CHANGE COLUMN NUMBER.....";:INPUT T
575 IF T=1 THEN 590
580 PRINT@920,"CHANGE TO ";:INPUT R6
585 A((T-1),E)=R6:GOTO 505
590 PRINT@920,"CHANGE TO ";:INPUT R$
595 A$(E)=R$:GOTO 505
599 IF J=L THEN 301
600 IF L>(8*X) AND L<=(J+8) THEN F=(8*X)+1:J=L:X=X+1:GOTO 200
605 X=X+1:F=F+8:J=J+8:GOTO 200
700 F=1:X=1
705 CLS:PRINT TAB(7)" R E S P O N S E   T I M E   C A L C U
L A T O R":PRINT STRING$(63,"-")
710 GOSUB 215:IF X8=1 THEN 715 ELSE 720
715 IF L<8 THEN J=L ELSE J=8
720 P=@:FOR I=F TO J
725 P=P+1
730 PRINTI;TAB(4);A$(I);TAB(19);A(1,I);TAB(30);A(2,I);
TAB(42);A(3,I);TAB(55);A(4,I)
735 NEXT I
740 PRINT@896,"ANY CHANGES (Y/N)":;INPUT G1$:IF G1$<>"Y"
AND G1$<>"N" THEN 740
750 IF G1$="N" THEN 825
760 PRINT@896,"CHANGE ";L$;" # ";:INPUT E3
770 PRINT@896,"CHANGE COLUMN NUMBER.....";:INPUT T6
780 IF T6=1 THEN 800
785 PRINT@920,"CHANGE TO ";:INPUT R5
790 A((T6-1),E3)=R5:GOTO 705
800 PRINT@920,"CHANGE TO ";:INPUT R5$
810 A$(E3)=R5$:GOTO 705
825 IF J=L THEN 82
830 IF L>(8*X8) AND L<=(J+8) THEN F=(8*X8)+1:J=L:X8=X8+1:
GOTO 705
835 X8=X8+1:F=F+8:J=J+8:GOTO 705
850 CLS:PRINT"POSITION DISKETTE AND PRESS ENTER":;INPUT I9
860 PRINT"ENTER NAME OF FILE ";:INPUT L$
870 OPEN"1",1,L$
880 INPUT#1,T$,L
885 FOR I=1 TO L
890 INPUT#1,A$(I),A(1,I),A(2,I),A(3,I),A(4,I)
893 NEXT I
895 PRINTL$;" LOADED FROM DISKETTE":FOR ZZ=1 TO 500:
NEXT ZZ:CLOSE1:GOTO 82
900 CLS:PRINT"POSITION DISKETTE AND PRESS ENTER":;INPUT I9
910 PRINT"SAVING.... ";L$;".... ON DISK"
920 OPEN"0",1,L$
930 PRINT#1,T$,L
940 FOR I=1 TO L
950 PRINT#1,A$(I);";";A(1,I);A(2,I);A(3,I);A(4,I)
955 NEXT I
960 PRINT:PRINT"SAVE ROUTINE COMPLETE...":FOR ZZ=1 TO 500:
NEXT ZZ:CLOSE1:GOTO 82
1000 REM FINITE QUEUE - INFINITE SOURCE - MULTIPLE SERVERS
1005 ON ERROR GOTO 1250
1010 M=A(1,I):T=A(2,I)
1011 GOSUB 3000
1015 MU=((A(4,I)/100)*2000)/A(3,I))/JX
1020 X5=@:FOR N=@ TO S
1025 IF N=@ OR N=1 THEN R=@:GOTO 1040
1030 IF N=2 THEN R=.3010:GOTO 1040

```

continued on page 40

IT STARTED WITH THE ABACUS

Frank Tymon

It started with the abacus. Or perhaps, the soroban. Regardless, the crazy world of computers is upon us. From Pascal's adding machine to Babbage's analytical engine. From Eniac to TI59. From huge room-filling monsters to small pocket-filling computers. Slowly at first, but with ever-increasing acceleration. And now, with the home computer, they have indeed arrived. From arcade games to TV tuners, from microwave ovens to computerized stereos, from morning to night—computers are busily serving us. They control our traffic lights, our auto ignitions, our cooking stoves—and, at times, I suspect my own actions are programmed.

Seriously, there is an ever-widening use of computers. And you will eventually come in contact with them? Perhaps you will resist. But, of course, to no avail. Why not, then, profit from them? And one way to profit is to write about them. A broad spectrum of computer-related subjects exists. You need only choose the one on which you want to write. Yes, the computers have arrived.

Not that I'm complaining. There's gold in them there gadgets. Literally, which doesn't really matter. But also figuratively. And that does matter to you, the potential writer.

It matters to you for several reasons. One is that you, like many authors, either now type on, or will soon type on a word processor. Which is merely a specialized computer.

Of great consequence, and the subject of this article, is the explosion of computer-related magazines. The meteoric arrival of computer-related books. The plethora of computer-related articles. And that is where the real gold lies. The problem—how to pan it. Or, rather, in a literary vein, how to pen it.

If you read any of the magazines devoted to the programmer you will detect two interesting characteristics of the authors. Either the author is an excellent author, but knows computers in only a general way—or the author is a technical expert with the literary skill of a, well, technical expert. I exaggerate. But not so very much.

I will admit, however, there are a few very competent writers in the field, and it is a pleasure to read them.

Can you cash in on the growing demand for computer-oriented articles, books, and stories? I think so. Let's examine how.

For starters, perhaps you might want to change fields and become a programmer. Sorry! Actually, I don't really think that is necessary. However, you would be wise to learn the jargon, the buzz words, if you plan to write on the subject. There are an extended lot of them. And in this dynamic field they are sprouting like weeds.

So you are up to speed on technical terms. What now? Why not look at the market as it exists today? What is being written? Where is it being published? What are the characteristics? To find out, read some of the numerous magazines on the market. Such magazines as *BYTE*, *Computronics*, *80 Micro*, *On Computing*, *Kilobaud*, etc. And these are but a few of the burgeoning set of computer-oriented magazines available. And new ones are surfacing at a rapid rate.

Another indicator of the increasing interest in

computers and computer programming is the count of computer books in the bookstore. At one time, if you looked at the technical book rack, at most there were only one or two treating computers. The majority of the texts were electronics related. Today the opposite situation prevails. Not only are the computer books dominant, but they now occupy much more space in the store than did the electronics volumes of a few years back. Yes, the computers are taking over. Taking over the bookshelves!

So, in addition to reading a few representative magazines, browse through some of the volumes. Get a feel for the microprocessor and its progeny. Learn what a program is, even how to write a simple one. (It really doesn't hurt!) Now, when you feel comfortable with this new world, think of how you can contribute.

I think you have recognized the gamut of computer articles. At one end of the spectrum you read the "Gee Whiz!" products that tell us how, tomorrow, we'll be the servants of a computerized set of tin cans. And, somewhere near the other end of the spectrum, you find that every action of the computer must be carefully and specifically programmed. (This is usually illustrated by an outpouring of cryptic assembly language code!)

A phenomenon to watch is that of change. At first the computer world was one of building computers. Then it became the world of BASIC language books. BASIC game books arrived. Next, "How to Do It in Assembly Language." Today the growing areas are two-fold—word processing and robotics.

These trends are important to you, if you propose to write about computers. It is here on the leading edge that new articles, new books, are most needed. And you need not know robotics nor computers to write, for example, "What Robotics Wrought!" Just so long as "What Robotics Wrought!" does indeed address the concept and introduce new and different points of view. It isn't all technology. In fact, the human interface with the mechanical world has always been an area of primary interest. And the computer doesn't change that.

To write the computer and programming type articles, then, the writer must either become technically expert and engineer the article; or, learn at least the fundamentals of the language, but write on the more humanistic side of the subject. There is a tremendous and growing market for both.

To illustrate, selecting 2 volumes of a major computer magazine, here is a rough break out of the article distribution. Technical articles constitute 33 of 42. Humanistic articles, 9 of 42. Percentage-wise, the latter make up nearly 25 percent of the articles. This may not seem so large, unless you consider that these are technically oriented magazines.

Perhaps you are skilled in logical thinking. Then you may offer the computer world a new and useful algorithm. An article explaining how to arithmetically perform a function of interest. Or how to generate a sequence of needed numbers in an efficient way. On the other hand, maybe you aren't really cut out for that approach.

continued on page 53

POCKET COMPUTER CORNER

Steven M. Zimmerman, Ph.D. and Leo M. Conrad
Finding the Effective After Tax Yield on Investments

Copyright[®] Zimmerman and Conrad 1983

Even before the tax law change in December 1982, requiring the deduction of taxes from saving accounts, the problem of finding the effective after tax yield on investments was difficult for many investors. Selecting between investments when the interest is stated in different words, when the compounding period is different, and when some investments are taxed and others are not, can be very troublesome. This program for the pocket computer makes the task easy.

What makes the computation task difficult for an individual, even one working with one of today's powerful calculators, is the law requires a tax of 10% be collected each quarter and then requires the full tax be paid at year's end. The exact operation of the law has yet to be worked out by the banks and other thrift institutions and tested in the courts. In this program it is assumed a tax will be collected quarterly only, on those saving accounts which compound interest payments quarterly or more frequently, and pay interest to the saver on a quarterly basis or more frequently. For those cases where interest is compounded and paid bi-annually or annually, the tax is collected only when the interest is paid.

To protect against future changes in the law the annual tax rate and the quarterly collection rate have been made input data items. If the law is changed to require a different collection percentage, the program will still be useful. If the law is changed back such that no tax need be collected all you need do is input a zero when asked what tax is collected quarterly.

WARNING

Be sure to check the assumptions made in the programs against the manner in which the institution has decided to withhold taxes from your savings before using the program. Check your newspaper for changes in the law. There is currently a big effort to have the law changed. As you can see by examining the calculation procedures, the collection of tax on saving accounts places a big burden individuals such as retired persons who are depending on savings. The question of the fairness of the law has yet to be determined.

SOME COMPOUND INTEREST THEORY

Most yields are stated as an annual amount. This amount is called nominal interest. To find the interest per compounding period the nominal rate is divided by the number of compounding periods. The federal government defined this value as the annual percentage rate or APR when a loan is being made by a lending institution to an individual. We have selected the term APR to be used as a prompt in the program. There is only a limited amount of characters on the display and short-hand methods of asking questions are needed.

An interest rate adjusted for the compounding period is called the effective rate. The effective rate assumes the interest earned in all but the last period earns interest. This concept is the foundation of compound

interest. For example, if an investment earns 6% APR nominal interest and is compounded every six months, the loan value at the end of six months is the original value plus one half the annual interest rate times the original principal.

Starting with the loan value at the end of six months, the value at the end of the year is calculated as the loan at the end of six months value plus one half the annual interest rate times the loan value at the end of six months. This sounds complex and it is difficult to state, but it is very easy for your computer to calculate.

TAX PAYMENT and WITHDRAW of FUNDS

In the program it is assumed the taxes are taken out each quarter if the interest is posted, paid to the account. In addition it is assumed you do not withdraw any funds from the account during the year for any reason.

To illustrate the potential results of the "New Tax" program a series of runs were made for an annual percentage rate of 10%. The results are shown below:

Tax Free investments

Annual	Bi-annually	Quarterly	Monthly	Daily
10%	10.25%	10.3812841%	10.4713067%	10.515578%

50% Tax bracket and 10% withholding per quarter

Annual	Bi-annually	Quarterly	Monthly	Daily
5%	5.1125%	5.1712954%	5.2159744%	5.237947%

50% Tax bracket and 50% withholding per quarter

Annual	Bi-annually	Quarterly	Monthly	Daily
5%	5.0625%	5.0945337%	5.1379094%	5.159237%

In the case of interest calculated and paid once a year, the withholding of earned interest does not have an effect. It is assumed that if interest is not earned and not posted to a person's account, no tax can be collected. Take care to check this assumption. This interest paying plan does not have any compounding effect and results in a lower effective interest, even though no taxes are deducted except at the end of the year.

Tax free results are always greater than the taxed results. If quarterly, a tax of 10% is collected and the effective after tax interest is greater than if a 50% tax is collected each quarter. The reason for this is you lose the interest earning ability of your money when the higher tax is paid.

The important fact this program tells us is that the amount of the interest available to earn additional interest is a function of how taxes are collected. If the tax is collected in the period the interest is earned, the interest earning ability of the interest is reduced greatly. The government gets the use of the money faster and in reality, this results in collection procedures serving as a tax increase.

RUNNING THE PROGRAM

The program may be operated in either the DEFInable or RUN modes. Type R. <CR>, (<CR> means to press the white ENTER key- carriage return) or RUN <CR> and you will see:

INVESTMENT YIELD
TAX RATE?

Let's assume you are in the 40% tax bracket. Type 40 <CR> and you will see:

QUARTER TAX RATE?

This question asks for the tax deduction rate per quarter. It is currently 10%. If your tax rate is less than the quarterly tax deduction rate the program will work and produce results. There is some question of exactly what the results mean relative to the tax refund you should receive from the government. Examine this aspect of your own situation before using the computer results.

To continue type 10 <CR> on the assumption of a 10% deduction. The next question is:

INVESTMENT/END?

This question allows you to name each one of your investments. Names cannot be greater than 7 characters on your pocket computer. Use the name Bank 1. Type BANK 1 <CR> to continue to the next question:

APR?

As noted earlier this question refers to the stated annual interest. Let's use an APR of 10%. Type 10 <CR> to continue to the next question:

AN,BI,QU,MO,DA?

This question is asking you to identify the compounding period, ANnual, or BI-annually, or QUarterly, or MOnthly, or DAily. You must answer using the exact two letter shown above. There are no error traps built into this program so be very careful in your data entry. Let's use a MOnthly compounding period. Type MO <CR> to continue to the next question:

TAX FREE(Y/N)?

Investments which are tax free have results that vary greatly from those that are not. Assume Bank 1 has a tax free investment and Bank 2 has one with the same characteristics which is taxed. This type type Y <CR>. The next question is:

INVESTMENT/END?

Answer BANK 2 <CR> and then answer all of the following questions as shown:

APR? 10 <CR>
AN,BI,QU,MO,DA? MO <CR>
TAX FREE(Y/N)? N <CR>
INVESTMENT/END? END <CR>

Since you answered END <CR> the input phase of the program is terminated and on your display or printer will be produced the following:

TAX RATE: 40.
QUARTER RATE: 10.
***** 1. *****
INVESTMENT: BANK
1
APR: 10.

Results designed to match
Printer format

COMPOUNDING: MO
TAX FREE: Y
AFTER TAX: 10.47
13067 %
***** 2. *****
INVESTMENT: BANK
2
APR: 10.
COMPOUNDING: MO
TAX FREE: Y
AFTER TAX: 6.259
1693 %

Compare the above results with the table for the 50% tax rate. The computer has the ability to handle any tax level with ease. You, the user, need only be concerned with the results and their meaning relative to your financial plan.

The program is not designed to recycle. You must type R. <CR> for another run.

EXAMINING THE PROGRAM

The program is structured by line numbers. Lines 1-7 are used to input the data. Lines 11-41 perform the calculations and print the results on the display and Printer.

The variables used are as noted below:

Variable	Use
A	Tax bracket
B	Counter
C	Counter
D	Counter
E	Compounding period
F	Effective after tax yield
G	Actual tax rate
H	Quarter tax rate paid
I	Actual quarter tax rate paid
J A(11)	First quarter earnings
K A(12)	Second quarter earnings
L A(13)	Third quarter earnings
M A(14)	Fourth quarter earnings
N	Counter
A(16)...	To limit of memory data base

Lines 11-15 print the input data, line 16 sets the value of I and G, depending upon whether the investment is tax free or not. Lines 20-24 are designed to route the program depending on the compounding period. If the period is annual line 20 operates and the program goes to line 39. If the period is bi-annually then line 21 sends the program to line 38 for the calculations and then to line 39. Otherwise lines 22-24 set the value of E and the program is routed to line 30 where the calculations are performed.

Examination of this routing code reveals no error traps. Such traps were not included to increase the capacity of the system to handle data. This is a trade-off one is often faced with when using the pocket computer.

The calculations for the quarterly, monthly, and daily options are performed in lines 30-35. After this task is complete the program is sent to line 39.

Line 39 sets the output as a percent rather than a decimal. Line 40 prints the results and line 41 indexes

to the next value of D. If you wish to add a recycling statement you may do so by add a GOTO 1 in line 42. This will use memory and reduce the capacity of the system.

In lines 30-32 the letter "O" is used. Take special care not to type the number zero! Also in the calculations the practice of tying .01A rather than .001 * A has been followed to reduce the amount of memory used.

PROGRAM LISTING

```

1:PAUSE "INVESTMENTS YIELD"
2:INPUT "TAX RATE?";A:INPUT "QUARTER TAX RATE?";H
3:B=16:C=1
4:INPUT "INVESTMENT/END?";A$(B):IF A$(B)="END" THEN 10
5:B=B+1:INPUT "APR?";A(B):B=B+1:INPUT "AN,BI,QU,MO,DA?";A$(B)
6:B=B+1:INPUT "TAX FREE(Y/N)?";A$(B):B=B+1
7:C=C+1:GOTO 4

10:C=C-1:PRINT "TAX RATE: ";A:PRINT "QUARTER RATE: ";H
11:FOR D=1 TO C:PRINT "***** ";D;" *****":B=4*(D-1)+16
12:PRINT "INVESTMENT: ";A$(B):B=B+1
13:PRINT "APR: ";A(B):B=B+1
14:PRINT "COMPOUNDING: ";A$(B):B=B+1
15:PRINT "TAX FREE: ";A$(B)
16:G=A:I=H:IF A$(B)="Y" LET G=0:I=0

20:IF A$(B-1)="AN" LET F=(1+.01A(B-2)-1)*(1-.01G):GOTO 39
21:IF A$(B-1)="BI" THEN 38

22:IF A$(B-1)="QU" LET E=4
23:IF A$(B-1)="MO" LET E=12
24:IF A$(B-1)="DA" LET E=365

30:0=1:FOR N=1 TO 4
31:F=0*(1+.001A(B-2)/E)^(E/4)      tis the power key on your
32:A(9+N)=F-O:F=F-(F-O)*.01I:0=F   pocket computer
33:NEXT N
34:F=F-(J+K+L+M)*(01G-.01I)
35:F=F-1:GOTO 39

```

38:F=1+.005A(B-2)*(1-.01I):F=F+F*.005A(B-2)*(1-.01G)-.005A(B-2)*(.01G-.001I)-1

39:F=100F

40:PRINT "AFTER TAX: ";F;" %"

41:NEXT D

Steven M. Zimmerman, Ph.D.

College of Business and Management Studies

University of South Alabama

Mobile, Alabama

Leo M. Conrad

Imagineering Concepts

P.O. Box 9843

Mobile, Alabama 36691-0843 ■

PROGRAM PREVIEWS

continued from page 15

As with the manual for "Infoscan" (previously reviewed), there are a number of spelling errors that should be corrected in later releases. I do not feel that I am being too harsh in my review of this manual—I would just like to see a better-than-average program, such as Screen Artist II, have good documentation. Overall, the manual rates a 6 on the usual scale of 1 to 10.

There are a number of interesting sample display programs on the Screen Artist II disk. However, neither the manual nor a July 1982 User Update sheet provides the names of these programs, although the latter mentions their presence. It was necessary to call for a "DIR" in System mode in order to determine the file names to request for the demonstrations.

SUMMARY EVALUATION

When I first approached Screen Artist II to review it, I was fairly ho-hum about it—just another graphics program. As I delved further into its operation, my attitude changed completely; here was a program that was, quite literally, interesting and exciting. I found myself reluctant to stop enjoying its creative possibilities.

continued on next page

REAL TIME CLOCK TRS-80 & APPLE

Now available from JKR ENGINEERING, versatile clock/calendars for the TRS-80 and APPLE computers.

Only for the "believable" price of **\$69** each.

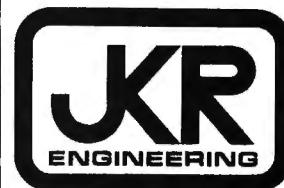
TRS-80 I,II,& IV	APPLE II+ & IIe	CLOCK FEATURES
—	•	1K of user available non-volatile RAM.
•	•	Loads time and date memory vectors.
Beeper	Apple spkr	Alarm: Sound and/or vector jump.
•	—	Operates in TRSDOS and LDOS.
•	—	Expansion connector provided.
—	•	Display relocatable on screen.
—	•	Time/Date/Day in 4 formats.
•	•	Software interrupt enable.
•	•	Rechargeable battery
—	•	Slot independent.
TIMES\$ = Time	AL\$ = Alarm	Add \$2 for shipping & handling per order.
DATE\$ = Date	TI\$ = Time	Add \$3 for COD orders. CA res add $6\frac{1}{2}\%$ Sales Tax.
	Assigns Basic variables	

Also available from JKR ENGINEERING:

IEEE-488 to TRS-80 Interface.

Only **\$299.**

(See the May issue of Computronics for details)



1687 Pinewood Way
Milpitas CA 95035

To order call
(408) 263-7139

Add \$2 for shipping & handling per order.

Add \$3 for COD orders. CA res add $6\frac{1}{2}\%$ Sales Tax.

The program operates as advertised, no flaws that I could detect, and its machine language speed is a pleasure to see in operation. The only restriction is the limitation on printer use. With the multitude of commands and controls available, the full graphics capabilities of the TRS-80 may be reached to the fullest extent. Extremely simple to operate, once you become familiar with all of those commands, Screen Artist II requires only very basic programming skills in order to enjoy its full effect. This is a program that I have no hesitancy in recommending for those who may be seeking to put some "pizazz" into their programs.

Screen Artist II—The Smallsystem Center, Box 268, New Hartford, CT 06007—Model I, III, requires 32K and one Disk Drive minimum. \$32.95, shipping \$2.50.

A. A. Wicks
30646 Rigger Road
Agoura, CA 91301 ■

PRACTICAL BUSINESS PROGRAMS

continued from page 26

```

9999 THEN I=I-1: GOTO 70
50 INPUT "ENDING MILES";EM(I): D(I)=SM(I)-EM(I):
D(I)=ABS(D(I)): SU=SU+D(I)
60 PRINT D(I): GOTO 40
70 PRINT "SUM=",SU
80 PRINT "MILES MENU": PRINT "      A    ADD MORE VALUES":
PRINT "      C    CHANGE VALUES"
90 PRINT"      M    CALCULATE MILEAGE": PRINT "      P
      PRINT RESULTS ON CRT"
100 PRINT"     LP    PRINT RESULTS ON LINE PRINTER": INPUT
"      R    RETURN TO GLMENU";K$
110 X$="## ##### ###### ##### ###### #####"
120 IF K$="A" THEN 40
130 IF K$="R" THEN LOAD"GLMENU",R
140 IF K$<"C" THEN 180
150 INPUT "NUMBER TO BE CHANGED, NEW STARTING MILES, NEW
ENDING MILES";J,SM(J),EM(J)
160 D(J)=SM(J)-EM(J):D(J)=ABS(D(J))
170 SU=0: FOR J=1 TO I: SU=SU+D(I): NEXT J: GOTO 80
180 IF K$="LP" THEN 240
190 IF K$<"P" THEN 280
200 K=0:SU=0: GOSUB 210 : GOTO 220
210 PRINT "NO    STARTING MILES    ENDING MILES    DIFFERENCE
SUM": RETURN
220 FOR J=1 TO I: SU=SU+D(J): PRINT USING X$;J,SM(J),EM(J),
D(J),SU: K=K+1: IF K=14 THEN K=0: INPUT "ENTER TO PAGE";DU$:
GOSUB 210
230 NEXT J: INPUT "ENTER TO CONTINUE";DU$: GOTO 80
240 INPUT"LINE COUNTER (Y/N)";LC$: IF LC$="Y" THEN CMD
"FORMS(T)"
250 SU=0: INPUT "TITLE, DATE";T$,D$: LPRINT T$,D$: LPRINT
"NO    STARTING MILES    ENDING MILES    DIFFERENCE
SUM"
260 FOR J=1 TO I: SU=SU+D(J): LPRINT USING X$;J,SM(J),EM(J),
D(J),SU: NEXT J
270 INPUT "CHARGE PER MILE";CC#:DD#=CC#*SU: LPRINT "  ";
LPRINT "COST PER MILE IS ";CC#;" TOTAL COST IS ";DD#: GOTO 80
280 IF K$<"M" THEN 80
290 INPUT "CHARGE PER MILE";CH#:DD#=CH#*SU:PRINT "COST IS
";DD#: GOTO 80

```

PROGRAM LISTING: MOVE

```

10 CLEAR 1000: CLS : REM  "MOVE"
20 CLS : PRINT "THIS PROGRAM MOVES FILES TO NEW DISKS":
INPUT "MAXIMUM NUMBER OF TRANSACTIONS";D
30 DIM A%(D,4),A#(D,2),B#(D,2):REM  LIMIT 250 FILES BY
DIMENSION STATEMENT
40 PRINT: PRINT "THIS PROGRAM DOES NOT DESTROY OLD FILES IT
ONLY CREATES NEW FILES *** USE KILL TO REMOVE UNWANTED FILES":
PRINT
50 PRINT "MENU":PRINT"      OB    MOVE OLD BALANCE/P&L":
PRINT"      R    RETURN TO GLMENU": INPUT"      TR    MOVE
TRANSACTION FILES";H$
60 IF H$="OB" THEN 190
70 IF H$="R" THEN LOAD"GLMENU",R
80 IF H$<>"TR" THEN 50
90 INPUT "NAME OF OLD CHECK FILE";OC$: INPUT "DISK";DD$:
OC$=OC$+": "+DD$
100 OPEN "I",1,OC$:I=0
110 I=I+1: INPUT #1,A%(I,1),A%(I,2),A%(I,3),A%(I,4),
A#(I,1),A#(I,2)
120 IF A%(I,1)>0 THEN 110
130 CLOSE 1: PRINT "THE FILE IS NOW IN MEMORY": INPUT "SETUP
NEW DISKS IN DISK DRIVES ENTER TO CONTINUE";DU$
140 INPUT "NAME TO BE USED FOR CHECK FILE";NC$: INPUT
"DISK";DD$: NC$=NC$+": "+DD$
150 OPEN "O",2,NC$:I=0
160 I=I+1: PRINT #2,A%(I,1),A%(I,2),A%(I,3),A%(I,4),A#(I,1),
A#(I,2)
170 IF A%(I,1)>0 THEN 160
180 CLOSE 2: GOTO 50
190 INPUT "NAME OF OLD BALANCE SHEET FILE";OB$: INPUT "DISK";
DD$: OP$=OB$+"P:"+DD$:OB$=OB$+": "+DD$
200 OPEN "I",1,"NO":SU=0: FOR I=1 TO 6: INPUT #1,J(I):
SU=SU+J(I): NEXT I: CLOSE 1
210 OPEN "I",1,OB$: FOR I=1 TO SU: INPUT #1,A#(I,1),A#(I,2):
NEXT I: CLOSE 1
220 OPEN "I",1,OP$: FOR I=J(1)+J(2)+1 TO J(1)+J(2)+J(3)+J(4):
INPUT #1,B#(I,1),B#(I,2): NEXT I: CLOSE 1
230 PRINT "THE FILE IS NOW IN MEMORY": INPUT "SETUP NEW DISKS
IN DISK DRIVES ENTER TO CONTINUE";DU$
240 INPUT "NAME TO BE USE FOR OLD BALANCE";NB$: INPUT "DISK";
DD$: NP$=NB$+": "+DD$: NB$=NB$+": "+DD$
250 OPEN "O",1,NB$: FOR I=1 TO SU: PRINT #1,A#(I,1),A#(I,2):
NEXT I: CLOSE 1
260 OPEN "O",1,np$:FOR I=J(1)+J(2)+1 TO J(1)+J(2)+J(3)+J(4):
PRINT #1,B#(I,1),B#(I,2): NEXT I: CLOSE 1
270 GOTO 50

```

SUMMARY

These two programs complete our general ledger package. The package is designed for the small business person who needs a general ledger system which works efficiently. We hope it will prove useful to you in the operation of your business.

Steven M. Zimmerman, Ph.D.
College of Business and Management Studies
University of South Alabama
Mobile, Alabama 36688

Leo M. Conrad
Imagineering Concepts
P.O. Box 9843
Mobile, Alabama 36691-0843 ■

XSNORF: THE NEW, EXCITING WORD GAME FOR THE POCKET COMPUTER

Anthony T. Scarpelli and Leigh T. Whittemore

This game was not meant to be for kids. In fact, I envisioned it as a game played by a number of creative individuals sitting at a bar with a few drinks under their belt. After a round or two, someone pulls out his trusty Pocket Computer and starts the game going.

The idea of the game is to create definitions for a number of random words produced by the computer. Each person inventing the definition is judged by the others, and the winner of the round is given seven points. The next best is given three points, and the next, one point. After the game, the computer tallies the points and the loser buys the next round of drinks. It could get very funny.

The following definitions have been created by people who were sober at the time:

JIVETA - a new car
IRRAXY - new diagnostic X-ray
EDATIL - new drug
OCNEA - Jacques Cousteau's new underwater world
IFATTP - new method for losing weight with toilet paper
NEEITF - sneeze sound made by orthopedic surgeon
PEOKU - new computer term from the inventors of "PEEK" and "POKE"
OUTEWI - Indian tribe
RMAADO - new galaxy
RHORNI - country in S. Africa
RORTEU - new store opening in Portland to compete with Porteous
ICMOKS - Russian word meaning "cold"
EZEHUA - hero in Old Testament
SULRRA - produced by rotten Tasmanian devil(ed) eggs
ESKEOF - Eskimo kiss
UEEWIE - exclamation from E.T.
IUBEEE - birth control device for bees
TWIZAF - new game from Milton-Bradlee
OYZEMA - Jewish word for grand-mother
EETOON - division of soldiers (1-1/2 men)
RUPTED - what Mt. St. Helens did

As you can see some of the definitions can be very funny. They could become funnier as the night goes on, too.

The program started when I borrowed a PC-1 Pocket Computer to see what it could do. One of the ideas I had was to see whether it could generate random words to put on a car license. I soon found it could, and when I showed a list of the words to Leigh, he turned it into a game.

The letters in the English language are used with varying frequency. For instance the letter "e" is used with a frequency of 13%, the letter "t," 9.5%, the letter "a," 8% and so on. In order for random words to be as similar to English words as possible, we would want the same frequency to exist in the Pocket Computer's program. However, the limited memory of the computer prevents this. Instead we have six e's, four t's and so on which simulates the frequency.

Listing 1. is a list of the data file that is to be loaded

after the program is run. Record this file (labeled "DATA") on your program tape directly following the program recording. The program will then load this directly if it has not already been loaded.

The random nature of the letters in the data was produced by a program written for my Model I. It insured that the letters were as random as possible. Added to the random numbers generated by the program, the words generated are pretty close to being truly random.

Now let's look at the program in listing 2, which is designed to be used with the Printer/Cassette Interface.

In line 5, the first line, we see the label for the program "X." With this label we can start the program by pressing SHFT X while in the DEF mode. The title of the program is printed and you are asked if the data has been loaded. Line 10 will load the data for you, as long as you are still connected to your cassette recorder, and then the rules are printed. You will then be asked for a seed number for the random number generator in line 35 which also initializes the round number. In any further rounds of play, this part is skipped.

In line 50 we input the number of players, from two to six, and then how many rounds you want to play. Line 60 inputs the names of the players.

Line 65 sets each player's initial score to zero.

Line 70 prints the round number, and then jumps to line 200 to generate the words.

Line 200 starts the loop to print the ten words, and is also the random number generator. What it does is to generate a number "I," which, when added to 26, points to a letter in the A\$ data array. This letter is then put into A\$(10) through A\$(15). The entire word is then printed in line 210 by concatenating each of the array variables. A "BEEP" is inserted in the line so that you will know that something is going on. It takes a while to generate the words.

Line 215 prints some asterisks and jumps us back to line 75 which prints the instructions "SHFT S TO SCORE." Then in line 80 we have the label "S." This label allows you to score the round more easily if you are laughing too hard. Just put the computer in the DEF mode and type "S."

Line 80 also starts the loop that takes us through each word. Lines 85 through 100 asks for the winners by name. Line 100 is skipped in case there are only two people playing. Each line uses a subroutine in line 150 that adds up the scores for that winner. If you enter a name that can't be found, line 160 asks for the name again. (Entering names gets more difficult as the night goes on.)

Finally line 110 sends us to line 220 which prints the round's winning names and points, and then checks to see if there are any more rounds to go.

Listing three is an example of what a printout of a game looks like.

Now you know what you can do with your Pocket Computer when you find yourself at a bar and you or your friends are at a loss for words.

Listing 1. Random letters for string data.

XSNORF WORD DATA

```

A$( 27) = H
A$( 28) = I
A$( 29) = I
A$( 30) = W
A$( 31) = S
A$( 32) = P
A$( 33) = I
A$( 34) = K
A$( 35) = R
A$( 36) = C
A$( 37) = N
A$( 38) = U
A$( 39) = Q
A$( 40) = D
A$( 41) = T
A$( 42) = A
A$( 43) = O
A$( 44) = H
A$( 45) = L
A$( 46) = E
A$( 47) = S
A$( 48) = A
A$( 49) = T
A$( 50) = D
A$( 51) = B
A$( 52) = E
A$( 53) = T
A$( 54) = E
A$( 55) = E
A$( 56) = M
A$( 57) = O
A$( 58) = H
A$( 59) = Z
A$( 60) = G
A$( 61) = A
A$( 62) = X
A$( 63) = E
A$( 64) = R
A$( 65) = J
A$( 66) = N
A$( 67) = N
A$( 68) = V
A$( 69) = I
A$( 70) = A
A$( 71) = T
A$( 72) = O
A$( 73) = F
A$( 74) = E
A$( 75) = S
A$( 76) = R

```

```

PRINT "3RD= 1 POINT
35 F=1 :
INPUT "SEED= ";G
40 PRINT -----
50 INPUT "NUMBER OF PLAYERS= ";B :
IF B>6 PAUSE "TOO MANY" :
GOTO 50
55 INPUT "NUMBER OF ROUNDS= ";C
60 INPUT "FIRST PLAYER NAME? ";A$(16) :
FOR D=17 TO (15+B) :
INPUT "NEXT PLAYER NAME? ";A$(D) :
NEXT D
65 FOR D=21 TO 26 :
A(D)=0 :
NEXT D
70 PRINT USING "####";"ROUND NUMBER";F :
GOTO 200
75 PRINT ">SHFT S TO SCORE *****" :
END
80 "S"
USING "####" :
FOR H=1 TO 10
85 PAUSE "WORD #";H :
INPUT "1ST PLACE WINNER?";A$ :
E=7 :
GOSUB 150
90 INPUT "2ND PLACE WINNER?";A$ :
E=3 :
GOSUB 150
100 IF B>2 INPUT "3RD PLACE WINNER?";A$ :
E=1 :
GOSUB 150 :
110 NEXT H
GOTO 220
150 FOR D=16 TO (15+B) :
IF A$(D)=A$ LET A(5+D)=A(5+D)+E :
D=15+B :
NEXT D :
RETURN
160 NEXT D :
INPUT "TYPE NAME AGAIN? ";A$ :
D=D-1 :
GOTO 150
200 FOR H=1 TO 10 :
FOR D=10 TO 15 :
G=221*G+2113 :
G=G-INT(G/5000)*5000 :
I=INT( (G/100)+1 :
A$(D)=A$(I+26)
210 BEEP 1 :
NEXT D :
PRINT "#";H;" ";A$(10);A$(11);A$(12);
A$(13);A$(14);A$(15) :
NEXT H
215 PRINT "*****" :
GOTO 75
220 USING "####" :
FOR D=16 TO (15+B) :
H=A(D+5) :
PRINT A$(D);":";H;" PTS" :
NEXT D
230 PRINT "...":
F=F+1 :
IF F>C PRINT "GAME OVER!" :
END
240 GOTO 70

```

Listing 2. XSNORF program.

```

5 "X"
PRINT "XSNORF-WORD GAME" :
PRINT "-----" :
INPUT "DATA LOADED? (Y OR N) ";A$
10 IF A$="N" CLEAR :
INPUT# "DATA"
20 PRINT "BEST DEFINITIONS" :
PRINT "OF RANDOM WORDS
30 PRINT "WIN POINTS!" :
PRINT "1ST= 7 POINTS" :
PRINT "2ND= 3 POINTS" :

```

continued on page 38

THE COMPUTER REVOLUTION: WHO ARE THE BENEFICIARIES?

(Name Withheld)

If you are reading this you are probably already one of the beneficiaries. You have discovered the intellectual excitement and satisfaction which the Personal Computer generates and provides.

You have forgone nights at the opera, evenings at the club, trips to the country in Autumn, and, yes, even television, to sit in front of your home computer discovering how it works, getting it to work for you, play with you, help you make a living, etc., etc.

One of the things that my friend Ray hates the most is television. When he is with people who are watching TV they do not want to bother with him, but Ray can't hear the sound coming from the television nor can he see what is happening on the screen, for he is deaf and blind. Ray's sensory world is largely the world of touch, and when Ray happens to touch a TV he quickly turns away from it as holding nothing of promise for him. If this article is successful, Ray's aversion to the TV will be a thing of the past.

I have a wonderful idea on how to turn just about any home computer into a computer for the deaf/blind. And when you understand the simplicity of it I want to challenge YOU to implement it on your home computer, with the result that the whole world of home computers will be opened to the deaf/blind programmer.

Ray has told me that most of the deaf/blind he has met are socially backward, preferring not to communicate as they pass through life almost, as it were, in a stupor. Ray was fortunate to have a teacher, and fortunate to have parents that didn't give up on him and forced him to stay in school even though it would have been a lot easier on them if Ray was just content to sit in a corner for the rest of his life.

I am an Amateur Radio Operator and I met Ray by chance one day many years ago. He surprised me by saying that he had been listening to about twenty other conversations (in morse code) and he decided that he wanted to talk to me. This certainly was an unusual statement. He quickly followed this up by telling me he was deaf and blind. I couldn't figure out how he was "hearing" me, so I asked him to explain. He said he had taken the grille off the loudspeaker, and could feel the vibrations of the cone with his fingers, and that he could copy morse code up to about 50 words per minute, which was a lot faster than I could. I had to apologize for sending so slow.

Well, that was many years ago, and over the years I have met Ray in person many times. Ray is one of the few deaf/blind who is able to speak (like Helen Keller) and to understand your speech. He does this by reading your lips with his thumb and by feeling the vibrations in your throat with his index finger. It is amazing—and so difficult that not many deaf/blind are lucky enough to be able to learn it.

I want to reproduce the Address of Miss Helen Keller for you, because if you take up the challenge of implementing the idea of turning your home computer into a programmable machine for the deaf/blind you will need a lot of staying power, since the hardware,

though simple (a morse code oscillator driven from an output port), must be supported by a program which is somewhat less simple. Here then is the address which Helen Keller delivered to the Association of Lions Clubs at Cedar Point, Ohio, USA, in 1925:

THE ADDRESS OF MISS HELEN KELLER

Dear Lions and Ladies. I suppose you have heard the legend that represents opportunity as a capricious lady, who knocks at every door but once, and if the door isn't opened quickly, she passes on, never to return. And that is as it should be. Lovely, desirable ladies won't wait. You have to go out and grab 'em.

I am your opportunity. I am knocking at your door. I want to be adopted. The legend doesn't say what you are to do when several beautiful opportunities present themselves at the same door. I guess you have to choose the one you love best. I hope you will adopt me. I am the youngest here, and what I offer you is full of splendid opportunities for service.

The American Foundation for the Blind is only four years old. It grew out of the imperative needs of the blind, and was called into existence by the sightless themselves. It is national and international in scope and in importance. It represents the best and most enlightened thought on our subject that has been reached so far. Its object is to make the lives of the blind more worthwhile everywhere by increasing their economic value and giving them the joy of normal activity.

Try to imagine how you would feel if you were suddenly stricken blind today. Picture yourself stumbling and groping at noonday as in the night, your work, your independence, gone. In that dark world, wouldn't you be glad if a friend took you by the hand and said, "Come with me and I will teach you how to do some of the things you used to do when you could see." That is just the kind of friend the American Foundation is going to be to all the blind in this country if seeing people will give it the support it must have.

You have heard how through a little word dropped from the fingers of another, a ray of light from another soul touched the darkness of my mind and I found myself, found the world, found God. It is because my teacher learned about me and broke through the dark, silent imprisonment which held me that I am able to work for myself and for others. It is the caring we want more than money. The gift without the sympathy and interest of the giver is empty. If you care, if we can make the people of this great country care, the blind will indeed triumph over blindness.

The opportunity I bring to you, Lions, is this: To foster and sponsor the work of the American Foundation for the Blind. Will you not help me hasten the day when there shall be no preventable blindness; no little deaf, blind child untaught; no blind man or woman unaided? I appeal to you Lions, you who have your sight, your

continued on page 38

THE COMPUTER REVOLUTION:

continued from page 37

hearing, you who are strong and brave and kind. Will you not constitute yourselves Knights of the Blind in this crusade against darkness?

I thank you.

The description of the personal computer for the deaf/blind is at the end of the article. I just wish to emphasize that your implementation should always keep in mind the communication between the deaf/blind and the sighted.

I see the implementation of this idea as opening up a new inexpensive world to the deaf/blind.

Will not some of you programmers take up this challenge and become—to paraphrase the words of Helen Keller—

KNIGHTS OF THE DEAF/BLIND!

DESCRIPTION OF A PERSONAL COMPUTER FOR THE DEAF/BLIND

OUTPUT: A port with a switch. The switch being turned on and off at the rate necessary to send morse code characters. The deaf/blind programmer will connect his own code oscillator and touch device to the output of your switch.

INPUT: The keyboard of the personal computer.

COMMUNICATION BETWEEN THE DEAF/BLIND AND THE SIGHTED: The video display.

PROGRAM: The basic idea is this: When your machine is in the command mode in BASIC you figure out how to patch a "preprocessor" into BASIC which allows your program to look for a special character, such as CONTROL-T (which is decimal 20 on the TRS-80 Model I). If the key hit is not CONTROL-T, you pass the character along to the BASIC command interpreter as usual. If the character is CONTROL-T you take control away from BASIC and wait for the operator to hit NN "ENTER" where NN is a LINE number—the number of the line displayed on the video screen, and "ENTER" is the carriage return usually found in BASIC. When you get the NN "ENTER" you convert the text found in RAM associated with the line number requested to morse code. You make an extension to the morse code for an code for an unprintable character: Unprintable Character = • • — — , a sort of backwards "Z". Try to keep extensions to the morse code to an absolute minimum. Once you know you have an unprintable character you can PEEK memory for its decimal representation and send out the morse for the numerical value.

Also, do not under any circumstances redefine the Morse code—since this is common ground to many people who are NOT deaf/blind. Try to keep in mind that the communication link between the sighted and the deaf/blind is what is shown on the video screen. The deaf/blind programmer "sees" the screen using the simple protocol described above and the sighted person sees the same thing by looking at the screen. This communication link is very important because when a sighted person hires a deaf/blind programmer he must be able to communicate effectively with him; and when you are interviewed by a deaf/blind entrepreneur you have a better chance of getting the job if you can communicate with him.

To accomodate different skill levels of deaf/blind

people reading the morse code, you will have to include a command in the preprocessor to change the rate of sending morse code characters.

You should incorporate a morse "feedback" sounding out every character hit on the keyboard.

You should probably incorporate a "stop sending" command to allow the operator to make notes in Braille. ■

XSNORF

continued from page 36

Listing 3. Printout of game.

XSNORF-WORD GAME

BEST DEFINITIONS
OF RANDOM WORDS

WIN POINTS!

1ST= 7 POINTS
2ND= 3 POINTS
3RD= 1 POINT

ROUND NUMBER 1

1 PTEWIE
2 SPBNPN
3 JUNEOR
4 DPARIM
5 ALTMUM
6 EDDRNL
7 NZEDOE
8 PFWMAT
9 RRITST
10 ERIOOQ

SHFT S TO SCORE

JOHN: 38 PTS

SALLY: 36 PTS

DAVID: 36 PTS

...

ROUND NUMBER 2

1 XQKUER
2 JOCDAO
3 RAKBUH
4 NFRMLY
5 YISHKP
6 KYQUGP
7 AQNMOD
8 GBJGZK
9 ESQAEV
10 TDXOEL

SHFT S TO SCORE

JOHN: 70 PTS

SALLY: 82 PTS

DAVID: 68 PTS

...

GAME OVER!

".

Anthony T. Scarpelli

98 Foxcroft Dr.

Scarborough, ME 04074

Leigh T. Whittemore

17 Latham St.

South Portland, ME 04106 ■

COBOL PRIMER #3

H. C. Goodrich

Text Abbreviations:

CUG: Cobol User's Guide

CRM: RSCOBOL Reference Manual

CEDIT: Source Program Editor

This is the third in the series of lessons on RSCOBOL (Ryan McFarland Corp.). Hopefully, you have keyed in and modified at least two COBOL programs and have become familiar with CEDIT for input and editing, RSCOBOL for the compilation of your source code, and RUNCOBOL for the running of your compiled object code. Also, you should have familiarized yourself with the various RSCOBOL manuals in addition to a general text on COBOL.

Just as a reminder, I will not present a great deal of explanation on COBOL in these articles. I do plan to give you code that will run while you read and study to gain understanding. In other words, you teach yourself while I provide practice material for the TRS-80 Model III (though they should also work on the Models I and II).

So far, I have shied away from file handling and concentrated on instructions and techniques which can function without files. This idea will continue for the next few lessons.

First, let's review the COBOL instructions we have learned:

DISPLAY; ACCEPT (LINE#, POSITION#, TAB & ERASE); STOP RUN; IF; & GO TO.

DATA (or variable) NAMES; PICTURE (with type & size using X 9, and VALUE SPACES/ZEROS).

Now we are going to do some elementary arithmetic. Load your CEDIT and I(nput) the following program. (I am going to skip some of the line-number-commentary on topics which have been covered previously, such as the DIVISIONS and SECTIONS.)

```
100 IDENTIFICATION DIVISION.  
110 PROGRAM-ID.      PROGRAM3.  
120 AUTHOR.          H C GOODRICH.  
130 DATE-WRITTEN.    AUG 13, 1982.  
140 ENVIRONMENT DIVISION.  
150 CONFIGURATION SECTION.  
160 SOURCE-COMPUTER.  TRS80III.  
170 OBJECT-COMPUTER.  TRS80III.  
180 DATA DIVISION.  
190 WORKING-STORAGE SECTION.  
200 01 ACTION        PIC X      VALUE SPACES.  
210 01 YES-NO-SWITCH PIC X      VALUE SPACES.  
220 01 NUMBER-1     PIC 9(2)   VALUE ZEROS.  
230 01 NUMBER-2     PIC 9(2)   VALUE ZEROS.  
240 01 TOTAL         PIC 9(4)   VALUE ZEROS.  
250 01 LITERAL-1    PIC X(10)  
260     VALUE "ARITHMETIC".  
270 01 LITERAL-2    PIC X(35)  
280     VALUE "SELECT THE FUNCTION BY SIGN (-+/*Q)".  
290 PROCEDURE DIVISION.  
300 START-PROGRAM.  
310     DISPLAY LITERAL-1 LINE 1, POSITION 25, ERASE.  
320 DIRECTIONS.  
330     DISPLAY "THIS PROGRAM WILL DO ARITHMETIC WITH TWO"  
340         LINE 3, POSITION 5.  
350     DISPLAY "DIGITS. YOU WILL BE ASKED TO SELECT THE
```

OPERATION".

```
360     DISPLAY "YOU WISH TO PERFORM. PRESS THE (- + / * OR  
Q) KEYS".  
370 CALCULATE.  
380     DISPLAY LITERAL-2 LINE 6, POSITION 5.  
390     ACCEPT ACTION LINE 6, POSITION 42, PROMPT.  
400     IF ACTION = "Q" GO TO END-IT.  
410     DISPLAY "ENTER FIRST NUMBER", LINE 7, POSITION 5.  
420     ACCEPT NUMBER-1, LINE 7, POSITION 25, TAB.  
430     DISPLAY "ENTER SECOND NUMBER", LINE 8, POSITION 5.  
440     ACCEPT NUMBER-2, LINE 8, POSITION 25, TAB.  
450     IF ACTION = "-" SUBTRACT NUMBER-2 FROM NUMBER-1  
GIVING TOTAL.  
460     IF ACTION = "+" ADD NUMBER-1, NUMBER-2 GIVING TOTAL.  
470     IF ACTION = "/" DIVIDE NUMBER-1 BY NUMBER-2 GIVING  
TOTAL.  
480     IF ACTION = "*" MULTIPLY NUMBER-1 BY NUMBER-2 GIVING  
TOTAL.  
490     DISPLAY NUMBER-1 ACTION NUMBER-2 " = " TOTAL.  
500     DISPLAY "DO YOU WANT ANOTHER (Y/N)"  
510         LINE 14, POSITION 14.  
520     ACCEPT YES-NO-SWITCH, LINE 14, POSITION 42, PROMPT.  
530     IF YES-NO-SWITCH = "N" GO TO END-IT  
540         ELSE DISPLAY " " LINE 1, POSITION 1, ERASE  
550         GO TO CALCULATE.  
560 END-IT.  
570 STOP RUN
```

COMMENTARY

Line 130: This line under the IDENTIFICATION DIVISION is not required. It is commentary only. In addition, you could also add INSTALLATION & SECURITY (CRM 28-9,270). I will not be using them in these lessons.

Lines 250-80: The VALUE phrase may contain the figurative constants SPACES (for alphanumerics) or ZEROS (for numerics). It may also be followed by a literal bounded by quotation marks.

Line 300: Paragraph name.

Line 310: Displays the title held in LITERAL-1.

Line 320: Paragraph name.

Lines 330-60: Displays the directions starting on line 5 of the screen and tabs over to the 5th column. The two lines which follow will be printed directly below line 5 (on 6 and 7) in column 1. If you wish, you may specify placement by entering LINE# and POSITION# for each DISPLAY instruction.

Line 370: Paragraph name.

Lines 380,410,430: Display prompts on alternate lines.

Line 390: Like INPUT, ACCEPT assigns a value to the data name ACTION. TAB (CRM 142) requires you to use the <ENTER> key when you wish to continue. No TAB is like INKEY\$ but differs in that it will accept the length of values specified in the PICTURE clause for that variable. In this case, the input requires only one key stroke, so as soon as it is entered, the program will fall down to line 400. PROMPT (CRM 141) will provide an underline character the length of which is specified by the PICTURE clause (CRM 64).

Lines 420, 440: As the input may be one or two characters, TAB is used to allow backspacing to edit the number.

CUSTOMER RESPONSE TIME CALCULATOR

continued from page 29

Lines 450-80: Four conditional (IF) sentences are followed by four different arithmetic operations. The calculations are performed using the two values being held in "NUMBER-1" and "NUMBER-2". The result of calculation is placed in the variable "TOTAL". A general description of arithmetic operations may be found: (CRM 112-14; ADD 145,279; DIVIDE 162,281; MULTIPLY 182,283; and SUBTRACT 218,286). The expression IS EQUAL TO or EQUAL may be used in place of the "==" sign.

Line 490: The current contents of the four DATA names plus the "==" sign are display.

Line 540: ELSE is another variation used with the IF conditional. (Notice: in lines 400 and 530, the GO TO goes to a Paragraph Name, not a line number.)

Now you should check and edit the program for errors of syntax. You may wish to try three additional CEDIT instructions: F(ind), H(ard copy) and R(eplace).

F (CEDIT 10) is used with a string such as F/PIC/1 which will find the first example of that string starting at the current cursor position and display it on the screen. If you are looking for only one occurrence the number is not required. If you wish to look for greater numbers, change the 1 to that number and each line in which the string occurs will be displayed. If you wish to find all occurrences and are unsure of the number, us an "*" in place of the number.

H (CEDIT 11) is similar to LLIST in BASIC. H 200 will send that line to the printer; H200:300 will send those lines inclusive; H#:* sends all lines from the top to the bottom of your program.

R (CEDIT 14) is used with a line number and combines the D(elete) and I(nsert) functions (e.g., Type R, line#. The line will be deleted and you will be in the I(nsert) mode for that line number.)

During compilation of your program with RSCOBOL, we've tried the O(utput to disk #), P(rinter), T(ube), and E(error) switches. Used in combination, these are the ones you will use most often. However, you might want to try D(ebbug)(CUG 17,26-7) if you wish to see the code in HEX. But a more convenient utility might be X(cross reference). This output will be printed at the end of your program and will become increasingly more useful as you get into longer programs to help trace those illusive bugs.

I should mention a number of points about program3. I have not included error routines so you should be careful with your input especially with dividing. Be sure that the first number is larger than the second and that it is an even answer (only integer input and answers are permitted with this simple sample source).

You will notice also that the NUMBER-1 and NUMBER-2 and TOTAL data names may display zeros if the data are small. No doubt you have remarked (politely I hope) that it doesn't look very good. Next lesson, I will show how COBOL edits the field and suppresses or formats the output. I will also provide a demonstration of more math.

Until then, read, practice, modify, experiment!

Hubbard C. Goodrich
RFD 1, Box 880
South Harpswell, ME 04079 ■

```
1035 GOSUB 2000
1040 X1=(-R)+(N*(LOG(T/MU)/LOG(10)))
1045 X2=10>X1: X5=X2+X5
1050 NEXT N
1055 Y=0:FOR N=(S+1) TO M
1060 IF S=1 THEN R=0:GOTO 1080
1065 IF S=2 THEN R=.3010:GOTO 1080
1070 GOSUB 2000
1080 Y9=((N-S)*(LOG(T/(MU*S))/LOG(10))):Y7=(-R+(S*(LOG(T/MU)/LOG(10))))
1085 Y1=10>(Y9+Y7)
1090 Y=Y1+Y
1095 NEXT N
1100 P5=1/(X5+Y)
1110 V=1:FOR H=1 TO S:V=H*V:NEXT H
1120 A1=P5*((T/MU)>S)*(T/(MU*S))
1130 A2=V*((1.00009-(T/(MU*S)))>2)
1140 A3=1-((T/(MU*S))>(M-S))
1150 A4=(M-S)*((T/(MU*S))>(M-S))
1155 A5=1-(T/(MU*S))
1160 LQ=(A1/A2)*(A3-(A4*A5))
1170 K=0:FOR N=0 TO (S-1)
1180 R=1:IF N=0 THEN 1200
1185 FOR L9=N TO 1 STEP-1
1190 R=L9*R:NEXT L9
1200 K2=(S-N)*(((1/R)*((T/MU)>N))*P5)
1210 K=K2+K:NEXT N
1220 L=(LQ+S)-K
1230 TH=MU*(S-K)
1240 WQ=LQ/TH
1250 RETURN
2000 R=.3010
2010 FOR B=3 TO S
2020 Y8=LOG(B)/LOG(10)
2030 R=R+Y8
2040 NEXT B:RETURN
3000 IF T$="MONTH" THEN JX=12:RETURN
3010 IF T$="WEEK" THEN JX=52:RETURN
3020 IF T$="YEAR" THEN JX=1:RETURN
3030 IF T$="DAY" THEN JX=260:RETURN
4000 REM TEAM STRUCTURE
```

CONCLUSION

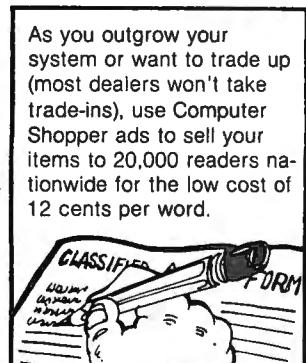
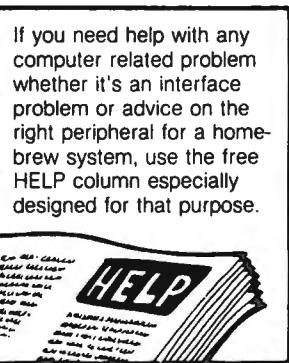
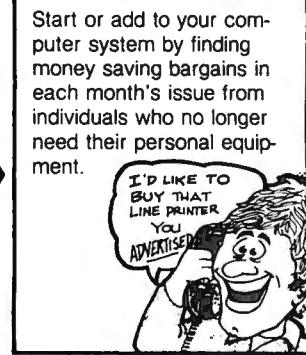
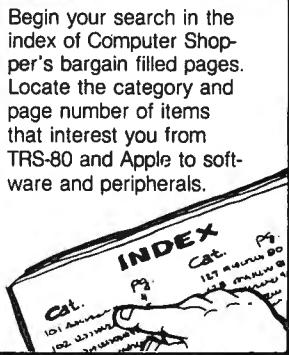
A customer waiting "too long" for repair service is not a good policy in any business. The need to know and understand service response times to customer calls has become a critical success factor for the field service manager. Computer programs like "RESPONSE" should assist field service managers to test different operating strategies and their impact on customer response times. The bottom line for field service manager is the customer's perceived value of the services provided; and the value of waiting "too long" is ominous.

PROGRAM NOTES

"RESPONSE" was written on a TRS-80 Model III microcomputer (TRSDOS 1.3). For additional information write to the author.

Dennis P. Avola
20 Prescott Street
Rutland, MA 01543 ■

How to Buy or Sell Computer Equipment and Software



Computer Shopper is THE nationwide publication for buying, selling and trading Micro and Mini-computer equipment and software. Each issue has over 100 pages full of bargains of new and used equipment.

You can save hundreds of dollars by getting the equipment you need from the hundreds of classified ads individuals place in Computer Shopper every month.

Now is the time for you to join over 20,000 other computer users who save time and money with a subscription to Computer Shopper.

Subscribe today and get your first issue and a classified ad absolutely FREE. Type or print your ad on a plain piece of paper and send it along with your subscription.

Just fill in the coupon or MasterCard or VISA holders can phone for faster service and start making your computer dollar go further today.

Cut out and mail to: **COMPUTER SHOPPER**
P.O. Box F535 • Titusville, FL 32780

Yes, I'll try Computer Shopper, I understand that if I'm not satisfied with my first issue I can receive a full refund and keep the first issue free.

- 1 year \$10.00 (\$30.00 in Canada)
 I have enclosed my free classified ad.
 I want to use my free ad later, send me a coupon.

NAME: _____

ADDRESS: _____

CITY: _____

STATE: _____ ZIP: _____

ASK RICHARD

Richard Kaplan

I know that a **modem** is a device used to help a computer "talk" to other computers with a telephone. How was that name derived?

The term **modem** stands for MODulator—DEModulator. Modulation and demodulation refer to the process needed to convert a computer's signals into signals to be sent over a telephone wire.

Are there other devices related to computers which have that type of abbreviation as part of their name?

Yes. Many "computerese" terms have their etymology (origins) in such abbreviations. An example of another acronym (a word made from the first letters of a series of words) is a **spooler**, which was discussed in a previous installment of "Ask Richard." A spooler is a device which lets your printer produce a report while it is actually doing something else. The word spooler stands for Simultaneous Peripheral Operations On Line.

Weren't the names of many computer languages formed in a similar manner?

Yes, they were. The following are the derivations of some of the most popular computer languages:

- * **BASIC** — Beginner's All-Purpose Symbolic Instruction Code
- * **FORTRAN** — FORmula TRANslator
- * **COBOL** — COmmon Business Oriented Language
- * **PL/I** — Programming Language I (This acronym is disputed—some say PL/I does not stand for anything at all.)

I know my computer can run BASIC programs. Is there any way I can use some of these other languages, as well?

Yes, most computers can use a variety of languages. In the case of TRS-80's (and many other computers, as well), you may need to use CP/M instead of TRSDOS if you wish to use a language other than BASIC.

What are CP/M and TRSDOS?

CP/M was explained in a past installment of "Ask Richard." Put simply, an **operating system** is a "master program" which tells the computer in what format to store information on disks, how the video display works, etc. TRSDOS is the standard operating system for TRS-80 computers, but it works only on TRS-80 computers, and it is even different for different model TRS-80's. CP/M, on the other hand, works on hundreds of different computers. Therefore, the makers of languages such as FORTRAN, COBOL, and PL/I for microcomputers usually choose to write the programs for CP/M, since in that way the largest percentage of computer owners will be able to use their programs.

I have heard that BASIC is the easiest computer language to learn. Why would I want to use FORTRAN, COBOL, or PL/I?

BASIC is undeniably the easiest computer language to learn. However, there are several reasons why you might choose to use a language other than BASIC.

BASIC is relatively slow compared to many other languages. One solution to this slow speed is to **compile** your BASIC program when it is finished, which involves running a special program (called a **compiler**) which translates the program from BASIC to machine language.

Machine language is the computer's built-in language of 1's and 0's. Although it is very difficult to read or understand, it is extremely fast.

Another solution to BASIC's slow speed is to use another language. Although machine language is the fastest language available on a microcomputer, it is extremely difficult to understand. Therefore, a compromise often struck is the use of a language such as FORTRAN, COBOL, or PL/I, which are very often faster for their specific purposes.

What do you mean when you say FORTRAN, COBOL, or PL/I are often faster for their specific purposes?

That question brings us to the second reason why many people choose not to use BASIC. FORTRAN, COBOL, and PL/I are each designed for a specific purpose. When used for this purpose, the language is easier and faster to use than BASIC.

What is the primary use for FORTRAN?

FORTRAN's derivation — FORmula TRANslator — essentially sums up this language's use. FORTRAN is designed for programs which solve complex mathematical formulas. The language has been designed to allow mathematical problems to be entered into the computer in a manner similar to the way a mathematician or a scientist thinks.

What is the primary use for COBOL?

COBOL stands for COmmon Business-Oriented Language. As the name implies, this language is designed for applications common to businesses. Someone looking at a COBOL program might accidentally think he is looking at an outline or an essay! This is because COBOL programs are written in paragraph form so that they will be somewhat understandable to people not very familiar with the language. COBOL is also designed to develop detailed reports, which are often necessary in business.

What is the primary use for PL/I?

PL/I is in many ways a mixture of other languages. It was designed to be an all-purpose language which could do anything that any of the other "specialized" language could do. Unfortunately, PL/I has not caught on to the degree that IBM, its designer, had originally hoped. I personally prefer PL/I over any other language, including BASIC. However, the main argument against this language is that, by adding so many features to a language, it became too cumbersome to understand, and someone could never master the entire language. I say this does not matter—one need learn only as much of the language as he needs. In any case, PL/I has not caught on with anywhere near the enthusiasm IBM had hoped when it first introduced the language, with the intention of its becoming the most popular language in existence.

Questions not attributed to a specific person represent a typical computer user, a composite "character" drawn from the author's personal experience speaking with customers of H & E Computronics.

Richard Kaplan
H & E Computronics
50 N. Pascack Road
Spring Valley, NY 10977 ■

METRIC CONVERSION

Hubbard C. Goodrich

The following program was written on a TRS-80 Model III with TRSDOS. It should work equally well under Level II, with the possible exception of the POKEs (lines 150 and 490). The program is 2320 bytes long. This could be reduced by eliminating all REMs, spaces and combining DATA statements without changing the program flow.

I was stimulated to write this program by a friend who had proudly given me a copy of his efforts. A brief glance was enough to show that he had been unnecessarily redundant in his routines—using little of the power of the computer to reduce memory requirements, and making little effort to increase the speed or flexibility of the program. While not the final word in compactness or speed, I thought it might be of interest to others who seem to ignore possibilities of subroutines and loops to increase programming efficiencies.

This program was designed to show how a generalized program can be written to retain sufficient flexibility so that expansion becomes simple. Once these subroutines have been written, you can add other conversions with relative ease. My purpose is to demonstrate one way that this efficiency may be achieved. First, I will describe the program, and then show how additions (or substitutions) may be made.

All of the subroutines are at the top of the program so line 20 send us to the actual beginning at line 150, where we load the array variables and perform other house keeping functions. Line 160 clears the screen, establishes the number of Menu choices (4) and prints the title. Line 170 displays the Menu (use down arrow and tab) and goes to subroutine 60 to pick up your choice. Line 180 sends you to the end or to the conversion of your choice. Lines 200, 210, 220, and 230 are all identical in function but provide opportunities to convert Length, Area, Volume and Weight. I'll use line 200 as an example.

Line 200 establishes the number of choices, the starting array number, clears the screen, prints the subtitle, and goes to subroutine 130. Line 130 sends us to display a message (line 40), displays the appropriate menu using the array variable and loop (70), and asks for our choice (60).

Returning to line 200, there is a check for a "quit" response. Then we GOSUB to 90 for the display of our choice in the proper order (metric to nonmetric or nonmetric to metric). We are then asked to enter the amount we wish to have converted (K=1 is the default), and in line 110 the appropriate display and conversion are completed. (Notice: these routines are used in all the conversions.)

We remain in this loop for "length conversion" until we enter a "0" to exit the routine, which returns us to the main menu.

Now, how can we add additional conversions?

1. Determine your subject items for the data statement:

(ie. Dry Measure)

CUBIC METERS,28.37759,BUSHELS
CUBIC LITERS,.908083,QUARTS

and add them to the bottom of the DATA.

2. Change data line 500 to include the two new lines.
- Change 12 to 14.
3. Change in line 160, N=4 to N=5.
4. Add to line 170 Menu "5 ... DRY MEASURE".
5. Finally, Add line 240: N=4:N1=12:(and continue).

The final statement should be GOTO240.

Conversions of any length should work. Screen formatting presents the only limitations.

Variable Table

A\$	ANSWER
E\$()	CUSTOMARY LABEL
F()	CONVERSION FACTOR
H\$	HAND (delete for level II?)
I	LOOP
K	CHOICE
M	METRIC / CUSTOMARY
M\$()	METRIC LABEL
N	NUMBER OF CHOICES
N1	START ARRAY
N2	NUMBER OF DATA STATEMENTS
X	MENU CHOICE

PROGRAM LISTING

```
10' ## METRICON/BAS ## METRIC CONVERSION
20 GOTO 150
30' ## START SUBROUTINES ##### 
39'
40 PRINT:PRINT "WHICH DO YOU WISH TO CONVERT? (0 TO END)":
PRINT:RETURN
49'
50 PRINT @985,STRING$(5,136);";:INPUT" <ENTER> TO CONTINUE";
A$: RETURN
59'
60 PRINT @640,CHR$(31)::PRINT "ENTER THE NUMBER OF YOUR CHOICE
(0 -";N;"") -->";: INPUT X: IF X=0 THEN RETURN ELSE IF X<1 OR
X>N THEN 60 ELSE RETURN
69'
70 M=(N*.5): FOR I=1 TO N: PRINT @200+(I*64),I;" - ";
IF I<M+1 PRINT M$(I+N1);" TO ";E$(I+N1):: ELSE PRINT
E$(I+N1-M);" TO ";M$(I+N1-M);
80 NEXT I: GOSUB 60 : RETURN
89'
90 PRINT @705,CHR$(31):: IF X<=M THEN PRINT M$(X+N1):: ELSE
PRINT E$(X+N1-M);
100 K=1:INPUT": NUMBER TO CONVERT -->;K
110 PRINT @835,H$;" "; K;" ":"; IF X<=M PRINT M$(X+N1);" = ";
K*(F(X+N1));E$(X+N1): ELSE PRINT E$(N1+X-M);" = ";
K*(1/(F(N1+X-M)));M$(N1+X-M)
120 GOSUB 50 :RETURN
129'
130 GOSUB40 :GOSUB70 :RETURN
140'
149' ##### START PROGRAM #####
150 CLEAR 200:CLS: DEFINT I-X: RESTORE:READ N2:
DIM X,M,N,N1,K,I,F(N2),E$(N2),M$(N2): PRINT @475,"LOADING
DATA":FOR I=1 TO N2:READ M$(I),F(I),E$(I): NEXT I:N2=0:
POKE 16420,1: H$=CHR$(191)+CHR$(191)+CHR$(244)+CHR$(245)+CHR$(246)
160 CLS:N=4: PRINT @10,"METRIC-ENGLISH CONVERSION": GOSUB 40
```

continued on page 54

BEGINNER'S CORNER

continued from page 16

ficult in BASIC.

I just found out that not all assemblers are alike or even use the same mnemonics for the same machine. The Z80, for example, is capable of executing 8080 code as well as Z80, but if you look at some texts on the subject, 8080 code looks quite different from Z80. For one thing the Z80 has quite a few more instructions than the 8080. The reason for this is because the Z80 is a newer machine and was designed to be "upwardly compatible" (is UPWARDLY really a word?) with its predecessor processor.

Some companies even use different mnemonics for the same machine. Intel, for example has an assembler that is different for the Z80 than that of Zilog. I've read somewhere that the reason for the different mnemonics was to avoid law suits. This may or may not be true, but it points out that anyone can design an assembly language using their own mnemonics if they want to do it.

Some of the various features that assembly language programmers demand account for the variety in the types of assemblers. Here's a brief list of some of the kinds of assemblers around:

MACRO ASSEMBLERS

CONDITIONAL ASSEMBLERS

CROSS ASSEMBLER

"HIGH LEVEL" ASSEMBLERS

A macro assembler is a very interesting type of program that has nothing what-so-ever to do with spaghetti programming (macro-rone—get it? Sorry!). In fact, it's designed to alleviate many problems that arise due to the constant repetition of code.

A "macro" is in essence a subroutine. No doubt you've come across this kind of concept before. But in assembly, a macro is a series of instructions that appears over and over in various places but with slight changes. Wouldn't it be nice simply to be able to tell the program to insert what you want into the place you want it and to get the routine from a library you created?

That's what a macro assembler does. It allows you to insert a subroutine at the moment of assembly. This is usually done by telling the program to "call sub a,b" and then letting the system do the rest.

It's considered very good programming practice to write the same way in BASIC. In fact, the book *BASIC Faster and Better and Other Mysteries* is based on this principle. Write small routines to accomplish simple tasks, and then combine them to create perfect working programs. It's a great idea you should take advantage of.

Now, imagine that you have a program to write and a group of subroutines with which to assemble it. However, not all the routines are necessary or should be used. In some situations one routine is needed and in other situations other routines are required. This calls for a conditional assembler.

This is used to create programs that should have lots of options. The program calls a particular routine to be assembled depending on what went before. You accomplish this in a way similar to BASIC by using assembler directives (orders) that state what to do, i.e. AGO stands for Assembler GOTO and AIF is for Assembler If. This kind of program allows for a very flexible system that

creates a finished program to your specifications.

By far one of the most interesting types of assemblers is the CROSS ASSEMBLER. This kind of assembler allows you to assemble a program for a different processor. For example, I could use a cross assembler on my Z80 to assemble code for a 6502 or 6809. This allows the user to input code in the language he is most familiar and output code usable by the "target" machine.

The idea of a HIGH LEVEL assembler is new to me. I've never seen one on a micro, but I've read about them. The term HIGH LEVEL as it's used here is a little misleading. The term here means you have a precise control over the machine but the language looks more like high level languages in construction. You have the familiar loops and conditional statements with the usual functions and procedures in a fixed format.

This kind of assembler allow the use of labels (names for specific routines), reserved words and identifiers. The code is in block format (subroutine structure). Conditional and compound lines and arrays are supported (as opposed to being created by the user). Procedures and functions are possible, where data is passed from one function to another by an intermediate address being used for storage, and expressions are used as long as they're kept simple.

All these kinds of constructs and forms are derived from higher level languages and are not usually found in assemblers. At least they're not as easy to create in assembly language. There is one assembler around with some of these features sold by the Alternate Source. Someday I'll look into it.

I don't think there is an assembler with all these features in one convenient program, but if you look around and happen to be thinking of learning assembler (we all get there sometime), try to find the one that will suit your needs, not the one with the most bells and whistles.

You'll find most often that having the program that has the most is not always the best. That's why many texts still start the beginner using the Radio Shack series one assembler. It may be simple, but it does the job.

If your looking for a good monitor on the other hand, that's another story. Get MON-5 — it happens to be one of the best and easiest with which to work.

Well, it's getting close to that time, and I'm afraid there's bad news. By now you know that ~~CCFT2~~ Computronics ~~CCFT1~~ will be no more. So, I want to take the time now to say, it's been a pleasure writing for you, and I hope I gave you some insights and information you wouldn't have gotten easily elsewhere.

I want to say thanks to those of you who have written to me over the past year expressing interest and giving criticism (Mom and Pop, you've been terrific). You've been a great help and I appreciate it. I'd like to say thanks especially to Hubert S. Howe Jr. for giving me the opportunity to write for you in the first place and to Joseph Rosenman for having the nerve to even suggest the idea.

It's been a pleasure. Thank you and keep hacking. You'll get the hang of it yet.

Spencer Koenig
153-27 73 Avenue
Flushing NY 11367 ■

MICROFUTURE

Sallie Stephenson

Mei Ling gave the computer a final keyboard command, nodded to Dr. Tzu, and stood up. Her eyes followed the program in process on the CRTscreen and the curvature of the huge electromagnet behind it. From the windings of the coil, she followed the arch up and over to where it terminated at the surface of a quartzite sphere two meters in diameter. The whole apparatus was the work of Dr. Tzu, who formulated the core material that enabled incredibly dense magnetic fields to be generated. Mei Ling's eyes came to rest on the flask she had just attached to the sphere. Following Dr. Tzu's instructions, she had taken the flask from the cryogenic vault where dozens of similar vessels were stored. Frost-covered, it now clung wart-like to the side of the crystal sphere.

"Composition, Mei Ling?" called the deep voice of Dr. Tzu.

She stepped forward and cleared the condensation from the identification plate. "Carbon—10 units, hydrogen—4 units, nitrogen—30 units, oxygen—20 units, trace element mixture—7837."

Dr. Tzu stepped to the computer and keyed in the parameters. The nearby console lights flashed and the printer's staccato indicated that the operations were being recorded.

"Prepare to energize," came the command.

Mei Ling turned to the console and depressed several switches. After a moment she noted several readings on meters and recorded them on a notepad.

When her notetaking was completed, she approached the sphere and peered intently into its interior. The lights in the room faded and only a few panel indicators and the console lights furnished illumination.

"10 seconds to energization."

Mei Ling's skin prickled as she imagined the tremendous currents starting to flow through the windings of the electromagnet. Soon, the very air near the magnet seemed to pulse and throb in response to the magnetic fields forming and building between the poles.

Mei Ling turned her attention to the sphincter valve to which she had earlier secured the flask. It throbbed and pulsed and a small puff of gas entered the sphere where it was immediately gathered in by the magnetic fields and compressed into the center. Rhythmically, the valve pulses again and again. Each time another wisp of gas was seized by the intense fields. The tiny ball of gas at the sphere's center began to glow. The valve stopped operating while the computer put out data on its printer.

The gaseous ball started behaving differently. The ball was flattened and compressed, kneaded and twisted. It writhed and glowed even brighter. And then the "transformation" occurred.

Mei Ling uttered a gasp of surprise and awe. Where there had been a writhing twisting gaseous mass, there was now a slowly expanding mass of pinpoint-sized lights. So intense were these tightly packed points that in the darkened room it hurt her eyes to look at them. As the pulsations from the electromagnet slowed, so did the expansion of the mass of lights.

Mei Ling stepped back from the sphere as Dr. Tzu wheeled over an electron camera and directed it against the surface of the sphere. Moving to the rear of the camera, Mei Oing looked into the viewer. On a relatively low magnification, the points of light grew larger and were transformed into spirals. Fascinated, Mei Ling watched as they slowly came into focus and then drifted out, only to be replaced by others. She increased the magnification even further, and the spiral, slowly coming into focus, resolved itself into millions of tiny light points, a miniature galaxy. It blurred, and another took its place, this one edgewise. Reluctantly stepping back, Mei Ling surrendered the instrument to Dr. Tzu, who made the final adjustments.

Soon the camera started an erratic, subdued clicking as the computer controlled the photographic device at the critical time when an object was in focus. Several hours passed during which the camera continued its clicking while the computer output data on its printer.

"Prepare to deactivate."

Dr. Tzu's voice broke Mei Ling's concentration. She quickly assumed her place at the console and prepared to record readings.

Dr. Tzu once again made adjustments at the computer and the fields generated by the electromagnet again seemed to cause the room to vibrate.

Mei Ling's instruments monitored the size and growth rates, field densities and emanations generated inside the sphere. The readings now showed that the growth of the universe had ceased and the final phase, the decay period, would begin.

"Growth has stabilized, Dr. Tzu," Mei Ling reported.

"Good," responded Dr. Tzu. "I can handle the rest by myself."

"I'd like to stay a while and watch the decay," said Mei Ling. Returning to the sphere, she became completely absorbed in the scene before her eyes. She adjusted the viewer and brought a galaxy into focus. Raising the magnification, she watched. The tiny pinpoints of light composing the spiral now in focus shone coldly. Now and then one would flare with increasing brilliance and then slowly die out.

While Mei Ling was watching the decay, Dr. Tzu stopped the camera momentarily and retrieved the pictures that had been taken during the experiment. Then he reset the camera, making additional adjustments for the final phase just beginning. Dr. Tzu handed Mei Ling the stack of photos.

"See any you like?" he asked, taking a seat near her so they could look over the pictures together.

Mei Ling nodded in approval as she observed the pictures, each seemingly more breathtaking than the previous one. The pictures unveiled galaxies, nebulas, star clusters, single stars, binaries, and occasionally, a planet.

"Look," exclaimed Mei Ling. The photo she held showed a distant yellow sun, and in the foreground was a planet with rings. "Isn't it beautiful?"

"Like Saturn," remarked Dr. Tzu, suddenly caught up

continued on page 46

EVERLASTINGLY AT IT

YOU'RE AS YOUNG AS YOUR MICROPROCESSOR

RETIREE ENTHUSIASTICALLY "BYTES" INTO

WORLD OF PERSONAL COMPUTERY

Mike Shadick

Stewart Buckingham of Anaheim, California, rather lovingly refers to his personal computer as "my TRS-82." Does he, then, have a specially-modified TRS-80, or what?

"Actually," Stew explains with a grin, "it's more like the other way around. My computery has modified me!" Which is a rather remarkable statement, coming from someone whose modifying days are over, some might say.

For Stew, you see, has been around as long as this century.

Born in 1899, this native Californian (a rare enough species, these days), is also one of computerdom's newest converts.

"I've had my TRS for only a couple of months," Stew points out with justifiable pride, "and already I'm up to my disk drives in programs, more than one of which I've written myself."

A most amazing accomplishment? Not to hear Stew himself tell of it! "I borrowed all the BASIC programming literature I could get my hands on from the library," he recalls, "and now I'm teaching myself the rudiments of PASCAL."

Stewart is not at all unmindful of his "senior" status, and has in fact developed several BASIC programs to help his fellow septuagenarians and others. One of his programs he calls SENFIN (derivation of name unknown, although you can guess), a financial management software package (that's what Stew calls it!) designed especially to meet the needs of Stew and his peers. "We seniors have special financial needs and situations," he explains. "So we worked up a program which takes all our special variables into account, so to speak."

Though Stew has no plans to market SENFIN as such, he is willing to "work something out" with any interested parties.

Yet another area of home computery which fairly fascinates Stew is—can you guess—video games! As you might surmise, he has designed and programmed a couple of his own, including a "Monopoly"-like game where people "roll" their way around, not on a board but through life! "You begin the game by being born," Stew explains, "and—well, you take it from there!" Perhaps needless to say, this particular video game has its share of surprise endings!

Not at all incidentally, Stew has managed to get quite a number of his fellow seniors interested in home computery in general, and in TRS-80 operation and programming in particular. "It's not that difficult," he admits. "I mean, the sort of logic which is required for BASIC programming, is something which comes rather naturally, really, to those of us from the old school. Once we get over the culture-shock or amazement or whatever-you-wish-to-call-it, of just how much a

computer can do for you at the touch of a couple of keys, why, the rest is almost child's play."

Those of younger generations sometimes see people of Stew's vintage as being—well, "over the hill." And Stew is the first to agree! "Y'know," he likes to philosophize, "a lot of life is spent in climbing, struggling, striving to get to the top of the heap, wherever that is. Well, when you've reached the top, or at least feel that you have, then the rest of life is really all downhill from there! And as far as I'm concerned, it's pretty smooth sailing. I wouldn't switch sides for the world!"

What, you might well be wondering, did Stewart Buckingham do for a living, prior to his retirement and his subsequent micro-discoveries. He was—and is—a furniture refinisher and restorer. "I love working with my hands," he observes. "I guess maybe that's one of the reasons why I got into TRS computery, really. I can see my own handiwork, and appreciate it—and so can others, especially fellow computer nuts. There's more of us around than I ever dreamed. And some of them have even had me refinish some of their treasured old wood antiques."

This is about the only thing which Stew has yet to devise a program to help him do. "But I'm working on it," he claims. And indeed, once having met the man and observed what he and his TRS-80 (excuse me, his TRS-82!) are already accomplishing, one is likely to believe that this senior partnership could accomplish most anything!

Mike Shadick
Cedar Square West, Apt. E-414
1515 South Fourth Street
Minneapolis, MN 55454 ■

MICROFUTURE

continued from page 45

in the beauty of the photographic exhibition.

They looked at the next picture, a planet. Wreathed in feathery clouds, its watery surface gleamed, broken by continental outlines. Mei Ling gasped when she saw the planet.

"No, no!" stammered Dr. Tzu as he ran to the console. "We've got to reverse the decay!"

Mei Ling did not respond as she helplessly looked up at the evening sky through the window above her. Her throat constricted in terror as one by one the stars winked out.

Sallie Stephenson
Cedar Square West, Suite F-810
1601 South Fourth Street
Minneapolis, MN 55454 ■

HOUSEHOLD INVENTORY

William O. Byberg

This program, "Household Inventory," is very useful in keeping track of what you own, and your insurance agent will love you for the print out, which can be updated as necessary. The program is for cassette users, but I'm sure that you disk operators can change the input and output routines for your own use. I'm not sure if it will run on a Model I (I wrote it on a Model III), but I'm sure that Model I users can adapt it.

```
10 REM ***** INVENTORY PROGRAM FOR HOUSEHOLD ITEMS *****
20 REM ***** BY WILLIAM BYBERG *****
30 REM ***** 103 SPIT BROOK ROAD, BLDG F, APT 11 *****
40 REM ***** SOUTH NASHUA, N.H. 03062 *****
50 REM ***** NO COPYRIGHTS - ANYONE CAN USE IT *****
60 ON ERROR GOTO 0
70 CLS: PRINT TAB(15) "DO YOU WISH INSTRUCTIONS ";: INPUT R$ 
80 IF R$="Y" THEN 1280 ELSE 90
90 CLEAR 10000: DIM A$(8,100): DEFINT A-Z
100 CLS: PRINT STRING$(63,61): PRINT TAB(18) "RECORD OF
HOUSEHOLD ITEMS": PRINT STRING$(63,61): PRINT: PRINT TAB(20)
"MASTER MENU": PRINT
110 PRINT TAB(5) "< A > ADD ITEMS TO FILE"; TAB(35) "< D >
DELETE FILE(S)"
120 PRINT TAB(5) "< C > CREATE NEW FILE(S)"; TAB(35) "< I >
INDEX - COMMON FACTORS"
130 PRINT TAB(5) "< L > LIST STORED FILE(S)"; TAB(35) "< M >
MODIFY FILE(S)"
140 PRINT TAB(5) "< P > PRINT FILE(S)"; TAB(35) "< T > LOAD
FILE(S) FROM TAPE"
150 PRINT TAB(5) "< S > SAVE FILE(S) ON TAPE"; TAB(35) "< E >
EXIT PROGRAM
160 PRINT @ 896, "ENTER SELECTION < >"
170 K$=INKEY$: PRINT @ 914, CHR$(176);: FOR X=1 TO 200: NEXT:
IF K$="" THEN PRINT @ 914, " ";: FOR X=1 TO 200: NEXT: GOTO
170ELSE PRINT @ 914, K$;: FOR X=1 TO 200: NEXT
180 IF K$="A" THEN 410ELSE IF K$="D" THEN 860ELSE IF K$="C" THEN
630ELSE IF K$="I" THEN 670ELSE
190 IF K$="L" THEN 660ELSE IF K$="M" THEN 430ELSE IF K$="P"
THEN 990ELSE IF K$="T" THEN 1160ELSE IF K$="S" THEN 1080ELSE IF
K$="E" THEN 1220ELSE
200 FOR Z=1 TO 20: PRINT @ 896, STRING$(15,32): FOR ZZ=1 TO 15:
NEXT ZZ: PRINT @ 896, "INVALID REQUEST": FOR ZZ=1 TO 20: NEXT
ZZ: NEXT Z
210 PRINT @ 896, "ENTER SELECTION": GOTO 170
220 CLS: PRINT "FILE # "; MID$(STR$(A),2): PRINT: INPUT "WHAT IS
THE NAME OF THE ITEM "; A$(1,A)
230 INPUT "WHO IS THE MANUFACTURER "; A$(2,A)
240 INPUT "WHAT IS THE MODEL "; A$(3,A)
250 INPUT "WHAT IS THE SERIAL NUMBER "; A$(4,A)
260 INPUT "WHEN WAS IT ACQUIRED "; A$(5,A)
270 INPUT "WHAT IS THE COST OR VALUE "; A$(6,A): IF A$(6,A) <>
"UNKNOWN" THEN A$(6,A)="$"+A$(6,A)
280 INPUT "WHERE IS THE ITEM LOCATED "; A$(7,A)
290 INPUT "ARE THERE ANY REMARKS "; A$(8,A):RETURN: ELSE RETURN
300 K=A: CLS: FOR P=1 TO K
310 PRINT "ITEM/MANUFACTURER"; TAB(22) A$(1,P); CHR$(47);
A$(2,P)
320 PRINT "MODEL/SERIAL NUMBER"; TAB(22) A$(3,P); CHR$(47);
A$(4,P)
```

```
330 PRINT "DATE ACQUIRED/VALUE"; TAB(22) A$(5,P); CHR$(47);
A$(6,P)
340 PRINT "LOCATION/REMARKS"; TAB(22) A$(7,P); CHR$(47);A$(8,P)
350 PRINT STRING$(63,61)
360 IF P/3 = INT(P/3) THEN 370ELSE 390
370 O$=INKEY$: PRINT @ 977, "PRESS ANY KEY TO CONTINUE ";: FOR
X=1 TO 200: NEXT: PRINT @ 983, STRING$(7,32);: FOR X=1 TO 200:
NEXT: IF O$="" THEN 370ELSE CLS
380 IF P=K THEN 100
390 NEXT P
400 O$=INKEY$: PRINT @ 977, "PRESS ANY KEY TO RETURN TO MENU ";: FOR
X=1 TO 200: NEXT: PRINT @ 983, STRING$(7,32);: FOR X=1 TO
200: NEXT: IF O$="" THEN 400ELSE 100
410 GOSUB 960
420 A=A+1: GOSUB 220: PRINT: PRINT: INPUT "DO YOU WANT TO ADD
MORE FILE(S) (Y OR N) ";W$: IF LEFT$(W$,1)="Y" THEN 420ELSE 100
430 H=23: GOSUB 960: CLS: PRINT "MODIFICATION MENU"
U$: GOSUB 690: GOTO 450
440 W=0: PRINT @ 960, CHR$(30);: INPUT "TYPE '0' TO CONTINUE,
OR # OF RECORD TO MODIFY ";W$: W=VAL(W$): IF W=0 THEN RETURN
450 V$=STRING$(3,27): CLS: PRINT "FILE # "; MID$(STR$(W),2):
PRINT: PRINT: PRINT "ITEM"; TAB(22) A$(1,W): GOSUB 620: IF
G$="Y" THEN 470ELSE PRINT
460 INPUT "WHAT IS THE NAME OF THE ITEM "; A$(1,W): PRINT V$
470 PRINT STRING$(3,27) CHR$(31): PRINT "MANUFACTURER"; TAB(22)
A$(2,W): GOSUB 620: IF G$="Y" THEN 490ELSE PRINT
480 INPUT "WHO IS THE MANUFACTURER "; A$(2,W): PRINT V$
490 PRINT STRING$(3,27) CHR$(31): PRINT "MODEL"; TAB(22)
A$(3,W): GOSUB 620: IF G$="Y" THEN 510ELSE PRINT
500 INPUT "WHAT IS THE MODEL "; A$(3,W): PRINT V$
510 PRINT STRING$(3,27) CHR$(31): PRINT "SERIAL NUMBER"; TAB(22)
A$(4,W): GOSUB 620: IF G$="Y" THEN 530ELSE PRINT
520 INPUT "WHAT IS THE SERIAL NUMBER "; A$(4,W): PRINT V$
530 PRINT STRING$(3,27) CHR$(31): PRINT "DATE ACQUIRED"; TAB(22)
A$(5,W): GOSUB 620: IF G$="Y" THEN 550ELSE PRINT
540 INPUT "WHEN WAS IT ACQUIRED "; A$(5,W): PRINT V$
550 PRINT STRING$(3,27) CHR$(31): PRINT "COST OR VALUE"; TAB(22)
A$(6,W): GOSUB 620: IF G$="Y" THEN 570ELSE PRINT
560 INPUT "WHAT IS THE COST OR VALUE "; A$(6,W): PRINT V$
570 PRINT STRING$(3,27) CHR$(31): PRINT "LOCATION"; TAB(22)
A$(7,W): GOSUB 620: IF G$="Y" THEN 590ELSE PRINT
580 INPUT "WHAT IS THE LOCATION "; A$(7,W): PRINT V$
590 PRINT STRING$(3,27) CHR$(31): PRINT "REMARKS"; TAB(22)
A$(8,W): GOSUB 620: IF G$="Y" THEN 610ELSE PRINT
600 INPUT "WHAT ARE THE REMARKS "; A$(8,W): PRINT V$
610 GOTO 100
620 G$="Y": INPUT "IS THIS CORRECT (Y OR N) "; G$: RETURN
630 Q$="N": PRINT @ 896, "THIS SETS UP A NEW FILE AND WILL
DESTROY ALL PREVIOUS FILE(S)"
640 FOR Z=1 TO 20: PRINT @ 929, STRING$(7,32): FOR ZZ=1 TO 15:
NEXT ZZ: PRINT @ 929, "DESTROY": FOR ZZ=1 TO 20: NEXT ZZ: NEXT Z
650 INPUT "DO YOU WISH TO DESTROY ALL FILE(S) IN MEMORY (Y OR N)
";Q$: IF LEFT$(Q$,1)="N" THEN 100ELSE L=0: W=0: Q=0: R=0: P=0:
O=0: I=0: T=0: X=0: Z=0: K=0: A=0: D=0: GOTO 420
660 GOSUB 960: GOSUB 300: GOTO 100
670 GOSUB 960
680 H=0: CLS: PRINT "INDEX MENU"
```

continued on page 54

RECIPE LOCATOR

James Scalise

Here is a simple to use, menu driven system I wrote for my wife to help her remember where her favorite recipes are located. She has about two dozen recipe books which she browses and when she finds an interesting recipe, she marks the page. Later, however, she doesn't remember which book contained a particular recipe. With this system, she is able to store the recipe title, book title and page number for later recall. She can save information for 55 recipes in each of 11 categories. Table 1 lists the categories.

BREAD	VEGETABLES
BEEF	PASTA
PORK	FISH
POULTRY	SALADS
LAMB	DESSERTS
SOUPS	

TABLE 1: Recipe Categories

As I said before, the system is completely menu driven. After loading the diskette, all the operator need do is press the reset button and enter the date. A "BUILD" file has been created which will AUTOmatically load and run the Master Menu program.

USING THE MASTER MENU PROGRAM

This program's function is to let you choose which program to run. You will be shown the Master Menu which gives you three options.

- Option 0: end the session and return to BASIC.
- Option 1: load and run the Update Program (RECPUPDT/BAS)
- Option 2: load and run the Report Program (RECPREPT/BAS)

USING THE UPDATE PROGRAM

The first menu you will see is the Action Menu. You have 6 choices as follows:

0. Returns you to the Master Menu program.
 1. Allows you to add a recipe to the file.
 2. Allows you to change the recipe title, book title or page number.
 3. This option will flag a recipe for deletion. The recipe is not physically removed from the file, only flagged as such.
 4. Lookup allows you to find out what book and page number a recipe is located at.
 5. Initialization should be done once and only once prior to entering your recipes. When you choose this option, you will be given the opportunity to stop. A response of "N" will return you to the Action Menu. A "Y" will move blanks to every record of the file.
- The first time, each session, you choose an option which will access the file (option 1-4), the message "ONE MOMENT-BUILDING TABLES" is displayed. Table building will take 1 to 5 minutes depending on the size of the file.
- After picking options 1-4, you will be presented with the Major Category Recipe Menu. If you pick 2 (MEATS), you will also be presented with the Meat Menu. If you choose actions 2, 3, or 4, you will be pre-

sented with a menu for that category containing all the recipes in that category. A recipe title of all "X" means this recipe has been flagged for deletion. If a category has no recipes, "NO RECIPES IN THIS CATEGORY" will be displayed. If there are recipes, enter your choice to continue.

Add Mode:

You will be prompted to enter the recipe title (30 characters), book title (30 characters) and page number (max 9999). Hit ENTER to add the recipe to the file. If there is no room on the file, you will be told. A successful add will return you to the Action Menu.

Change Mode:

You will be shown the Edit Menu. This menu will show you the current values of each field. If you make changes, but decide not to keep them, use option 4. To save your changes, use option 0.

Delete Mode:

You will be shown the information in the record to be deleted and then asked if you really wish to delete it. Entering a "D" will flag the record for deletion. Any other key will bypass the delete. If you accidentally delete a record, any change will reactivate it.

Lookup Mode:

Use this option to locate your recipe. After selecting your choices off the menu screens, you will be shown the recipe title, book title and page number where the recipe can be found.

USING THE REPORT PROGRAM

This program will allow you to print all the recipes on the file or all recipes in a particular category. The Action Menu will allow you to route the report to video (option 1) or the printer (option 2). Option 0 will return you to the program menu.

After choosing the reporting method, you will be shown the Major Category Recipe Menu which is the same as in the Update Program with the addition of choice number 9, "ALL CATEGORIES." This option will give you a report, by category, of the whole file.

A CLOSER LOOK

If you're interested in how the system works, the following tables will help you understand the Update and Report programs.

UPDATE PROGRAM — RECPUPDT/BAS

FUNCTION	
Initialize tables and variables.	10-90
Loads file parameters table, CA	100-110
CA(1) = # records per block	
CA(2) = # bytes per record	
CA(3) = start position of less than pointer	
CA(4) = start position of greater than pointer	
CA(5) = length of record key	

200-210	CA(6) = max # of records on file Counter initialization.	2000-3030	variables for meat recipes. Reads records and prints report on printer or screen based on Action Menu choice.	
500-600	Display Action Menu.			
1000-1140	Add a record to the file.	4000	Closes file and returns control to program menu.	
2000-2510	Change a recipe record.	5500-5580	Gets a record from file and places it in variable A1\$.	
3000-3110	Delete a record.	8000-8200	Contains the logic to read the record with the next greater key than the key of the record previously read. Detects end-of-file condition (E=1 is end-of-file).	
4000	Close file and return to program menu.	9000-9530	Handles printer and screen headings.	
4400-4460	POKE variable V at address M.	10000-10050	Moves fields to variables.	
5000-5050	Gets next free record space on the file.	10500-10550	Checks for last key (LK) equal to current key (CK). If not equal, Major Recipe Category has changed and index to report title table is reset.	
5500-5580	Gets a record from file and places it in variable A1\$.			
6000-6540	Writes a record to the file.			
7000-7210	Contains logic to read the record asked for.			
	Uses routine at 5500 to do physical read.			
8000-8070	Reads the file sequentially.			
10000-10170	Displays Major Category Recipe Menu. Accepts input from screen. Builds file key in variable I\$.			
11000-11080	Displays Meat Menu. Accepts input from screen.			
13000	Clears variables T\$, BK\$, P\$.			
14000-14998	Handles the display of the 11 recipe menus.			
15000	Delay loop.	I\$	Alphanumeric	Recipe file key
16000-16030	Moves data from field variables to variable A\$ for output to the file.	GR	Integer	Greater pointer
16600-16610	Moves data from variable A1\$ to individual field variables.	LR	Integer	Lesser pointer
17000	Displays message "DATA ENTERED IS TOO LONG"	T\$	Alphanumeric	Recipe title
17500-17540	Take 2-byte string recipe number and translates it to an integer.	BK\$	Alphanumeric	Book title
18000-18400	Updates recipe title tables.	P\$	Alphanumeric	Page number
20000-20070	Initialize CA(6) records to spaces.			Record length = 73
21000-21030	Opens file and sets buffer B\$.			Records per block = 3
22000-22010	Displays "no record found" message.			# records in file = 605
23000-23700	Build tables with the recipe titles.			
24000-24010	Displays "NO MORE ROOM" message.			The first record in the file contains a pointer to the next available record space.
24100-24310	Routines for PEEK and POKE. variables for meat recipes.			

REPORT PROGRAM — RECPREPT/BAS

LINES	FUNCTION
10-210	Initialize tables and variables. Open file RECIPE/DAT on drive 0.
500-560	Displays program's Action Menu and accepts input.
700-720	Prompts operator to READY PRINTER.
800-940	Displays Major Category Recipe Menu and accepts input. Prepares starting key (FK\$) and ending key (EK\$) variables.
950-1030	Displays Meat Menu and accepts input. Prepares starting key and ending key

Variable	Start	Length	Type	Description
I\$	1	4	Alphanumeric	Recipe file key
GR	5	2	Integer	Greater pointer
LR	7	2	Integer	Lesser pointer
T\$	9	30	Alphanumeric	Recipe title
BK\$	39	30	Alphanumeric	Book title
P\$	69	4	Alphanumeric	Page number

Record length = 73
Records per block = 3
records in file = 605

The first record in the file contains a pointer to the next available record space.

MINIMUM HARDWARE REQUIREMENTS

- TRS 80 Model III
- 32K
- one disk
- printer is optional

NOTE:

The file structure is a binary tree type structure with pointers to the next greater and the lesser record. Because of this structure, I have not been able to design a program which will physically remove records flagged for deletion. You may design a program yourself or you may simply re-use those records flagged for deletion by updating them with a new recipe of the same category.

Create a DO FILE as follows:

```
BUILD DO <ENTER>
CLS <ENTER>
BASIC <ENTER>
1 <ENTER> (For number of files request)
<ENTER> (For MEMORY request)
MENU/BAS <ENTER>
<BREAK>
```

Activate at TRSDOS READY with AUTO DO DIR.

PROGRAM LISTING: MENU/BAS

```

1 REM ****MASTER MENU PROGRAM - MENU/BAS****
2 REM ****
3 REM ****
20 CLS:PRINTTAB(20)"RECIPE LOCATOR MASTER MENU":PRINT:PRINT
30 PRINTTAB(28)"0. STOP"
40 PRINTTAB(28)"1. UPDATE RECIPE FILE"
50 PRINTTAB(28)"2. RECIPE REPORT"
60 PRINT:PRINT
70 INPUT"ENTER YOUR CHOICE";A
80 ONA+1GOT0200,300,400:GOT020
200 PRINT@897,"END OF SESSION":END
300 RUN"RECPUPDT/BAS"
400 RUN"RECPREPT/BAS"

```

RECIPE UPDATE PROGRAM: RECPUPDT/BAS

```

1 REM ****
2 REM RECIPE UPDATE PROGRAM - RECPUPDT/BAS
3 REM ****
10 CLEAR7000:DIMCA(6):DEFINTA-Z
20 M1=55:M2=4
30 DIMC1$(M1),C2$(M1),C3$(M1),C4$(M1),C5$(M1),C6$(M1),C7$(M1)
35 DIMC8$(M2,M1),TT$(11),C1(8),C2(M2)
40 FS=0:AC$="A":DL$="D":X$="X":LB=0:BL$="" :DV=0
50 CH$="ENTER YOUR CHOICE":AD$="ADD A RECORD":DD$="DELETED"
55 AV$="ACTIVE":LU$="LOOK UP A RECIPE"
60 UPS$="UPDATE A RECORD":DE$="DELETE A RECORD"
65 C0$="HIT ENTER TO CONTINUE":NR$="NO RECIPES IN THIS CATEGORY"
70 FORI=1TO11:READTT$(I):NEXTI:GOT0100
80 DATABREAD RECIPE MENU,VEGETABLE RECIPE MENU,SOUP RECIPE MENU
83 DATAROLE RECIPE MENU,SALAD RECIPE MENU,PASTA RECIPE MENU
85 DATADESSERT RECIPE MENU,BEEF RECIPE MENU
90 DATAPoultry RECIPE MENU,PORK RECIPE MENU,LAMB RECIPE MENU
100 FORI=1TO6:READCA(I):NEXTI:GOT0200
110 DATA3,73,5,7,4,606
200 FORI=1TO8:C1(I)=0:NEXTI
210 FORI=1TO2:C2(I)=0:NEXTI
500 CLS:PRINTTAB(16)"RECIPE FINDER UPDATE-ACTION MENU"
510 PRINT:PRINT:A=99
520 PRINTTAB(20)"0. RETURN TO MASTER MENU"
530 PRINTTAB(20)"1. ADD RECIPE TO FILE"
540 PRINTTAB(20)"2. CHANGE A RECIPE"
550 PRINTTAB(20)"3. DELETE A RECIPE"
555 PRINTTAB(20)"4. LOOKUP A RECIPE"
560 PRINTTAB(20)"5. INITIALIZE FILE"
570 PRINT:PRINT
580 PRINTCH$::INPUT"":A
590 ONA+1GOT04000,1000,2000,3000,3005,20000
600 GOT0500
999 REM
1000 CLS:PRINTTAB(26)AD$:GOSUB15000
1010 IFFS=OTHENGOSUB21000
1020 GOSUB10000:IFMJ=2THENGOT01030
1025 IF C1(MJ)=M1THENGOSUB24000:GOT0500ELSEC1(MJ)=C1(MJ)+1:
RC=C1(MJ):GOT01040
1030 IF C2(MN)=M1THENGOSUB24000:GOT0500ELSEC2(MN)=C2(MN)+1:
RC=C2(MN)
1040 RC!=RC/10:R=INT(RC!)+48:RN$=CHR$(R)+" "
1043 IF RC<9THENRN2=RC-((R-48)*10)+48:ELSER2=RC+48
1045 MID$(RN$,2,1)=CHR$(R2)
1046 IS=MJ$+MN$+RN$
1047 AS=I$+STRING$(69,32)
1050 CLS:PRINTTAB(26)AD$:PRINT:PRINT
1060 INPUT"ENTER RECIPE TITLE";T$:IFT$=""1060
1070 D=VARPTR(T$):GOSUB24100
1075 IF PEEK(D)>30THENGOSUB17000:GOT01050

```

```

1080 INPUT"ENTER BOOK TITLE";BK$:IFBK$=""1080
1090 D=VARPTR(BK$):GOSUB24100
1095 IF PEEK(D)>30THENGOSUB17000:GOT01080
1100 INPUT"ENTER PAGE NO. ";P$:IFP$=""1100
1110 D=VARPTR(P$):GOSUB24100
1115 IF PEEK(D)>4THENGOSUB17000:GOT01100
1120 ST$=AC$:GOSUB16010:GOSUB6000
1130 IFE<2THENGOT01140
1135 PRINT@897,"NO MORE ROOM ON FILE":GOSUB15000:GOSUB13000:
GOT0500
1140 GOSUB18000:GOT0500
1999 REM
2000 CLS:PRINTTAB(24)UP$:GOSUB15000
2010 IFFS=OTHENGOSUB21000
2020 GOSUB10000:IFE=2GOT0500ELSEA$=I$+STRING$(69,32):E=0:
GOSUB7000
2030 IFE=1THENGOSUB22000:GOT02000ELSEGOSUB16600
2040 CLS:PRINTTAB(24)UP$:B=99
2050 PRINTTAB(27)"EDIT MENU":PRINT:PRINT
2055 PRINTTAB(10);ENTER";TAB(17);TO CHANGE";TAB(30);CURRENT
VALUE"
2060 PRINTTAB(14)"0. END AND SAVE CHANGES"
2070 PRINTTAB(14)"1. RECIPE TITLE";TAB(30);T$
2080 PRINTTAB(14)"2. BOOK TITLE";TAB(30);BK$
2090 PRINTTAB(14)"3. PAGE NO. ";TAB(30);P$
2100 PRINTTAB(14)"4. QUIT AND DO NOT UPDATE FILE"
2105 PRINT:PRINTCH$::INPUT"":B
2110 ONB+1GOT02200,2310,2410,2510,500
2120 GOT02040
2200 ST$=AC$:GOSUB16000:GOSUB18000
2210 GOSUB5500:B1$=B$
2220 MID$(B1$,CA(2)*(RN-1)+1,CA(2))=A$
2230 LSETB$=B1$:PUT1,BN:GOT0500
2310 INPUT"ENTER NEW RECIPE TITLE";T$:GOSUB15000:GOT02040
2410 INPUT"ENTER NEW BOOK TITLE";BK$:GOSUB15000:GOT02040
2510 INPUT"ENTER NEW PAGE NO. ";P$:GOSUB15000:GOT02040
2999 REM
3000 CLS:PRINTTAB(25)DE$:GOSUB15000:GOT03010
3005 CLS:PRINTTAB(25)LU$:GOSUB15000
3010 IFFS=OTHENGOSUB21000
3030 GOSUB10000:IFE=2GOT0500
3034 E=0:A$=I$+STRING$(69,32):GOSUB7000
3035 IFE=1THENGOSUB22000:GOT03032ELSEGOSUB16600
3040 CLS:IFA=3PRINTTAB(25)DE$ELSEPRINTTAB(25)LU$:GOT03060
3050 PRINT@142,"THE FOLLOWING RECIPE WILL BE ";DD$
3060 PRINT@270,"TITLE":PRINT@280,T$:PRINT@334,"BOOK:"
3061 PRINT@344,BK$:PRINT@398,"PAGE":PRINT@408,P$
3062 PRINT@462,"STATUS":IFST$=DL$THENPRINT@472,DD$ELSE
PRINT@472,AV$
3065 IFA=4PRINT@897,C0$::INPUT"":E$:GOT0500
3070 PRINT@833,"ENTER 'D' TO DELETE"
3080 PRINT@897,"OR ANY KEY TO BYPASS DELETE";INPUT"":C$,
3090 IFC$<DL$THEN500
3100 ST$=DL$:GOSUB16000:D=VARPTR(T$):GOSUB24100
3105 AR=30:V=68:GOSUB=4400:GOSUB=18000
3110 GOT02210
3999 REM
4000 CLOSE1:RUN"MENU/BAS"
4399 REM
4400 S!=PEEK(D+1)+256*PEEK(D+2)
4410 GOSUB24200
4420 FORJ=1TOAR:M!=S+J-1
4430 GOSUB24300
4440 POKEM,V
4450 NEXTJ
4460 RETURN
4999 REM
5000 GET1,1
5010 FR=CVI(LEFT$(B$,2))+1

```

```

5020 B1$=B$
5030 MID$(B1$,1,2)=MKI$(FR)
5035 LSETB$=B1$
5040 PUT1,1
5050 RETURN
5499 REM
5500 BN=INT(RR/CA(1))
5510 RN=RR-BN*CA(1)+1
5520 BN=BN+1
5530 GET1,BN
5540 A1$=MID$(B$,CA(2)*(RN-1)+1,CA(2))
5560 GR=CVI(MID$(A1$,CA(3),2))
5570 LR=CVI(MID$(A1$,CA(4),2))
5580 RETURN
5999 REM
6000 GOSUB5000
6010 IFFR>1THEN6100
6015 A1$=A$+STRING$(CA(2),32)
6020 A$=LEFT$(A1$,CA(2))
6022 MID$(A$,CA(3),2)=MKI$(0)
6024 MID$(A$,CA(4),2)=MKI$(0)
6030 RR=FR
6040 GOSUB5500
6050 B1$=B$
6060 MID$(B1$,CA(2)*(RN-1)+1,CA(2))=A$
6070 GOSUB5035
6090 E=0:RETURN
6100 RR=1
6102 IFFR<=CA(6)THEN6110
6104 E=2:RETURN
6110 GOSUB5500
6120 IFLEFT$(A$,CA(5))<LEFT$(A1$,CA(5))THEN6200
6139 IFLEFT$(A$,CA(5))>LEFT$(A1$,CA(5))THEN6300
6140 E=1
6142 GET1,1
6144 B1$=B$
6146 MID$(B1$,1,2)=MKI$(FR-1)
6148 GOSUB5035
6152 RETURN
6200 IFLR=OTHEN6240
6210 RR=LR
6220 GOT06110
6240 MID$(A1$,CA(4),2)=MKI$(FR)
6250 GOSUB6500
6290 GOT06400
6300 IFGR=OTHEN6340
6310 RR=GR
6320 GOT06110
6340 MID$(A1$,CA(3),2)=MKI$(FR)
6350 GOSUB6500
6400 RR=FR
6420 GOSUB5500
6430 A$=A$+STRING$(CA(2),32)
6440 A1$=LEFT$(A$,CA(2))
6442 MID$(A1$,CA(3),2)=MKI$(0)
6444 MID$(A1$,CA(4),2)=MKI$(0):GOSUB6500:E=0:RETURN
6499 REM
6500 B1$=B$
6510 MID$(B1$,CA(2)*(RN-1)+1,CA(2))=A1$
6520 LSETB$=B1$
6530 PUT1,BN
6540 RETURN
6999 REM
7000 RR=1
7010 GOSUB5500
7020 IFLEFT$(A$,CA(5))<LEFT$(A1$,CA(5))THEN7100
7030 IFLEFT$(A$,CA(5))>LEFT$(A1$,CA(5))THEN7200
7040 E=0
7050 RETURN
7100 IFLR=OTHENE=1:RETURN
7110 RR=LR:GOT07010
7200 IFGR=OTHENE=1:RETURN
7210 RR=GR:GOT07010
7999 REM
8000 IFRR>CA(6)THENE=1:RETURN
8010 BN=INT(RR/CA(1))
8020 RN=RR-BN*CA(1)+1
8030 BN=BN+1
8040 IFBN<=LBTHENGET1,BN:LB=BN
8050 A1$=MID$(B$,CA(2)*(RN-1)+1,CA(2))
8060 IFLEFT$(A1$,1)=CHR$(32)THENE=1ELSE=0
8070 RETURN
9999 REM
10000 CLS:PRINTTAB(19)"MAJOR CATEGORY RECIPE MENU"
10010 PRINT:PRINT:MJ$="0"
10020 PRINTTAB(27)"1. BREADS"
10030 PRINTTAB(27)"2. MEATS"
10040 PRINTTAB(27)"3. VEGETABLES"
10050 PRINTTAB(27)"4. SOUPS"
10060 PRINTTAB(27)"5. CASSEROLES"
10070 PRINTTAB(27)"6. SALADS"
10080 PRINTTAB(27)"7. PASTA"
10090 PRINTTAB(27)"8. DESSERTS":PRINT
10100 PRINTCH$;:INPUT"":MJ$
10110 MJ=ASC(MJ$):MJ=MJ-48
10120 IFMJ<10RMJ>8THEN1000
10130 MN$="0":IFMJ=2THENGOSUB11000:GOT010150
10140 MN=ASC(MN$):MN=MN-48
10145 IFC1(MJ)=0THENGOSUB14997:RETURNSEGOT010155
10150 IFC2(MN)=0THENGOSUB14997:RETURN
10155 IFA=1THENRETURNSEGOT014000
10160 I$=MJ$+MN$+RN$
10170 RETURN
10990 REM
11000 CLS:PRINTTAB(27)"MEAT MENU":PRINT:PRINT
11010 PRINTTAB(27)"1. BEEF"
11020 PRINTTAB(27)"2. POULTRY"
11030 PRINTTAB(27)"3. PORK"
11040 PRINTTAB(27)"4. LAMB":PRINT
11050 PRINTCH$;:INPUT"":MN$
11060 MN=ASC(MN$):MN=MN-48
11070 IFMN<10RMN>4THEN11000
11080 RETURN
12999 REM
13000 TS=""":BK$=""":P$=""":RETURN
13999 REM
14000 CLS:C=0:E=0:CP=142:SW=0
14010 ONMJGOT014100,14500,14150,14200,14250,14300,14350,14400
14100 TX=1:GOSUB14970
14110 FORX=1TOC1(MJ):BR$=C1$(X):GOSUB14800
14120 IFE=1THENRETURN
14130 NEXTX:GOT014000
14150 TX=2:GOSUB14970
14160 FORX=1TOC1(MJ):BR$=C2$(X):GOSUB14800
14170 IFE=1THENRETURN
14180 NEXTX:GOT014000
14200 TX=3:GOSUB14970
14210 FORX=1TOC1(MJ):BR$=C3$(X):GOSUB14800
14220 IFE=1THENRETURN
14230 NEXTX:GOT014000
14250 TX=4:GOSUB14970
14260 FORX=1TOC1(MJ):BR$=C4$(X):GOSUB14800
14270 IFE=1THENRETURN
14280 NEXTX:GOT014000
14300 TX=5:GOSUB14970
14310 FORX=1TOC1(MJ):BR$=C5$(X):GOSUB14800
14320 IFE=1THENRETURN
14330 NEXTX:GOT014000

```

```

14350 TX=6:GOSUB14970
14360 FORX=1TOC1(MJ):BR$=C6$(X):GOSUB14800
14370 IFE=1THENRETURN
14380 NEXTX:GOTO14000
14400 TX=7:GOSUB14970
14410 FORX=1TOC1(MJ):BR$=C7$(X):GOSUB14800
14420 IFE=1THENRETURN
14430 NEXTX:GOTO14000
14500 ONMNGOTO14510,14550,14600,14650
14510 TX=8:GOSUB14970
14520 FORX=1TOC2(MN):BR$=C8$(1,X):GOSUB14800
14530 IFE=1THENRETURN
14540 NEXTX:GOTO14000
14550 TX=9:GOSUB14970
14560 FORX=1TOC2(MN):BR$=C8$(2,X):GOSUB14800
14570 IFE=1THENRETURN
14580 NEXTX:GOTO14000
14600 TX=10:GOSUB14970
14610 FORX=1TOC2(MN):BR$=C8$(3,X):GOSUB14800
14620 IFE=1THENRETURN
14630 NEXTX:GOTO14000
14650 TX=11:GOSUB14970
14660 FORX=1TOC2(MN):BR$=C8$(4,X):GOSUB14800
14670 IFE=1THENRETURN
14680 NEXTX:GOTO14000
14700 IFSW=0THENRETURN
14710 FORI=CPT0782STEP64
14720 PRINT@I,BL$:NEXTI:RETURN
14800 GOSUB14980:IFC>10HENGOSUB14950:C=0:CP=142:SW=1:RETURN
14810 IFMJ<2ANDX=C1(MJ)HENGOSUB14700:GOSUB14950
14820 IFMJ=2ANDX=C2(MN)HENGOSUB14700:GOSUB14950
14830 RETURN
14950 RN$=X$:PRINT@833,CO$
14955 PRINT@897,"OR ";CH$::INPUT"';RN$
14960 IFRN$=X$THENE=0:RETURN:ELSED=VARPTR(RN$):GOSUB17500:E=1
14965 IFRC<10RRC>XGOTO14950
14967 IFRC<10HENRN$=CHR$(48)+RN$
14968 RETURN
14970 PRINTTAB(23)TT$(TX):PRINT:RETURN
14980 PRINT@CP,X;".".;BR$:C=C+1:CP=CP+64:RETURN
14997 IFA<>1THENPRINT@833,NR$:PRINT@897,CO$::INPUT"';E$:E=2
14998 RETURN
15000 FORI=1TO700:NEXTI:RETURN
15999 REM
16000 A$=STRING$(CA(2),32)
16010 MID$(A$,1,4)=I$
16012 MID$(A$,9,30)=T$
16015 MID$(A$,5,2)=MKI$(GR):MID$(A$,7,2)=MKI$(LR)
16020 MID$(A$,39,30)=BK$:MID$(A$,69,4)=P$
16030 MID$(A$,73,1)=ST$:RETURN
16599 REM
16600 T$=MID$(A$,9,30):BK$=MID$(A$,39,30):I$=LEFT$(A$1$,4)
16610 P$=MID$(A$,69,4):ST$=MID$(A$,73,1):RETURN
17000 PRINT@897,"DATA ENTERED IS TOO LONG FOR THIS FIELD":GOSUB15000:RETURN
17500 GOSUB24100
17510 S!=PEEK(D+1)+PEEK(D+2)*256:GOSUB24200
17515 LN=PEEK(D):IFLN=1THENRC=PEEK(S)-48:RETURN
17520 FORJ=1TO2:M!=S+J-1:GOSUB24300
17530 IFJ=1THENRC=(PEEK(M)-48)*10ELSERC=RC+(PEEK(M)-48)
17540 NEXTJ:RETURN
17999 REM
18000 ONMJJGOT018050,18100,18150,18200,18250,18200,18350,18400
18050 C1$(RC)=T$:RETURN
18100 C8$(MN,RC)=T$:RETURN
18150 C2$(RC)=T$:RETURN
18200 C3$(RC)=T$:RETURN
18250 C4$(RC)=T$:RETURN
18300 C5$(RC)=T$:RETURN
18350 C6$(RC)=T$:RETURN
18400 C7$(RC)=T$:RETURN
19999 REM
20000 CLS:PRINT"** FILE INITIALIZATION NOW IN PROGRESS **"
20002 INPUT"ENTER 'Y' TO CONTINUE OR 'N' TO STOP";EN$
20004 IFEN$<>"N"ANDEN$<>"Y"THENGOTO20002
20006 IFEN$="N"THENGOTO500
20010 GOSUB21020
20020 B1$=STRING$(255,32)
20030 FORI=1TOCA(6)/CA(1)
20040 LSETB$=B1$:PUT1,I
20050 NEXTI
20055 MID$(B1$,1,2)=MKI$(0)
20060 LSETB$=B1$:PUT1,1
20065 CLOSE1
20070 GOTO500
20999 REM
21000 FS=1:GOSUB21020
21010 PRINT"ONE MOMENT -- BUILDING TABLES":GOSUB23000:CLOSE1
21020 OPEN"R",1,"RECIPE/DAT"
21030 FIELD 1,255 AS B$:RETURN
21999 REM
22000 PRINT@833,"NO RECORD FOUND"
22010 PRINT@897,CO$::INPUT"';E$:RETURN
22999 REM
23000 RR=1:IFCVI(LEFT$(B$,2))=0THENRETURN
23001 GOSUB8000:IFE=1THENRETURN:ELSET$=MID$(A$1$,9,30)
23002 IFRIGHT$(A$1$,1)=DL$THENEND=VARPTR(T$):GOSUB24100:AR=30:V=68:GOSUB4400
23005 D=VARPTR(A$):GOSUB24100:S!=PEEK(D+1)+PEEK(D+2)*256:GOSUB24200
23010 MJ=PEEK(S):MJ=MJ-48
23015 S!=S+1:GOSUB24200
23020 MN=PEEK(S):MN=MN-48
23030 S!=S+1:GOSUB24200:GOSUB17520
23035 IFMJ<2THENC1(MJ)=C1(MJ)+1ELSEC2(MN)=C2(MN)+1
23040 ONMJJGOT023200,23600,23250,23300,23350,23400,23450,23500
23200 C1$(RC)=T$:GOTO23700
23250 C2$(RC)=T$:GOTO23700
23300 C3$(RC)=T$:GOTO23700
23350 C4$(RC)=T$:GOTO23700
23400 C5$(RC)=T$:GOTO23700
23450 C6$(RC)=T$:GOTO23700
23500 C7$(RC)=T$:GOTO23700
23600 C8$(MN,RC)=T$
23700 RR=RR+1:GOTO23001
23999 REM
24000 PRINT@897,"NO MORE ROOM IN CATEGORY SELECTED"
24010 GOSUB15000:RETURN
24100 IFD>32767THEND=D-65536
24110 RETURN
24200 IFS>32767THENS!=S!-65536
24210 S=INT(S!):RETURN
24300 IFM!>32767THEMM!=M!-65536
24310 M=INT(M!):RETURN

```

RECIPE REPORT PROGRAM: RECPREPT/BAS

```

1 REM ****
2 REM RECIPE REPORT PROGRAM - RECPREPT/BAS
3 REM ****
10 CLEAR7000:DIMCA(6):DEFINTA-Z
20 M1=55:M2=4:PC=0:LC=51:SC=12:LK=0:FS=1
30 DIMT$(11)
35 POKE16424,67:POKE16425,0
40 AC$="A":DL$="D":S$="00":ED$="56"
45 CA$="RECIPE

```

BOOK

PAGE

```

50 CH$="ENTER YOUR CHOICE":DD$="DELETED":AV$="ACTIVE"
60 HA$="RECIPE LOCATOR SYSTEM":PG$="PAGE"
65 CO$="HIT ENTER TO CONTINUE":NR$="NO RECIPES IN THIS CATEGORY"
70 FORI=1TO11:READTT$(I):NEXTI:GOT0100
80 DATABREAD RECIPES,VEGETABLE RECIPES,SOUP RECIPES,CASSEROLE
RECIPES
85 DATASALAD RECIPES,PASTA RECIPES,DESSERT RECIPES,BEEF RECIPES
90 DATAPOULTRY RECIPES,PORK RECIPES,LAMB RECIPES
100 FORI=1TO6:READCA(I):NEXTI:GOT0200
110 DATA3,73,5,7,4,606
200 OPEN"R",1,"RECIPE/DAT"
210 FIELD1,255ASB$
500 CLS:PRINTTAB(16)"RECIPE LOCATOR REPORT-ACTION MENU":A=99:E=0
510 PRINT@212,"0. RETURN TO MASTER MENU"
520 PRINT@276,"1. REPORT TO VIDEO"
530 PRINT@340,"2. REPORT TO PRINTER"
540 PRINT@896,CH$;:INPUT"":A
550 ONA+1GOT04000,600,700
560 GOT0500
600 X0=1:GOT0800
700 CLS:PRINT@470,"READY PRINTER":PRINT@534,"THEN ";CO$;:INPUT;E$=
705 ST=PEEK(14312)AND240:IFST=48THENX0=2:GOT0800
710 CLS:PRINT@470,"PRINTER NOT READY -- READY PRINTER"
720 PRINT@534,"THEN ";CO$;:INPUT;E$:GOT0705
799 REM
800 CLS:PRINT@19,"MAJOR CATEGORY RECIPE MENU":MJ=99
810 PRINT@155,"1. BREADS"
820 PRINT@219,"2. MEATS"
830 PRINT@283,"3. VEGETABLES"
840 PRINT@347,"4. SOUPS"
850 PRINT@411,"5. CASSEROLES"
860 PRINT@475,"6. SALADS"
870 PRINT@539,"7. PASTA"
880 PRINT@603,"8. DESSERTS"
890 PRINT@667,"9. ALL CATEGORIES"
900 PRINT@896,CH$;:INPUT"":MJ$
910 MJ=ASC(MJ$):MJ=MJ-48
920 IFMJ=9THENFK$="1000":EK$="8055":MJ=1:GOT02000
930 IFMJ<10RMJ>8THENGOT0800
940 IFMJ>2THENFK$=MJ$+"0"+S$:EK$=MJ$+"0"+ED$:GOT02000
950 CLS:PRINT@27,"MEAT MENU":MN=99
960 PRINT@219,"1. BEEF"
970 PRINT@283,"2. POULTRY"
980 PRINT@347,"3. PORK"
990 PRINT@411,"4. LAMB"
1000 PRINT@896,CH$;:INPUT"":MN$
1010 MN=ASC(MN$):MN=MN-48
1020 IFMN<10RMN>4THENGOT0950
1030 FK$=MJ$+MN$+S$:EK$=MJ$+MN$+ED$
1999 REM
2000 A$=FK$
2005 GOSUB8000
2010 IFE=1THENPRINT@896,"REPORT COMPLETED      ":PRINT@960,CO$;:
INPUT";E$;FS=1:SC=12:LC=51:LK=0:PC=0:GOT0500
2020 GOSUB10000
2030 IFX0=1THENGOT03000
2040 IFLC>50THENGOSUB9000
2045 IFCK=>LKTHENPC=0:GOSUB9000
2050 LPRINTTAB(1)T$,TAB(3)BK$,TAB(2)P$;" ";ST$
2060 LC=LC+1:A$=A$1:GOT02005
2999 REM
3000 IFSC>11THENGOSUB9500
3005 IFCK=>LKTHENCOSUB9500
3010 PRINT"TITLE: ";T$;PRINT"BOOK: ";BK$
3020 PRINT"PAGE: ";P$;" SSTATUS: ";ST$
3030 PRINT:SC=SC+4:A$=A$1:GOT02005
3999 REM
4000 CLOSE1:RUN"MENU/BAS"
5499 REM

```

5500 BN=INT(RR/CA(1))
 5510 RN=RR-BN*CA(1)+1
 5520 BN=BN+1
 5530 GET1,BN
 5540 A1\$=MID\$(B\$,CA(2)*(RN-1)+1,CA(2))
 5560 GR=CVI(MID\$(A1\$,CA(3),2))
 5570 LR=CVI(MID\$(A1\$,CA(4),2))
 5580 RETURN
 7999 REM
 8000 A\$=A\$+STRING\$(CA(2),32)
 8005 MID\$(A\$,CA(5),1)=CHR\$(ASC(MID\$(A\$,CA(5),1))+1)
 8010 IFMJ=9THENX2\$=STRING\$(CA(5),255)ELSEA2\$=EK\$+STRING\$(CA(5),
 255)
 8020 RR=1
 8030 GOSUB5500
 8040 IFLEFT\$(A\$,CA(5))<LEFT\$(A1\$,CA(5))THEN8100
 8050 IFLEFT\$(A\$,CA(5))>LEFT\$(A1\$,CA(5))THEN8200
 8060 RETURN
 8100 IFLEFT\$(A1\$,CA(5))<LEFT\$(A2\$,CA(5))THENX2\$=A1\$
 8110 IFLR>0THEN8150
 8120 A1\$=A2\$
 8130 GOSUB8300
 8140 RETURN
 8150 RR=LR:GOT08030
 8200 IFGR>0THENRR=GR:GOT08030ELSEGOT08120
 8300 E=0
 8310 IFLEFT\$(A1\$,1)=CHR\$(255)THENE=1:GOT08330
 8320 IFLEFT\$(A1\$,4)=EK\$THENE=1
 8330 RETURN
 8999 REM
 9000 PC=PC+1:LC=0:GOSUB10500
 9010 LPRINTCHR\$(12):LPRINTTAB(30)HA\$,TAB(20)PG\$;PC
 9020 LPRINTTAB(33)TT\$(TX):LPRINT" "
 9030 LPRINTTAB(12)CA\$:LPRINT"":RETURN
 9499 REM
 9500 IFFS=1THENFS=0ELSEPRINT@960,CO\$;:INPUT"":E\$;
 9510 SC=0:GOSUB10500:CLS
 9520 PRINTTAB(21)HA\$:PRINTTAB(25)TT\$(TX)
 9530 PRINT:RETURN
 9999 REM
 10000 T\$=MID\$(A1\$,9,30):BK\$=MID\$(A1\$,39,30)
 10010 P\$=MID\$(A1\$,69,4)
 10020 IFMID\$(A1\$,73,1)=AC\$THENST\$=AV\$ELSEST\$=DD\$-
 10030 NA=ASC(LEFT\$(A1\$,1)):NA=NA-48
 10040 NB=ASC(MID\$(A1\$,2,1)):NB=NB-48
 10050 CK=(NA*10)+NB:RETURN
 10499 REM
 10500 IFCK=LKTHENGOT010550
 10510 LK=CK
 10520 IFNA=1THENTX=NA:GOT010550
 10530 IFNA>2THENTX=NA-1:GOT010550
 10540 TX=NB+
 10550 RETURN
 James Scalise
 10812 S. Homan Avenue
 Chicago, IL 60655 ■

IT STARTED WITH THE ABACUS continued from page 30

You may well be a game designer. Innovative games are, in fact, rather rare. You don't necessarily need to program one. But if you can describe it, perhaps flowchart it, you could sell an article to an interested magazine. Original thinkers are not always *programmers*. Programmers are not always original thinkers. So you might find that a collaboration with an established programmer would be profitable.

Maybe the ideas aren't coming. Don't worry! Immerse yourself in the problem and then go play a round of golf. Or a game of checkers. Or whatever. Give the ideas a chance to incubate. There is a good chance that brilliance will come through.

You could examine other aspects of the computer world. For example, how to sell. The market process is a new one to many programmers. If you are the expert in this field, let your light shine through. Explain to the poor programmer how to sell his masterpiece.

Are you a humorist? How many humorous articles have you seen in the computer magazines? Not that many! So you have an opportunity. Perhaps you might start off "Why does a computer cross the road?" and come up with a punch line. I cheerfully admit, I can't!

Do you write medical stories? Is the computer involved in your story plot? It might well be next time. And quite profitably at that. Why not a newly computerized administration section? And what happens when the computer dies? Who is the hero—the computer technician or the handsome new doctor? Some reader might want to know.

Of course, if you are writing science fiction then the computer is already old hat to you. but there are twists which haven't been used. And you may well be the one to use them. Oddly enough, many of the current advances have hopscotched over the imaginations of our writers. Can you expand their horizons?

Computers are here. They are here to stay. Our magazine racks, our bookshelves, our homes are their domain. Why fight them? Rather, take advantage of their capability, donate your skills to extending understanding them, and profit from their existence. To do so, learn the fundamentals of the field, the jargon, and let your imagination run wild. You may find your world, too, is the crazy, wonderful, exciting world of computers.

Frank Tymon
4749 W. K12 Avenue
Lancaster, CA 93534 ■

METRIC CONVERSION

continued from page 43

```

170 PRINT TAB(5);" 1 ... LENGTH
      2 ... AREA
      3 ... VOLUME
      4 ... WEIGHT": GOSUB 60
180 IF X=0 THEN 490 ELSE IF X>N THEN 160 ELSE ON X
GOSUB 200,210,220,230
190 GOTO 160
200 N=6:N1=0:CLS: PRINT @10,"LENGTH CONVERSION": GOSUB 130 :
IF X=0 THEN RETURN ELSE GOSUB 90 :GOTO 200
210 N=6:N1=3:CLS: PRINT @10,"AREA CONVERSION":GOSUB 130 :
IF X=0 THEN RETURN: ELSE GOSUB 90: GOTO 210
220 N=6:N1=6:CLS: PRINT @10,"VOLUME CONVERSION": GOSUB 130:
IF X=0 THEN RETURN ELSE GOSUB 90: GOTO 220
230 N=6:N1=9:CLS: PRINT @10,"WEIGHT CONVERSION": GOSUB 130:
IF X=0 THEN RETURN ELSE GOSUB 90: GOTO 230
490 GOSUB 65000: POKE 16420,0:END
499 ##### DATA #####
500 DATA 12
510 DATA KILOMETERS,.62137119,MILES
520 DATA METERS,3.28084,FEET
530 DATA CENTIMETERS,.3937008,INCHES
540 DATA SQ KILOMETERS,.386102,SQ MILES

```

```

550 DATA SQ METERS,10.76391,SQ FEET
560 DATA SQ CENTIMETERS,.1550003,SQ INCHES
570 DATA CUBIC KILOMETERS,.3861,CUBIC MILES
580 DATA CUBIC METERS,35.31467,CUBIC FEET
590 DATA CUBIC CENTIMETERS,.06102377,CUBIC INCHES
600 DATA METRIC TONS,1.1023113,TONS
610 DATA KILOGRAMS,2.204623,POUNDS
620 DATA GRAMS,.03527396,OUNCES

```

Hubbard C. Goodrich
RFD1, Box 880
South Harpswell, ME 04079 ■

HOUSEHOLD INVENTORY

continued from page 47

```

690 PRINT: PRINT TAB(10) "PARAMETERS": PRINT TAB(17) "( 1 ) ITEM"
700 PRINT TAB(17) "( 2 ) MANUFACTURER": PRINT TAB(17) "( 3 ) MODEL"
710 PRINT TAB(17) "( 4 ) SERIAL NUMBER": PRINT TAB(17) "( 5 ) DATE ACQUIRED"
720 PRINT TAB(17) "( 6 ) COST OR VALUE": PRINT TAB(17) "( 7 ) LOCATION"
730 PRINT TAB(17) "( 8 ) REMARKS"
740 PRINT @ 896, "ENTER SELECTION ( )"
750 $$=INKEY$: PRINT @ 914, CHR$(176):: FOR X=1 TO 200: NEXT: IF $$="" THEN PRINT @ 914, " ";: FOR X=1 TO 200: NEXT: GOTO 750ELSE PRINT @ 914, $$:: FOR X=1 TO 200: NEXT
760 ON VAL($$) GOTO 780, 780, 780, 780, 780, 780, 780: FOR Z=1 TO 20: PRINT @ 896, STRING$(15,32): FOR ZZ=1 TO 15: NEXT ZZ: PRINT @ 896, "INVALID REQUEST": FOR ZZ=1 TO 20: NEXT ZZ:NEXT Z
770 PRINT @ 896, "ENTER SELECTION": GOTO 750
780 I=VAL($$): IF I=1 THEN R$$="ITEM" ELSE IF I=2 THEN R$$="MANUFACTURER" ELSE IF I=3 THEN R$$="MODEL" ELSE
790 IF I=4 THEN R$$="SERIAL NUMBER" ELSE IF I=5 THEN R$$="DATE ACQUIRED" ELSE IF I=6 THEN R$$="COST OR VALUE" ELSE
800 IF I=7 THEN R$$="LOCATION" ELSE IF I=8 THEN R$$="REMARKS"
810 CLS: PRINT @ 0, "#";: PRINT @ 5, R$$;: PRINT @ 32, "#";: PRINT @ 37, R$$: PRINT STRING$(64,61)
820 0=128: FOR P=1 TO L: PRINT @ 0, MID$(STR$(P),2);": PRINT @ 0+5, A$(I,P):: 0=0+64: IF 0=960 THEN 0=160: ELSE IF 0=992 AND H=23 THEN 0=128: GOSUB 440: GOSUB 850: ELSE IF 0=992 THEN 0=128: GOSUB 840
830 IF P=L THEN GOSUB 840: GOTO 100ELSE NEXT P
840 IF H=23 THEN GOTO 440ELSE 0$$=INKEY$: PRINT @ 977, "PRESS ANY KEY TO CONTINUE ";: FOR X=1 TO 200: NEXT: PRINT @ 983, STRING$(7,32):: FOR X=1 TO 200: NEXT: IF 0$$="" THEN 840ELSE IF P=L THEN RETURN
850 CLS: PRINT @ 0, "#";: PRINT @ 5, R$$;: PRINT @ 32, "#";: PRINT @ 37, R$$: PRINT STRING$(64,61);: RETURN
860 GOSUB 960
870 FOR W=1 TO L: CLS: PRINT "FILE # "; MID$(STR$(W),2): PRINT: PRINT "ITEM": TAB(22) A$(1,W)
880 PRINT "MANUFACTURER": TAB(22) A$(2,W): PRINT "MODEL": TAB(22) A$(3,W)
890 PRINT "SERIAL NUMBER": TAB(22) A$(4,W): PRINT "DATE ACQUIRED": TAB(22) A$(5,W)
900 PRINT "COST OR VALUE": TAB(22) A$(6,W): PRINT "LOCATION": TAB(22) A$(7,W): PRINT "REMARKS": TAB(22) A$(8,W)
910 Z$$="N": PRINT: INPUT "DO YOU WISH THIS FILE DELETED (Y OR N) "; Z$$: IF LEFT$(Z$$,1)="Y" THEN 920ELSE 930

```

continued on page 61

DECODING PASSWORDS

Spencer Koenig

Okay, so you bought this great program for your TRS-80 Model III. It came in this little plastic bag that contained all kinds of dire warnings about how you couldn't return it if you opened the bag, and when you opened it you discovered another annoying fact: you couldn't back up the disk. The only way you could run this program was to use the diskette that it came on.

Now, ordinarily, that might not be so unrealistic. Even if you copied the program, you would probably put it on one diskette and use that over and over again. However, in this case that may not be such a good idea, because the diskette you received looked and felt pretty junky, and in order to save money, the company has provided you with a "flippy" diskette that had one program on one side and another on the other, even though each was referred to as a "separate" diskette.

This is a common frustrating situation for many purchasers of software. Manufacturers have resorted to these sorts of tactics to thwart software piracy, but the result is that legitimate purchasers of the software are inconvenienced. A good friend of mine recently had this experience with a series of programs that he purchased, and we became so angry that we decided to do something about it. Our investigations led to the development of the program accompanying this article. I have decided to publish it to help others avoid these frustrations.

BASIC REQUIREMENTS

The methods that we have developed here apply only to the TRS-80 Model III running under TRSDOS 1.3. In order to be able to extract the needed information off of the diskette, you will have to have some kind of a disk "zapping" program which can read and display the directory track of your diskette without going through the disk operating system itself. There are several such programs around, such as Super Utility 3.0 (available from Powersoft), SUPERZAP (available from Apparat on NEWDOS/80 version 2), but the one we used was Monitor #5, which is available together with the excellent book *TRS-80 Model III Assembly Language* by Hubert S. Howe, Jr. (who also is the Editor of this magazine) for \$29.95 (a steal). The book and monitor program are sold through a number of outlets, but if you can't find it, you can order it from Howe Software, 14 Lexington Road, New City, New York 1056, (914) 634-1821.

There are two basic methods you can employ to make a back-up copy of a password protected diskette. First, you can zap the disk so that you change the passwords to blank ones, and thus eliminate the password protection all together. If you prefer, you can change them back to the original passwords when you are finished, so that the diskette is restored to its original condition. The only problem with this method is that you have to change the diskette itself. The second method is to copy down the hash code for the passwords and run the accompanying program until you discover a possible password, and then use that to copy the files or back up the diskette. Either method will work, but both of them require that you have the original method of extracting the hash code in the first

place. The second method is perhaps the more elegant way to do it, but it certainly takes longer.

TECHINICAL INFORMATION

In order to understand passwords, you have to know about the *File Directory Entries* (FDEs) and *Hash Codes*. All the information about the files on the diskette is contained on the directory track, which consists mostly of File Directory Entries. FDEs are 48 bytes long, and they contain ALL the information about the file: its name, password, file attributes, creation date, number of records, record length, file extents, and so forth. A File Directory Entry for a file named TEST/BAS would appear as follows:

(Hexadecimal)	(ASCII)
1006 532A 0054 4553 5420 2020 2042 4153 ..S*.TEST BAS	
EF5C EF5C 0000 2201 FFFF FFFF FFFF FFFF 0.0...".....	
FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF	

The left column shows the contents of the FDE in hexadecimal form, and the right in ASCII. You can see the file name, TEST, and the extension, BAS, as well as many other things. The items that interest us are shown in bytes 17 through 20, at the left side of the second column: the hex digits "EF5C". The first of these is the update password, and the second is the access password. The code "EF5C" is for the blank password, or what a file has if it has no password.

In order to understand passwords, you have to know about "hash codes," which is the method according to which passwords are stored. "Hashing" describes a method for taking a character string and converting it into a number. The exact process is actually illustrated in the assembly language program shown below, but most people wouldn't be able to figure it out, and it doesn't really matter. There are numerous hashing methods, used for many different purposes. The surprising fact is that there are MANY different passwords that produce the SAME hash code. So if you can come up with ANY ONE of the passwords that produce the same value, you can have access to any program "protected" with that password. (Nifty, isn't it?)

FIRST METHOD: ZAP THE PASSWORD

The information presented above immediately suggests our first method of accessing a password-protected file: change the directory so that the four password bytes shown above read "EF5C EF5C." Using the Monitor #5 program described above, we have to use the Input Sectors (IS) command to read the directory, the Edit Memory (EM) command to alter it, and the Output Sectors (OS) command to write it back to the diskette.

The first thing to do, before anything else, is to make sure that the diskette is not write protected, so that it can be altered. (Make sure that there is no write-protect tab over the notch.) Then, run the *Monitor #5 program*, and insert the diskette in question into one of your disk drives. To read the directory, type:

IS17 1 18 8000

This will read the entire directory track into memory starting at address 8000H. We need to have the entire directory, because we don't yet know where the FDE in question is located.

Next, scroll through memory in order to find the FDE. This is most easily accomplished with the command:

DH8000 91FF

Use the SPACE bar to scroll one line at a time. When you find the directory, make note of the address that contains the SECOND line, containing the passwords. Suppose that you find the FDE starting at address 87C0: this line will contain the file name. The second line begins at 87D0, and in order to zap the passwords you will need to use the command:

EM87D0

Just type the information you want the data to be changed to: "EF (space) 5C EF 5C". Then, if you want to check it, do another "DH87D0".

When this has been successfully completed, write the directory track back to the diskette. This can be accomplished with the command:

OS17 1 18 8000

Now, return to TRSDOS and do a DIR command on the diskette in question. You will see that the files no longer have any passwords!

SECOND METHOD: FIND THE PASSWORD

The second method requires that you write down the password that you discover after executing the IS and DH commands above. Suppose that the FDE bytes read 72D7 72D7. Write these down, and then enter Disk Basic. In this case, you can see that both the update and access passwords have been used, which would be the case if the file were saved with a name including a password, such as "TEST/BAS.TEST". (In fact, 72D7 is the password for ".TEST".) The four digits 72D7 represent two bytes, and the least significant byte is stored first, so that the value of the number is D772H. The first thing to do is to discover the decimal value of this number, which you can do by executing the BASIC statement "? &HD772". The value is -10382. Enter this number into line 150 of the BASIC program shown below. If you have any other passwords you want to try, enter those into line 160 or anywhere before line 170.

Finally, you need to run the BASIC program shown below, but this requires that you also load in an assembly-language program and set memory size. Although the assembly-language program is shown in source code form, the BASIC program pokes this into memory, so all you have to do is to set memory size to 61439 and run the program, with line 150 altered to look for the password you want to find.

The BASIC program simply tests all possible passwords, from "AAAAAAA" to "ZZZZZZZ" and prints them on the line printer if they match. This program will take a VERY long time to go through all those possibilities. We ran it for over 24 hours and the first letter still did not change to "B"! However, in that time it had found numerous examples of the password we were looking for, and none of the combinations spelled a recognizable word. Nevertheless, any of the passwords could be used to retrieve the file.

The BASIC program has been written so that you can

very simply change it to look for a different string than "AAAAAAA" or "ZZZZZZZ". "A" is represented by the ASCII value 65, and "Z" by 90. If you want to start the first letter from "B", simply change 65 in line 50 to 66.

Once you have the password, you can copy the file, rename it, or do whatever you like. If you use this method, you can keep the original copy of the program you purchased in its original form, so that you can return it to the manufacturer in this condition, which is usually required if there is an update or new version. (Even when companies go to these lengths to protect programs, there are still errors in them!)

Best of luck with your password zapping, and happy hacking.

BASIC PROGRAM LISTING

```
10 'SET HIMEM TO 0FFFFH (61439) BEFORE RUNNING THIS
20 CLS : DEFINT A-Z: GOSUB 270
30 DEFUSR0=&HF000: L=&HF03A '-=4038
40 FOR I=1 TO 8: POKE L+I-1,32: NEXT I
50 FOR A=65 TO 90: POKE -4038,A
60 FOR B=65 TO 90: POKE -4037,B
70 FOR C=65 TO 90: POKE -4036,C
80 FOR D=65 TO 90: POKE -4035,D
90 PRINT CHR$(A);CHR$(B);CHR$(C);CHR$(D)
100 FOR E=65 TO 90: POKE -4034,E
110 FOR F=65 TO 90: POKE -4033,F
120 FOR G=65 TO 90: POKE -4032,G
130 FOR H=65 TO 90: POKE -4031,H
140 X=USR0(X)
150 IF X=-10382 THEN GOSUB 260
160 'IF ANOTHER PASSWORD, INSERT HERE
170 NEXT H
180 NEXT G
190 NEXT F
200 NEXT E
210 NEXT D
220 NEXT C
230 NEXT B
240 NEXT A
250 END
260 LPRINT    CHR$(A);CHR$(B);CHR$(C);CHR$(D);CHR$(E);CHR$(F);
CHR$(G);CHR$( H), X: RETURN
270 FOR I=&HF000 TO &HF03A: READ A: POKE I,A: NEXT I: RETURN
280 DATA 205,127,10,17,58,240,33,255,255,6,8,123,198,7,95
290 DATA 48,1,20,26,213,87,92,125,230,7,15,15,15,173,111
300 DATA 38,0,237,82,41,41,172,170,87,125,41,172,171,95,95
310 DATA 235,209,27,16,224,124,181,32,1,44,195,154,10,0
```

ASSEMBLY LANGUAGE PROGRAM LISTING

```
00100 ;SUBROUTINE TO RETURN BASIC HASH CODES
00110 ;SET MEMORY SIZE TO &HEFF
00120      ORG  0F000H
00130 ENTRY   CALL  0A7FH ;ARG TO HL
00140      LD   DE,BUF
00150      LD   HL,0FFFH
00160      LD   B,8
00170      LD   A,E
00180      ADD  A,7
00190      LD   E,A
00200      JR   NC,LOOP
00210      INC  D
```

continued on page 58

CASSIO — AN END TO THE CASSETTE BLUES

John Wilson

One of the nicest features of the Model III is the improved cassette transfer rate. The 1500 Baud rate is not only fast, but accurate. All is not perfect, however. All of the data (PRINT#, INPUT#) are still handled at 500 baud. Why would Radio Shack leave out such a much needed upgrade? What is worse is that, even with this upgrade, the data transfer would still be slow and tedious. A string variable cannot be written out if it contains delimiters (commas, colons, line feeds, etc.). The strings must be searched, and then any invalid characters changed. A long data base (such as in a BASIC word processor), would take too much time to be practical, even if 1500 baud was used. Wouldn't it be nice if someone could come up with a routine that would allow string characters containing ANY characters to be written or read at 1500 baud?

CASSIO (CASStette Input/Output routine) solves the problem! It will allow any string up to 255 characters to be written at either baud rate. The listing is in Basic and can be used in conjunction with your own programs.

BASIC LISTING

```
60000 MSB= :ST= :EN=
60005 FOR X=ST TO EN : READY : POKE X,Y : RETURN
60010 'Save A$ to cassette.
60020 POKE 16526,159:POKE 16527,MSB:RETURN
60030 'Load A$ to cassette.
60040 POKE 16526,211:POKE 16527,MSB:RETURN
60050 'Write.
60060 A=VARPTR(A$):X=USR(A):RETURN
60070 'Read.
60080 A$=STRING$(255,32):X=USR(VARPTR(A$)):RETURN
61000 DATA 205,127,10,197,213,245,70,35,94,35,86,205,201,255
61010 DATA 205,135,2,62,143,205,100,2,120,205,100,2,26,205,100
61020 DATA 2,19,16,249,205,248,1,241,209,193,195,154,10,62,0
61030 DATA 211,224,62,9,50,17,66,201,205,127,10,197,213,245,35
61040 DATA 94,35,86,43,43,205,150,2,205,53,2,250,143,32,15,205
61050 DATA 53,2,119,71,205,53,2,18,19,16,249,195,192,255,62,63
61060 DATA 50,63,61,195,192,255
```

Like any Basic program that uses a machine-language program, the machine code must be poked into high RAM. Be sure to set your memory size first! The memory sizes for 16K, 32K and 48K are 32670, 49054 and 65438 respectively. A GOSUB 60000 will set the code in RAM. The listing shown will work for either of the three memory sizes. Use the following chart for your specific machine:

Ram	Line 60000 should read:
16K	60000 MSB=127:ST=32671:EN=32767
32K	60000 MSB=191:ST=-16481:EN=-16385
48K	60000 MSB=255:ST=-97:EN=-1

Note: Be sure that you don't use the variables MS, ST or EN anywhere in your program!

The routine is called by the USR command, and therefore the entry point must be set accordingly. GOSUB's will be used to set the entry vectors and to read or write the data.

SAVING YOUR DATA

In order to set the USR vector, GOSUB 60020 must be executed. This will only need to be called when you want to write data, and the entry point is not set for the save. You will not need to do this every time that you want to save data.

The routine, as is, will write out the variable A\$. You could have any string variable written. To change the variable name, change line 60060 to suit. Please note that the variable must exist, or an FC error will result. GOSUB 60060 writes the variable and turns off the cassette after completion. For CASSIO, the baud rate is not changed by location 16913. Location 32704 (16K), 49088 (32K) or 65472 (48K) holds the cassette transfer rate. POKEing this with a zero (0) sets 500 baud, while a non-zero sets 1500 baud.

LOADING YOUR DATA

CASSIO is set up such that it will load only data files created by itself. If an attempt is made to read a non-CASSIO file, a question mark (?) will appear over the right asterisk in order to signal an error.

Like the save routine, the USR entry must be set accordingly. A GOSUB 60040 will perform this function. Again, this only needs to be called when the entry is not set for loading.

Line 60080 holds the load instruction. The variable A\$ is set to the maximum possible length (255 characters). The data is loaded from cassette, and the length set accordingly. Be sure that you have cleared at least 255 characters of string space before initially calling line 60000.

DEMO PROGRAMS

The following programs are intended to give you an idea of how CASSIO can be efficiently used. The program will allow you to enter a person's name, address and age, as a simulated data base.

```
10 CLS:CLEAR 1000:GOSUB 6000:R$=STRING$(40,32):GOSUB 60010
20 INPUT "Enter the Name ";N$
30 INPUT "Enter address ";AD$
40 INPUT "Enter the age ";A
50 'Set all 3 variables to 1 long string.
60 N$=LEFT$(N$+R$,20) 'Set name to 20 characters.
70 AD$=LEFT$(AD$+R$,30)'Set address to 30 chars.
80 Q$=MID$(STR$(A),2) 'Convert A to string Q$.
90 A$=N$+AD$+Q$           'Put 3 together.
100 GOSUB 60060:GOTO 20
110 'Be sure to add lines 60000 - 61060!
10 'Read names back in.
20 CLS:CLEAR 1000:GOSUB 60000:GOSUB 60040
30 INPUT "Press (ENTER) to load data ";C$
40 GOSUB 60050 'Read.
50 PRINT "Name: ";TAB(10);LEFT$(A$,20) 'Separate name.
60 PRINT "Address: ";TAB(10);MID$(A$,21,30)
70 PRINT "Age: ";TAB(10);MID$(A$,51)
80 PRINT:INPUT "Input another set of data? (Y/N) ";C$
90 IF C$="Y" THEN 30
100 IF C$<>"N" THEN 80
110 END
120 'Add lines 60000 - 61060!
```

The programs above are simplistic, but give a general idea of how the routine can be used. I hope that you will find CASSIO to be a valuable tool. Anyone who uses cassette data bases (or can everyone out there afford disk drives?) can surely take advantage of the increased time for data I/O.

SUMMARY

GOSUB	Function
60000	Initialize CASSIO. Don't use LS, ST or EN!
60020	Set USR entry for Save operation.
60050	Set USR entry for Load operation.
60060	Write A\$ to cassette.
60080	Read A\$ from cassette.

Memory location 32704 (16K), 49088 (32K) or 65472 (48K) contains the baud rate.

ASSEMBLER LISTING

```

00100    ORG    OFF9FH
00110 SAVE   CALL   0A7FH ;HL=VARPTR(ARG$)
00120      PUSH   BC    ;SAVE REGISTERS
00130      PUSH   DE
00140      PUSH   AF
00150      LD     B,(HL) ;B=LEN(ARG$)
00160      INC    HL
00170      LD     E,(HL) ;LSB OF ARG$
00180      INC    HL
00190      LD     D,(HL) ;MSB OF ARG$
00200      CALL   INIT   ;CLOCK OFF, CASS? H
00210      CALL   0287H ;WRITE LEADER & SYNC
00220      LD     A,143  ;IDENT. BYTE
00230      CALL   0264H ;WRITE
00240      LD     A,B    ;WRITE LEN(ARG$)
00250      CALL   0264H
00260 LOOP0  LD     A,(DE) ;A=MID$(ARG$,B,1)
00270      CALL   0264H ;WRITE CHAR
00280      INC    DE
00290      DJNZ   LOOP0  ;NEXT B
00300 EXIT   CALL   01F8H ;CASSETTE OFF
00310      POP    AF    ;RESTORE REGISTERS
00320      POP    DE
00330      POP    BC
00340      JP     0A98H ;RETURN TO BASIC
00350 INIT   LD     A,0    ;CLOCK OFF
00360      OUT   (224),A
00370      LD     A,9    ;1500 BAUD
00380      LD     (4211H),A
00390      RET
00400 LOAD   CALL   0A7FH ;HL=VARPTR(ARG$)
00410      PUSH   BC    ;SAVE REGISTERS
00420      PUSH   DE
00430      PUSH   AF
00440      INC    HL
00450      LD     E,(HL) ;LSB
00460      INC    HL
00470      LD     D,(HL) ;MSB
00480      DEC    HL
00490      DEC    HL
00500      CALL   0296H ;READ LEADER & SYNC
00510      CALL   0235H
00520      CP     143   ;IDENT BYTE

```

```

00530      JR     NZ,ERR
00540      CALL   0235H ;READ LEN(ARG$)
00550      LD     (HL),A ;ADJUST VARPTR
00560      LD     B,A    ;SET COUNTER
00570 LOOP1  CALL   0235H ;READ CHAR
00580      LD     (DE),A ;SAVE IN STRING
00590      INC    DE
00600      DJNZ   LOOP1  ;NEXT B
00610      JP     EXIT
00620 ERR    LD     A,'?' ;ERROR FLAG
00630      LD     (3C00H+63),A ;SAVE
00640      JP     EXIT
00650      END

```

John Wilson
204-79 Russell St. W.
Lindsay, Ontario
Canada K9V 2X3 ■

DECODING PASSWORDS

continued from page 56

```

00220 LOOP   LD     A,(DE)
00230      PUSH   DE
00240      LD     D,A
00250      LD     E,H
00260      LD     A,L
00270      AND   7
00280      RRCA
00290      RRCA
00300      RRCA
00310      XOR   L
00320      LD     L,A
00330      LD     H,O
00340      SBC   HL,DE
00350      ADD   HL,HL
00360      ADD   HL,HL
00370      XOR   H
00380      XOR   D
00390      LD     D,A
00400      LD     A,L
00410      ADD   HL,HL
00420      XOR   H
00430      XOR   E
00440      LD     E,A
00450      LD     E,A
00460      EX    DE,HL
00470      POP   DE
00480      DEC   DE
00490      DJNZ  LOOP
00500      LD     A,H
00510      OR    L
00520      JR     NZ,DONE
00530      INC    L
00540 ;
00550 DONE   JP     0A9AH ;RETURN ARG. TO PGM.
00560 ;
00570 BUF    DEFS  8
00580 END

```

Spencer Koenig
153-27 73rd Avenue
Flushing, NY 11367 ■

ADD A FANCY HORIZONTAL SCROLL TO YOUR PROGRAMS

Dan Keen and Dave Dischert

Here is a routine that you can easily add to your BASIC programs that will make them more exciting and dynamic.

This subroutine will scroll a screen full of whatever you want horizontally off the left hand side of the video display.

Game applications for this technique are endless. Even programs which use a "menu" can benefit from this trick. You can set up the screen with any graphics or text that you desire. In a menu situation, after the operator chooses a selection, simply execute this short subroutine and everything on the screen will hustle along horizontally until the entire contents have magically scrolled off the screen.

We only need a few lines of BASIC and a short machine language subroutine to achieve this neat effect.

Basically, the machine code moves every character on the screen to the character location preceding it. Naturally anything on the far left hand side of the screen that is moved to the next lower address will wrap back around and appear on the line above it at the right hand side of the screen.

Since we actually want to erase the data once it hits the left side of the screen, the easiest thing to do is to place a blank space at the beginning of each line on the video and then move every character back one character position. By doing this, we insure that once the data has been moved to the left, it appears as though it is dumped off the screen. The right hand side of the display will be black, with an increasingly large black margin as each blank space character is brought up by wrap around.

This is accomplished by lines 140 to 240 in the assembly language listing in figure one. There is a nasty trailing piece of data which follows along behind the scroll, located in the last video address, that is, the one at the bottom right hand corner of the display. Every time all of the characters are decremented, lines 250 and 260 stick a blank space in that location to get rid of the tag along character.

The machine routine does one advance, dresses up the screen with the proper "blanking" and then returns to BASIC.

Since there are 64 horizontal character positions, the BASIC program calls the routine 64 times by way of a USR command (see figure two). If the whole routine were in machine code, time delays would have to be added, otherwise it would scroll too fast for human eyes to catch.

It conveniently works out that the mere act of placing the USR call inside a FOR/NEXT loop provides the right delay action (never thought the slow speed of BASIC could be an advantage!).

In the BASIC program shown in figure two, the actual routine that you will need to incorporate into your own programs is located in lines 110 to 130. The rest of the program merely sets up the screen with some data on it to demonstrate this technique.

The listing in figure three gives the complete program

for 16K Level II BASIC users who do not have an editor assembler or who have not yet ventured into the obscure depths of assembly language programming.

Level II users should set MEMORY SIZE to 32511, and the DISK BASIC version should be set at 61399.

This routine can really spice up a program. It's worth considering in your next programming effort. Try it! You'll like it!

ASSEMBLY LANGUAGE SUBROUTINE LISTING — (for 48K Disk Basic).

```
00100 ;*****
00110 ;SCROLL VIDEO LEFT OFF SCREEN
00120 ;*****
00130     ORG    0F000H ;ORIGIN ADDRESS=DECIMAL 61440
00140     LD     A,32   ;LOAD BLANK SPACE TO REG. A
00150     LD     B,16   ;LOAD COUNT (16 LINES)
00160     LD     DE,64   ;LOAD DE WITH 64 (INCREMENT)
00170     LD     HL,15360 ;HL=1ST VIDEO ADDRESS
00180 LOOP   LD     (HL),A ;LOAD A INTO ADDRESS IN HL
00190     ADD   HL,DE   ;ADD 64 TO HL
00200     DJNZ  LOOP   ;FOR/NEXT LOOP-16 TIMES
00210     LD     HL,15361 ;HL=STARTING ADDRESS
00220     LD     DE,15360 ;DE=RECEIVING ADDRESS
00230     LD     BC,1024 ;BC=NO. OF BYTES TO MOVE
00240     LDIR      ;BLOCK MOVE
00250     LD     HL,16383 ;BOTTOMMOST RIGHT VIDEO
00260     LD     (HL),32   ;POKES BLANK TO VIDEO
00270     RET      ;RETURN TO BASIC PROGRAM
00280     END
```

BASIC PROGRAM LISTING

```
0 REM HORIZONTAL SCROLL OFF SCREEN
10 CLS
20 REM SETS UP A BORDER AROUND THE SCREEN & SOME TEXT IN
MIDDLE
30 FOR X=0 TO 127: SET(X,0): SET(X,47): NEXT
40 FOR Y=0 TO 47: SET(0,Y): SET(127,Y): NEXT
50 PRINT @ 470,"H & E COMPUTRONICS";
60 FOR X=1 TO 1000: NEXT
100 REM THIS SECTION CALLS THE MACHINE LANGUAGE SUBROUTINE
110 DEFUSR=&HFF00: FOR Y=1 TO 64
120 X=USR(0)
130 NEXT
```

Complete self-contained Level II BASIC version, 16K. Set memory size to 32511.

```
0 REM LEVEL II COMPLETE VERSION FOR HORIZONTAL VIDEO SCROLL
10000 POKE 16526,0: POKE 16527,127: REM DEFINE USR ADDRESS
10010 FOR X=0 TO 30
10020 READ A
10030 POKE 32512+X,A
10040 NEXT
```

continued on page 60

PERPETUAL CALENDAR

Gail D. Landis

This program will show the day of the week (i.e., Sunday, Monday, etc.) for any date desired beginning with 1753 through the year 2030. This calendar begins with the year 1753, since this was the year that Great Britain adopted our present "Gregorian Calendar."

Program lines 100 through 180 use data contained in lines 1000 and 1010 to set up array "Y" (table of years - 278). Program line 230 establishes "MM\$" (table of months) using data statement lines 1040 through 1080. (It should be noted that any of the 278 years fits one of the 14 different monthly arrangements.) This table is the key needed to find out when the first day of the month occurs for all months in the year requested. That is, if the year table shows a number 1 for the year being requested, this number will produce a string consisting of "BEEACFADGBEG." The letter to the extreme left is for January, the next for February, etc., through December. The letter "A" means that the first day for that month falls on Sunday, "B" the first for Monday, etc. through "F" for Saturday. Program lines 350 through 540 take the information from array MM\$ and prepare a calendar for the month requested (array D(6,7)). The rest of the program applies the date requested to the D(6,7) array to arrive at the "Weekday" answer being sought.

PROGRAM LISTING

```
10 DIM Y(279),MM$(14),D(6,7),DY$(7),MO$(12):C$=CHR$(23)
20 CLS:PRINT C$:PRINT@340,"CALENDAR":PRINT@404,"1753-2030"
30 PRINT:PRINT"I AM NOW SETTING-UP TABLES"
100 FORI=16TO31:READ T:Y(I)=T:Y(I+40)=T:Y(I+68)=T:Y(I+96)=T:
Y(I+136)=T:Y(I+164)=T:Y(I+192)=T:Y(I+220)=T:Y(I+248)=T:NEXT
110 FORI=4TO15:READ T:Y(I)=T:Y(I+28)=T:Y(I+68)=T:Y(I+96)=T:
Y(I+124)=T:Y(I+164)=T:Y(I+192)=T:Y(I+220)=T:Y(I+248)=T:NEXT
120 FORI=1TO3:READ T:Y(I)=T:Y(I+45)=T:Y(I+147)=T:NEXT
130 FORI=44TO45:READ Y(I):Y(I+96)=Y(I):NEXT
140 FORI=50TO51:READ Y(I):Y(I+95)=Y(I):NEXT
160 FORI=142TO144:READ Y(I):NEXT
170 FORI=52TO54:READ Y(I):NEXT
180 Y(147)=7:Y(279)=0:Y(49)=4:Y(55)=4:Y(151)=4
230 FORI=1TO14:READ MM$(I):NEXT
260 FORI=1TO7:READ DY$(I):NEXT
270 FORI=1TO12:READ MO$(I):NEXT
300 CLS:PRINT C$:PRINT"I WILL DISPLAY THE DAY OF THE"
310 PRINT"WEEK FOR ANY DATE BETWEEN THE"
320 PRINT"YEARS 1753-2030":PRINT:INPUT"YEAR <EX. 1753>";YR
330 INPUT"MONTH <1-12>";MO:INPUT"DAY <1-31>";DY:GOT0900
340 PRINT:PRINT"HANG ON--I'M FIGURING"
350 XX=YR-1752
360 MT=Y(XX):TA=1
370 MS=MM$(MT)
380 MS=MID$(MS,MO,1)
460 IFMS$="A"THEN TB=1:GOT0530:ELSE470
470 IFMS$="B"THEN TB=2:GOT0530:ELSE480
480 IFMS$="C"THEN TB=3:GOT0530:ELSE490
490 IFMS$="D"THEN TB=4:GOT0530:ELSE500
500 IFMS$="E"THEN TB=5:GOT0530:ELSE510
510 IFMS$="F"THEN TB=6:GOT0530:ELSE520
520 TB=7
530 X=X+1
```

```
540 FORI=TAT06:FORJ=TBT07:D(I,J)=X:X=X+1:NEXTJ:TB=1:TA=TA+1:
NEXTI
600 IFMO=2THEN650
610 IFMO=4 OR MO=6 OR MO=8 OR MO=11THEN960ELSE970
650 ZZ=yr/4:YY=ABS(ZZ-FIX(ZZ)):GOT0920
700 FORI=1TO6:FORJ=1TO7:IF D(I,J)=D(YR)THEN710ELSE720
710 D$=DY$(J):GOT0730
720 NEXTJ,I:GOT0800
730 M$=MO$(MO)
740 CLS:PRINT C$
750 PRINT"The WEEKDAY IS --";D$;"--"
760 PRINT:PRINT"FOR ";M$;DY";";YR:PRINT
770 INPUT"WANT ANOTHER <Y OR N>";H$:X=0
780 IFH$="Y"THEN790ELSEEND
790 FORI=1TO6:FORJ=1TO7:D(I,J)=0:NEXTJ,I:GOT0300
800 PRINT"CAN'T LOCATE DAY OF WEEK"
810 PRINT" MUST BE PROCESSING ERROR":END
900 IF YR<1753 OR YR>2030 THEN PRINT"YEAR IS OUT OF
BOUNDS-REDO":GOT0990
910 IF MO<1 OR MO>12 THEN PRINT"MONTH IS INCORRECT-REDO":GOT0990
915 X=0:GOT0340
920 IF YY>0 THEN930ELSE940
930 IF DY>28 THEN PRINT"THIS IS NOT LEAP YEAR-REDO":GOT0990
940 IF DY>29 THEN PRINT"FEB DOES NOT HAVE >29 DAYS-REDO":GOT0990
950 GOT0700
960 IF DY>30 THEN980ELSE700
970 IF DY>31 THEN980ELSE700
980 PRINT"NR OF. DAYS INCORRECT-REDO"
990 FORI=1TO1000:NEXT:X=0:GOT0300
1000 DATA 12,7,1,2,10,5,6,7,8,3,4,5,13,1,2,3,11,6,7,1,9,4
1010 DATA 5,6,14,2,3,4,1,2,3,12,7,5,6,1,2,10,14,2,3
1040 DATA"BEEACFADGBEG","CFFBDGCEACFA","DGGCEACFBDBG"
1050 DATA"EADFBDGCEAC","FBBEGCEADFBDB","GCCFADFBEGCE"
1060 DATA"ADDGBEGCFADF","BEFBDBGCEACFA","CFGCEACFBDBG"
1070 DATA"DGADEFBDGCEAC","EABEGCEADFBDB","FBCFADFBEGCE"
1080 DATA"GCDGBEGCFADF","ADEACFADGBEG"
1100 DATA"SUNDAY", "MONDAY", "TUESDAY", "WEDNESDAY", "THURSDAY",
"FRIDAY", "SATURDAY"
1110 DATA"JANUARY", "FEBRUARY", "MARCH", "APRIL", "MAY", "JUNE", "JULY"
1120 DATA"AUGUST", "SEPTEMBER", "OCTOBER", "NOVEMBER", "DECEMBER"
```

Gail D. Landis
9430 W. Kilner Road
Covington, OH 45318 ■

ADD A FANCY HORIZONTAL SCROLL

continued from page 59

```
10050 DATA 62,32,6,16,17,64,0,33,0,60,119,25,16,252,33,1,60,17
10060 DATA 0,60,1,0,4,237,176,33,255,63,54,32,201
10070 REM CALLS THE ROUTINE
10080 FOR Y=1 TO 64
10090 X=USR(0)
10100 NEXT
```

Dan Keen
Dave Dischert
Soft Horizons
RD 1, Box 432
Cape May Court House, NJ 08210 ■

HOUSEHOLD INVENTORY

continued from page 54

```

920 D$="N": INPUT "ARE YOU SURE (Y OR N) "; D$: IF
LEFT$(D$,1)="Y" THEN A$(1,W)="" : A$(2,W)="" : A$(3,W)="" :
A$(4,W)="" : A$(5,W)="" : A$(6,W)="" : A$(7,W)="" : A$(8,W)=""
930 NEXT W
940 R=0: FOR K=1 TO L: IF L=0 THEN R=R+1 ELSE FOR X=1 TO 8:
A$(X,K-R)=A$(X,K): NEXT X
950 NEXT K: L=L-R: A=L: GOTO 100
960 L=A: IF L=0 THEN 970ELSE A=L: RETURN
970 CLS: PRINT "NO FILE(S) ON RECORD": PRINT: PRINT: INPUT
"PRESS < ENTER > TO RETURN TO MENU ";EN$: GOTO 100
980 L=A: RETURN
990 GOSUB 960: CLS: INPUT "ENTER HOUSEHOLD NAME ";J$: INPUT
"ENTER TODAY'S DATE (MM/DD/YYYY) ";Z$: PRINT "PREPARE PRINTER
FOR OUTPUT OF CURRENT FILE(S)": PRINT "PRESS < ENTER > TO START
PRINTER"
1000 O$=INKEY$: IF O$="" THEN PRINT @ 200, STRING$(5,32);: FOR
X=1 TO 200: NEXT: PRINT @ 200, "ENTER";: FOR X=1 TO 200: NEXT:
GOTO 1000
1010 PRINT: PRINT: PRINT "PLEASE STANDBY .....": PRINT
"NOW PRINTING CURRENT FILE(S) ON PRINTER"
1020 LPRINT: LPRINT STRING$(79,61): LPRINT TAB(27) "RECORD OF
HOUSEHOLD ITEMS": LPRINT STRING$(79,61): LPRINT "UPDATED "Z$;
TAB(30)"HOUSEHOLD NAME -> J$: LPRINT
1030 FOR T=1 TO L: LPRINT TAB(10)"FILE # "; MID$(STR$(T),2):
LPRINT: LPRINT TAB(10)"ITEM"; TAB(40)A$(1,T)
1040 LPRINT TAB(10)"MANUFACTURER"; TAB(40) A$(2,T): LPRINT
TAB(10)"MODEL"; TAB(40) A$(3,T)
1050 LPRINT TAB(10)"SERIAL NUMBER"; TAB(40) A$(4,T): LPRINT
TAB(10)"DATE ACQUIRED"; TAB(40) A$(5,T)
1060 LPRINT TAB(10)"COST OR ESTIMATED VALUE"; TAB(40)A$(6,T):
LPRINT TAB(10)"LOCATION"; TAB(40)A$(7,T): LPRINT TAB(10)
"REMARKS"; TAB(40)A$(8,T): LPRINT
1070 NEXT T: GOTO 100
1080 GOSUB 960: CLS: PRINT "PLACE DATA TAPE IN RECORDER"
1090 PRINT: PRINT "PRESS RECORD/PLAY BUTTONS"
1100 PRINT: INPUT "PRESS < ENTER > TO START RECORDER "; EN$
1110 PRINT #-1,A
1120 FOR Q=1 TO A: PRINT #-1,A$(1,Q),A$(2,Q),A$(3,Q),
A$(4,Q),A$(5,Q),A$(6,Q),A$(7,Q),A$(8,Q)
1130 NEXT Q: GOTO 1140
1140 CLS
1150 PRINT " "; MID$(STR$(A),2) " RECORD(S) HAVE BEEN SAVED ON
TAPE": PRINT: INPUT "PRESS < ENTER > TO RETURN TO MENU "; EN$:
GOTO 100
1160 CLS: PRINT "PLACE DATA TAPE IN RECORDER"
1170 PRINT "PRESS PLAY BUTTON"
1180 INPUT "PRESS < ENTER > TO START RECORD(S) TRANSFER TO
COMPUTER "; EN$
1190 INPUT #-1,A: IF A=0 GOTO 1210ELSE 1200
1200 FOR Q=1 TO A: INPUT #-1,A$(1,Q),A$(2,Q),A$(3,Q),
A$(4,Q),A$(5,Q),A$(6,Q),A$(7,Q),A$(8,Q)
1210 NEXT Q: GOTO 1220
1220 CLS: PRINT " "; MID$(STR$(A),2) " RECORD(S) HAVE BEEN
LOADED": PRINT: INPUT "PRESS < ENTER > TO CONTINUE ";EN: GOTO
660
1230 CLS: PRINT "NO DATA ON TAPE": PRINT: PRINT: INPUT "PRESS <
ENTER > TO RETURN TO MENU ";EN: GOTO 100
1240 CLS: PRINT "YOU HAVE ELECTED TO < END > THIS PROGRAM":
PRINT

```

```

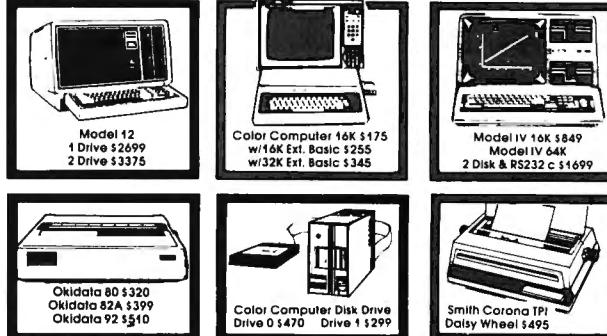
1250 INPUT "ARE YOU SURE (Y OR N) ";E$: IF LEFT$(E$,1)="Y" THEN
1260ELSE 1080
1260 CLS: INPUT "HAVE YOU SAVED YOUR FILE(S) (Y OR N) ";N$: IF
LEFT$(N$,1)="Y" THEN 1270ELSE 1080
1270 CLS: PRINT @ 535, "THANK YOU": PRINT: PRINT: END
1280 CLS: PRINT TAB(20)"I N S T R U C T I O N S": PRINT
1290 PRINT "THIS PROGRAM WILL INVENTORY 100 ITEMS. IF YOU WISH
TO ADD"
1300 PRINT "MORE CHANGE THE DIM(8,100) STATEMENT IN LINE 90 TO
WHAT YOU"
1310 PRINT "WANT. EXAMPLE > DIM(8,150) GIVES YOU 150 RECORDS.
WHEN THE"
1320 PRINT "PROGRAM STARTS YOU WILL BE PRESENTED WITH A MENU.
IN ORDER"
1330 PRINT "TO START THE PROGRAM YOU MUST PRESS THE LETTER 'C'
THIS"
1340 PRINT "WILL ERASE ANYTHING IN MEMORY AND WILL THEN START
WITH FILE # 1"
1350 PRINT "AND AS EACH ITEM IS RECORDED THE FILE #'S WILL
CHANGE."
1360 PRINT "JUST REMEMBER NOT TO USE ANY COMMAS (,) AS DATA WILL
NOT"
1370 PRINT "BE RECORDED OR SAVED AFTER THE COMMA."
1380 GOTO 400

```

William O. Byberg
103 Spit Brook Road
Bldg. F, Apt. 11
South Nashua, NH 03062 ■

From Computer Plus to YOU...

PLUS after PLUS after PLUS



BUY DIRECT

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

COMPUTERS		DISK DRIVES	
Model 12 40K 1 Drive	\$2699	R S Modem I.D.C.	129
Model 12 40K 2 Drive	3375	R S Modem I.D.C.	199
Model IV 16K	849	R S DC-1200	565
Model IV 64K	1699	Daisy Wheel II	1715
2 Disk & RS232 c	175	DWP-410	1320
Color Computer 16K	1699	Silver Reed EXP 550 D W	679
Color Computer 16	255	Smith Corona TPI Daisy Wheel	495
w/extended basic		Epson	ETC.
Color Computer 32	345	CGP-115	199
w/extended basic		DMP-400	315
Pocket Computer 2	165	DMP-120	410
Model 16 1DR 128K	4199	DMP-200	599
Model 16 2DR 128K	4799	DMP-400	1010
Model 100 8K	679	DMP-500	1539
Model 100 24K	835	DMP-2100	1779
MODEMS		Okidata 82A	399
Lynx Direct Connect Mi/Mi	235	Okidata 83A	655
Hayes Smart Modem II	235	Okidata 84 Parallel	999
Hayes Smart Modem 1200	565	Okidata 92	510
Novation Smartcall 1200	459	Okidata 93	859
Novation J-CAT	125	Gemini 10	319
R S Acoustic Coupler AC-3	129	Prowriter	129

TOLL FREE
1-800-343-8124

computer
plus

P O Box 926
480 King Street
Littleton, MA 01460
617-486-3193

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

Prices subject to change without notice
Not responsible for typographical errors
TRS-80 is a registered trademark of Tandy Corp

COMPUTRONICS CLASSIFIEDS

INNOVATIVE SIMULATION GAMES is our specialty—whether travel-, business-, or tennis-oriented. Write for information. Triangle Software, P.O. Box 58182, Raleigh, NC 27658.

Learn Radio Shack Cobol's built-in ISAM by studying source code for a 5-key mailing list and a 5-key article index, both with printing programs. Source code can be typed into TRS-80 I, II, III, 12, or 16 Cobol, then compiled. Receive all 24 pages of source code for \$10. Or send \$25 to get it on a Model III data disk. Check or money order to: Richard Bueche, 5704 Spring Valley #1056, Dallas TX 75240.

MIND SIDE OF COMPUTER KNOWLEDGE: thru Hypnosis complicated language/signs become second nature to you. Proven to work. Send \$19.95 H.T.I., Piney Point, MD 20674.

ASYNCHRONOUS SERIAL COMMUNICATIONS BOARD for IBM Personal Computer, never used, \$90.00. (914) 634-1821.

CHECKBOOK 2.2. Balances checkbook, Prints summary, Expenses for month, etc., Sorts too! Model I/III, \$14.95. Write to:

Gopher Software
4710 Valley View
Columbus, NE 68601

Learn Radio Shack Cobol's built-in ISAM by studying source code for a 5-key mailing list and a 5-key article index, both with printing programs. Source code can be typed into TRS-80 I, II, III, 12, or 16 Cobol, then compiled. Receive all 24 pages of source code for \$10. Or send \$25 to get it on a Model III data disk. Check or money order to: Richard Bueche, 5704 Spring Valley #1056, Dallas TX 75240.

TRS-80 Model 1, 2 drives, R.S. upper/lower case, doubler installed, green phosphor monitor, cassette recorder, dust covers, original manuals, excellent condition. Must sacrifice for \$1,295 or best offer. Call (914) 634-1406 after 5 P.M.

TRS-80 Model III Users! Make TRSDOS 1.3 work faster, better with Patch Sheet #1. \$3.00 postage paid. The Software Factory, 12101 Western View, Oklahoma City, OK 73132

DIABLO HyType I PRINTER, letter quality line printer with tractor feed attachment, full set of typewheels and ribbons, must sacrifice for \$1495 or best offer. Interfaces to expansion port of TRS-80 Model I. Word processing software included. Original cost over \$2300. Call (914) 634-1821 after 6:00 P.M.

PROGRAMS PUBLISHED IN Computronics on diskettes. Do you want to avoid typing in all those programs published each month in *Computronics*? For \$12.00, you can have a diskette containing all the programs published in any single issue *with corrections*. Specify the issue number you want and whether you want Model I or Model III media (not available for other models). Order from Box A, H & E Computronics, 50 North Pascack Road, Spring Valley, NY 10977, or call toll-free order number (800) 431-2818 outside New York state; inside New York, call (914) 425-1535. Add \$3.00 shipping and handling charge.

TRS-80 Model III Users! Make TRSDOS 1.3 work faster, better with Patch Sheet #1. \$3.00 postage paid. The Software Factory, 12101 Western View, Oklahoma City, OK 73132

Innovative Simulation Games is our specialty—whether travel-, business-, or tennis-oriented. Write for information. Triangle software, P.O. Box 58182, Raleigh, NC 27658.

Checkbook 2.2. Balances checkbook, Prints summary, Expenses for month, etc., Sorts too! Model I/III, \$14.95. Write to:
Gopher Software
4710 Valley View
Columbus, NE 68601

diskette contains the following programs: Regression Analysis with Confidence and Prediction Limits, Change baud rates on system tapes, PERT—Program Evaluation and Review Technique, Grid, and Monogram.

COMPUTRONICS Issue #53 (January 1983) program diskette contains the following programs: General Ledger Menu, Alpha Program, Roulette, Serious EDTASM, Surname Conversion, Jack and Jill, and Disk Checkbook Maintenance System.

Innovative Simulation Games is our specialty—whether travel-, business-, or tennis-oriented. Write for information. Triangle software, P.O. Box 58182, Raleigh, NC 27658.

Learn Radio Shack Cobol's built-in ISAM by studying source code for a 5-key mailing list and a 5-key article index, both with printing programs. Source code can be typed into TRS-80 I, II, III, 12, or 16 Cobol, then compiled. Receive all 24 pages of source code for \$10. Or send \$25 to get it on a Model III data disk. Check or money order to: Richard Bueche, 5704 Spring Valley #1056, Dallas TX 75240.

MIND SIDE OF COMPUTER KNOWLEDGE: thru Hypnosis complicated language/signs become second nature to you. Proven to work. Send \$19.95 to H. T. I., Piney Point, MD 20674.

COMPUTRONICS Issue #57 (May 1983) program diskette contains the following programs: Housekeeping programs UPDATE and YEAR (Month #5 of General Ledger package), 3-Across Mailing Labels, Service Territory and Manpower Planning Simulation program, Sword of Raschka game, Escape/Sketchit, Concentration, Tmult, Magic Squares, and Word Chase.

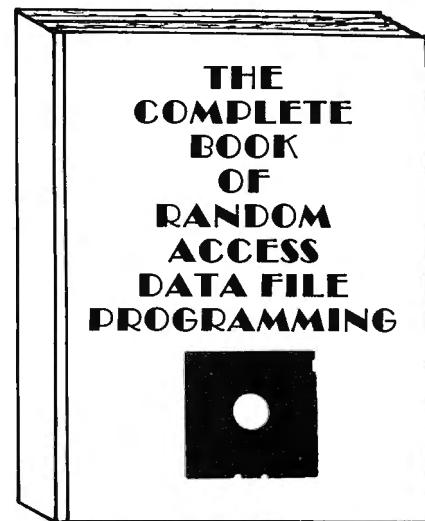
COMPUTRONICS Issue #58 (June 1983) program diskette contains the following programs: Trial Balance programs (Month #6 of General Ledger package), Chinese Zodiac, Easel program for drawing on screen in graphics.

The Complete Book Of Random Access Data File Programming

For TRS-80*, IBM Personal Computer*, Osborne*, and all Microsoft BASIC* computers

The last word on disk random access and file handling techniques, this series is intended for everyone — beginning programmers, businessmen and professionals will learn how to create custom programs to handle inventories, mailing lists, work scheduling, record keeping, and many other tasks, while more experienced programmers will learn advanced, professional programming techniques for faster, more efficient data storage and retrieval.

Although random access file handling is a matter of some complexity, the subject has been treated in a simple and down-to-earth fashion, so that anyone with some small familiarity with programming in Microsoft BASIC will be able to cope with the material. Each stage of learning uses a sample program as a starting point. The programs grow in capability and complexity as the books progress into all of the various aspects of file handling and record manipulation. An extensive effort has been made to keep the material coherent and every program line is explained in detail.



Volume I: Basic File Handling Techniques

- The writing of a Menu to Summarize program functions
- The writing of a screen format to accept data for records
- The creation of the basic record
- The FIELD and LSET routines for buffer preparation
- The writing of the record to disk in a random access mode
- The ability to change or edit a record
- The LPRINT capability from disk using three different formats
- Deleting a record from a random file
- Sorting the random file

- Searching the random file by name or key field
- The ability to search in a "NEXT" or "PRIOR" fashion
- The ability to purge deleted records from a disk file
- The ability to calculate with data from a disk file
- The provision for future expansion of the data fields
- The use of flags to prevent program crashes
- Date setting, printer on-line and many other routines to make a program run like a commercially-written program

Volume II: Advanced File Handling Techniques

- Blocking & de-blocking, Shell-Metzner sort, In-place screen editing, recovery of deleted record space
- Alpha-index record retrieval, fast machine/BASIC sort
- Linked list record structure and sort-merge, deleted record removal and file reorganization
- Multi-key file reorganization and record searching

- Relational database programming--comprehensive self-balancing accounting system with printouts
- Hashcoded data file manipulation--(probably the fastest method of data retrieval). Hashing the input key and recovery method explained
- Span-blocking techniques (allows; creation of records longer than 256 bytes without wasted space)

The Complete Book Of Random Access Data File Programming

Volume I: Basic File Handling Techniques	\$29.95
optional Vol. I Program Disk for Model I/III	\$28.50
optional Vol. I Program Disk for Model II	\$32.50
Volume II: Advanced File Handling Techniques	\$29.95
optional Vol. II Program Disks for Models I, II or III	\$49.95

COMPUTRONICS
INC.
50 N. PASCACK ROAD
SPRING VALLEY, NEW YORK 10977

AMERICAN
EXPRESS
24 HOUR
ORDER
LINE
(914) 425-1535

VISA
NEW TOLL-FREE
ORDER LINE
(OUTSIDE OF N.Y. STATE)
(800) 431-2818

ADD \$3.00 FOR SHIPPING IN UPS AREAS
ADD \$4.00 FOR C.O.D. OR NON-UPS AREAS
ADD \$5.00 TO CANADA AND MEXICO
ADD PROPER POSTAGE OUTSIDE OF U.S.,
CANADA AND MEXICO

*** ALL PRICES AND SPECIFICATIONS SUBJECT TO CHANGE ***
DELIVERY SUBJECT TO AVAILABILITY

— 30-DAY MONEY BACK GUARANTEE —



COMPUTRONICS^{INC.}

• • EVERYTHING FOR YOUR TRS-80™ • ATARI™ • APPLE™ • PET™ • CP/M™ • XEROX™ • IBM™ • OSBORNE™ • •

• • KAYPRO • •

* TRS-80 is a trademark of the Radio Shack Division of Tandy Corp. * ATARI is a trademark of Atari Inc. * APPLE is a trademark of Apple Corp. * PET is a trademark of Commodore
* CP/M is a trademark of Digital Research * XEROX is a trademark of Xerox Corp. * IBM is a trademark of IBM Corp. * OSBORNE is a trademark of Osborne Corp.



★ All orders processed within 24-Hours
★ 30-Day money back guarantee

BUSINESS PAC 100

100 Ready-To-Run Business Programs

(ON CASSETTE OR DISKETTE).....Includes 128 Page Users Manual.....
Inventory Control.....Payroll.....Bookkeeping System.....Stock Calculations.....
Checkbook Maintenance.....Accounts Receivable.....Accounts Payable.....

BUSINESS 100 PROGRAM LIST

NAME	DESCRIPTION
1 RULE78	Interest Apportionment by Rule of the 7's
2 ANNU1	Annuity computation program
3 DATE	Time between dates
4 DAYYEAR	Day of year a particular date falls on
5 LEASEINT	Interest rate on lease
6 BREAKEVN	Break-even analysis
7 DEPRSL	Straightline depreciation
8 DEPRSY	Sum of the digits depreciation
9 DEPRDB	Declining balance depreciation
10 DEPRDDB	Double declining balance depreciation
11 TAXDEP	Cash flow vs. depreciation tables
12 CHECK2	Prints NEBS checks along with daily register
13 CHECKBK1	Checkbook maintenance program
14 MORTGAGE/A	Mortgage amortization table
15 MULTMON	Computes time needed for money to double, triple, etc.
16 SALVAGE	Determines salvage value of an investment
17 RRVARIN	Rate of return on investment with variable inflows
18 RRCONST	Rate of return on investment with constant inflows
19 EFFECT	Effective interest rate of a loan
20 FVAL	Future value of an investment (compound interest)
21 PVAL	Present value of a future amount
22 LOANPAY	Amount of payment on a loan
23 REGWITH	Equal withdrawals from investment to leave 0 over
24 SIMPDFSK	Simple discount analysis
25 DATEVAL	Equivalent & nonequivalent dated values for oblig.
26 ANNDEF	Present value of deferred annuities
27 MARKUP	% Markup analysis for items
28 SINKFUND	Sinking fund amortization program
29 BONDVAL	Value of a bond
30 DEPLETE	Depletion analysis
31 BLACKSH	Black-Scholes options analysis
32 STOCVAL1	Expected return on stock via discounts dividends
33 WARVAL	Value of a warrant
34 BONDVAL2	Value of a bond
35 EPSEST	Estimate of future earnings per share for company
36 BETAALPH	Computes alpha and beta variables for stock
37 SHARPE1	Portfolio selection model-i.e. what stocks to hold
38 OPTWRITE	Option writing computations
39 RTVAL	Value of a right
40 EXPVAL	Expected value analysis
41 BAYES	Bayesian decisions
42 VALPRINF	Value of perfect information
43 VALADINF	Value of additional information
44 UTILITY	Derives utility function
45 SIMPLEX	Linear programming solution by simplex method
46 TRANS	Transportation method for linear programming
47 EOQ	Economic order quantity inventory model
48 QUEUE1	Single server queuing (waiting line) model
49 CVP	Cost-volume-profit analysis
50 CONDPROF	Conditional profit tables
51 OPTLOSS	Opportunity loss tables
52 FQUOQ	Fixed quantity economic order quantity model
53 FQEOWSH	As above but with shortages permitted
54 FQEOPB	As above but with quantity price breaks
55 QUEUECB	Cost-benefit waiting line analysis
56 NCFANAL	Net cash-flow analysis for simple investment
57 PROFIND	Profitability index of a project
58 CAP1	Cap. Asset Pr. Model analysis of project
59 WACC	Weighted average cost of capital
60 COMPBAL	True rate on loan with compensating bal. required
61 DISCBAL	True rate on discounted loan
62 MERGANAL	Merger analysis computations
63 FINRAT	Financial ratios for a firm
64 NPV	Net present value of project
65 PRINDLAS	Laspeyres price index
66 PRINPDA	Pasche price index
67 SEASIND	Constructs seasonal quantity indices for company
68 TIMETR	Time series analysis linear trend
69 TIMEMOV	Time series analysis moving average trend
70 FUPRINF	Future price estimation with inflation
71 MAILPAC	Mailing list system
72 LETWRIT	Letter writing system-links with MAILPAC
73 SORT3	Sorts list of names
74 LABEL1	Shipping label maker
75 LABEL2	Name label maker
76 BUSBUD	DOME business bookkeeping system
77 TIMECLK	Computerized weeks total hours from timeclock info.
78 ACCTPAY	In memory accounts payable system-storage permitted
79 INVOICE	Generate invoice on screen and print on printer
80 INVENT2	In memory inventory control system
81 TELDIR	Computerized telephone directory
82 TIMUSAN	Time use analysis
83 ASSIGN	Use of assignment algorithm for optimal job assign.
84 ACCTREC	In memory accounts receivable system-storage ok
85 TERMSPAY	Compares 3 methods of repayment of loans
86 PAYNET	Computes gross pay required for given net
87 SELLPR	Computes selling price for given after tax amount
88 ARBCOMP	Arbitrage computations
89 DEPRFS	Sinking fund depreciation
90 UPSZONE	Finds UPS zones from zip code
91 ENVELOPE	Types envelope including return address
92 AUTOEXP	Automobile expense analysis
93 INSFILE	Insurance policy file
94 PAYROLL2	In memory payroll system
95 DILANAL	Dilution analysis
96 LOANAFFD	Loan amount a borrower can afford
97 RENTPRCH	Purchase price for rental property
98 SALELEAS	Sale-leaseback analysis
99 RRCNVBD	Investor's rate of return on convertible bond
100 PORTVAL9	Stock market portfolio storage-valuation program

- TRS-80 Cassette Version \$99.95
- TRS-80 (Mod-I or III), Pet, Apple or Atari Versions \$99.95
- TRS-80 Mod-II, IBM, Osborne and CP/M Versions \$149.95

ADD \$3.00 FOR SHIPPING IN UPS AREAS
ADD \$4.00 FOR C.O.D. OR NON-UPS AREAS
ADD \$5.00 TO CANADA AND MEXICO
ADD PROPER POSTAGE OUTSIDE OF U.S., CANADA AND MEXICO

COMPUTRONICS^{INC.}
MATHEMATICAL APPLICATIONS INC.

50 N. PASCAK ROAD
SPRING VALLEY, NEW YORK 10977

ASK FOR OUR 64-PAGE CATALOG

DEALER INQUIRIES INVITED

NEW TOLL-FREE
ORDER LINE
(OUTSIDE OF N.Y. STATE)
(800) 431-2818



24 HOUR
ORDER
LINE

(914) 425-1535

ALL PRICES & SPECIFICATIONS SUBJECT TO CHANGE
DELIVERY SUBJECT TO AVAILABILITY



COMPUTRONICS^{INC.}

• EVERYTHING FOR YOUR TRS-80® • ATARI® • APPLE® • PET® • CP/M® •

* TRS-80 is a trademark of the Radio Division of Tandy Corp. - * ATARI is a trademark of Atari Inc. - * APPLE is a trademark of Apple Corp. - * PET is a trademark of Commodore - * CP/M is a trademark of Digital Research.

**100 SUPER
PROGRAMS**

MASTER PAC 100 2nd EDITION (COMPLETELY REVISED)

FOR YOUR TRS-80™ LEVEL II MICROCOMPUTER

ALL ON CASSETTE OR DISKETTE

BUSINESS AND PERSONAL FINANCE

- 1. CHECKBOOK MAINTENANCE
- 2. TIME FOR MONEY TO DOUBLE
- 3. FEDERAL FICA & WITHHOLDING TAX
- 3. COMPUTATIONS
- 4. HOME BUDGET ANALYSIS
- 5. ANNUITY COMPUTATION
- 6. UNIT PRICING
- 7. CHANGE FROM PURCHASE
- 8. NEBS CHECK PRINTER
- 9. DAYS BETWEEN DATES
- 10. MORTGAGE AMORTIZATION TABLE
- 11. INVENTORY CONTROL
- 12. PORTFOLIO VALUE COMPUTATIONS
- 13. VALUE OF A SHARE OF STOCK
- 14. SALES RECORD KEEPING SYSTEM
- 15. FUTURE VALUE OF AN INVESTMENT
- 16. EFFECTIVE INTEREST RATE (LOAN)
- 17. PRESENT VALUE OF A FUTURE AMOUNT
- 18. RATE OF RETURN-VARIABLE INFLOW
- 19. RATE OF RETURN-CONSTANT INFLOW
- 20. REGULAR WITHDRAWAL FROM INVESTMENT
- 21. STRAIGHT LINE DEPRECIATION
- 22. SUM OF DIGITS DEPRECIATION
- 23. DECLINING BALANCE DEPRECIATION
- 24. BREAK EVEN ANALYSIS
- 25. SALVAGE VALUE OF INVESTMENT
- 26. PAYMENT ON A LOAN
- 27. FUTURE SALES PROJECTIONS
- 28. CREDIT CARD FILE
- 29. ECONOMIC ORDER QUANTITY (EOQ)
- INVENTORY MODEL
- 30. VALUE OF HOUSE CONTENTS
- 31. TEXT EDITOR
- 32. MONTHLY CALENDAR
- 33. DAY OF WEEK
- 34. CASH FLOW VS. DEPRECIATION
- 35. COMPLETE MAIL SYSTEM
- 36. INTEREST RATE ON A LEASE

BUSINESS

STATISTICS AND MATHEMATICS

- 37. RANDOM SAMPLE SELECTION
- 38. ANGLO-METRIC CONVERSION
- 39. MEAN, STANDARD DEVIATION, MAXIMUM AND MINIMUM
- 40. SIMPLE LINEAR REGRESSION
- 41. MULTIPLE REGRESSION ANALYSIS
- 42. GEOMETRIC REGRESSION
- 43. EXPONENTIAL REGRESSION
- 44. SIMPLE MOVING AVERAGE
- 45. SIMPLE T-TEST
- 46. CHI-SQUARE TEST
- 47. NORMAL PROBABILITIES
- 48. BINOMIAL PROBABILITY
- 49. POISSON PROBABILITY
- 50. MATRIX ADDITION AND SUBTRACTION
- 51. MATRIX TRANPOSE
- 52. MATRIX INVERSE
- 53. MATRIX MULTIPLICATION
- 54. SOLUTION OF SIMULTANEOUS EQUATIONS
- 55. QUADRATIC FORMULA
- 56. LINEAR EQUATION SOLUTIONS
- 57. ROOT HALF INTERVAL SEARCH
- 58. ROOTS OF POLYNOMIALS
- 59. ROOTS-NEWTON'S METHODS
- 60. PRIME FACTORS OF INTEGER
- 61. LEAST COMMON DENOMINATOR
- 62. RADIAN-DEGREE CONVERSION
- 63. NUMERICAL INTEGRATION

STATISTICS

UTILITIES

- 64. QUICK SORT ROUTINE
- 65. PROGRAM STORAGE INDEX
- 66. MULTIPLE CHOICE QUIZ BUILDER
- 67. FORM LETTER WRITER
- 68. SHELL SORT
- 69. CASSETTE LABEL MAKER
- 70. CODES MESSAGES
- 71. MERGE TWO FILES
- 72. SORT WITH REPLACEMENT

MATH

GRAPHICS

- 73. DRAWS BAR GRAPH
- 74. DRAWS HISTOGRAM
- 75. MOVING BANNER DISPLAY

GAMBLING AND GAMES

- 76. RANDOM SPORTS QUIZ
- 77. GOVERNMENT QUIZ
- 78. HORSE RACE
- 79. MAGIC SQUARE
- 80. ARITHMETIC TEACHER
- 81. HIGH LOW GAMBLE
- 82. UNSCRAMBLE LETTERS
- 83. HANGMAN
- 84. GAME OF NIM
- 85. RUSSIAN ROULETTE
- 86. ROULETTE GAME
- 87. ONE-ARMED BANDIT
- 88. HIT THE TARGET
- 89. WALKING DRUNK
- 90. STATE CAPITAL QUIZ
- 91. TIC-TAC-TOE
- 92. DICE GAME
- 93. LUNAR LANDAR GAME
- 94. BIORHYTHM
- 95. HORSE SELECTOR (CLASS CALCULATOR)
- 96. RANDOM DICE ROLL
- 97. RANDOM ROULETTE ROLL
- 98. RANDOM CARD DEALER
- 99. GUESS THE NUMBER
- 100. WHITE OUT SCREEN

GAMBLING

**INCLUDES 110 PAGE
USER MANUAL**

GUARANTEED SATISFACTION

30-DAY MONEY BACK GUARANTEE ON ALL SOFTWARE

*** ALL PRICES AND SPECIFICATIONS SUBJECT TO CHANGE ***

COMPUTRONICS^{INC.}

MATHEMATICAL APPLICATIONS SERVICE
50 N. PASCACK ROAD
SPRING VALLEY, NEW YORK 10977

PLEASE SEND ME:

- MASTER PAC 100 CASSETTE VERSION \$ 99.95
- MASTER PAC 100 DISKETTE VERSION \$ 99.95
- MASTER PAC 100 (MODEL II DISKETTE VERSION) ... \$149.95



24 HOUR
ORDER
LINE
(914) 425-1535



NEW TOLL-FREE

ORDER LINE

(OUTSIDE OF N.Y. STATE)

(800) 431-2818

★ All orders processed within 24-Hours
★ 30-Day money back guarantee on all Software

CREDIT CARD NUMBER EXP. DATE

SIGNATURE

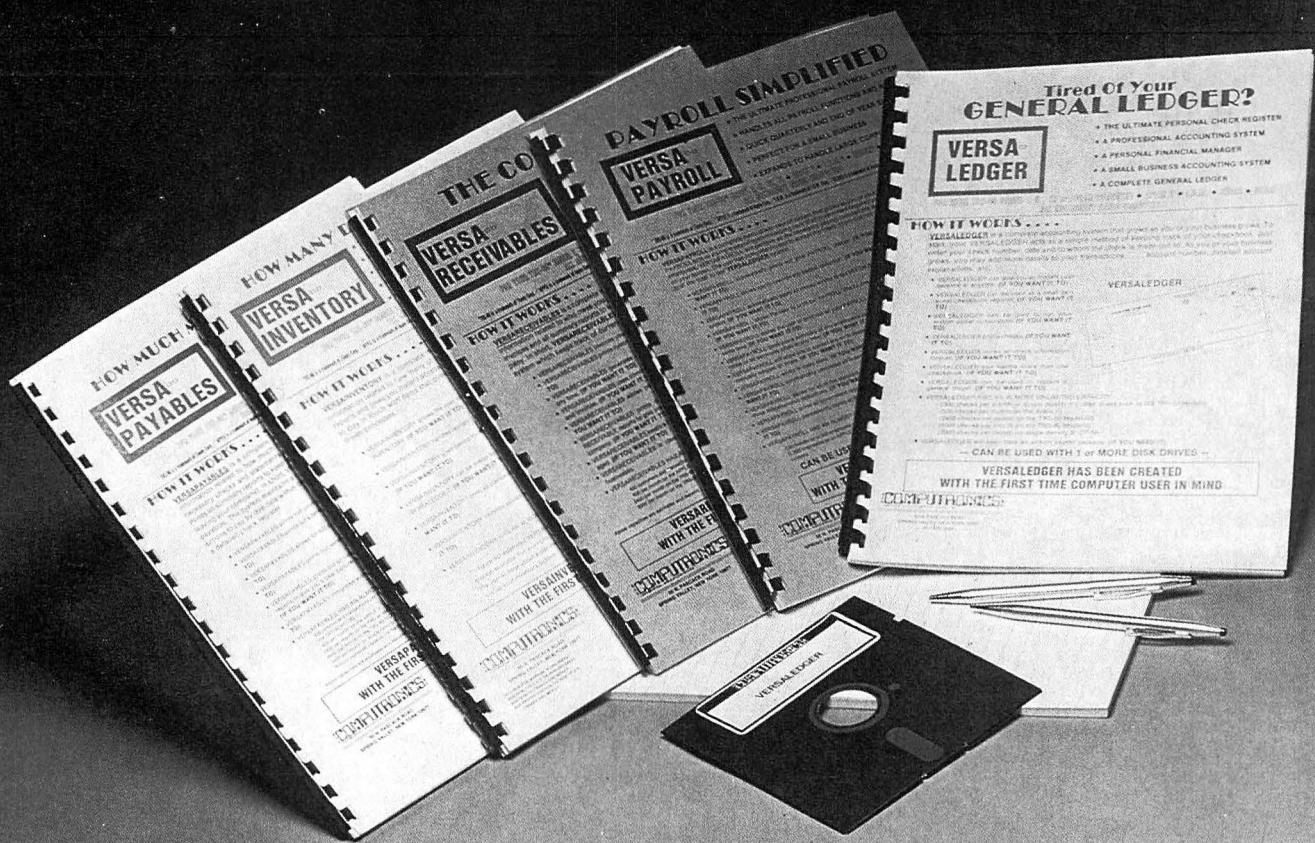
NAME

ADDRESS, CITY, STATE, ZIP

*** ADD \$3 FOR POSTAGE & HANDLING ADD \$4 FOR C.O.D. OR NON UPS AREAS ADD \$5 CANADA & MEXICO EXACT POSTAGE ELSEWHERE ***

Introducing the Most Powerful Business Software Ever!

TRS-80™ (Model I, II, III, or 16) • APPLE™ • IBM™ • OSBORNE™ • CP/M™ • XEROX™



The VERSABUSINESS™ Series

Each VERSABUSINESS module can be purchased and used independently, or can be linked in any combination to form a complete, coordinated business system.

VERSARECEIVABLES™

\$99.95

VERSARECEIVABLES™ is a complete menu-driven accounts receivable, invoicing, and monthly statement-generating system. It keeps track of all information related to who owes you or your company money, and can provide automatic billing for past due accounts. VERSARECEIVABLES™ prints all necessary statements, invoices, and summary reports and can be linked with VERSALEDGER II™ and VERSAINVENTORY™.

VERSAPAYABLES™

\$99.95

VERSAPAYABLES™ is designed to keep track of current and aged payables, keeping you in touch with all information regarding how much money your company owes, and to whom. VERSAPAYABLES™ maintains a complete record on each vendor, prints checks, check registers, vouchers, transaction reports, aged payables reports, vendor reports, and more. With VERSAPAYABLES™, you can even let your computer automatically select which vouchers are to be paid.

VERSAPAYROLL™

\$99.95

VERSAPAYROLL™ is a powerful and sophisticated, but easy to use payroll system that keeps track of all government-required payroll information. Complete employee records are maintained, and all necessary payroll calculations are performed automatically, with totals displayed on screen for operator approval. A payroll can be run totally, automatically, or the operator can intervene to prevent a check from being printed, or to alter information on it. If desired, totals may be posted to the VERSALEDGER II™ system.

VERSAINVENTORY™

\$99.95

VERSAINVENTORY™ is a complete inventory control system that gives you instant access to data on any item. VERSAINVENTORY™ keeps track of all information related to what items are in stock, out of stock, on backorder, etc., stores sales and pricing data, alerts you when an item falls below a preset reorder point, and allows you to enter and print invoices directly or to link with the VERSARECEIVABLES™ system. VERSAINVENTORY™ prints all needed inventory listings, reports of items below reorder point, inventory value reports, period and year-to-date sales reports, price lists, inventory checklists, etc.

COMPUTRONICS

50 N. PASCACK ROAD, SPRING VALLEY, N.Y. 10977

* TRS-80 is a trademark of the Radio Shack Division of Tandy Corp. • *APPLE is a trademark of Apple Corp. • *IBM is a trademark of IBM Corp. • *OSBORNE is a trademark of Osborne Corp. • *CP/M is a trademark of Digital Research • *XEROX is a trademark of Xerox Corp.

VERSALEDGER II™

\$149.95

VERSALEDGER II™ is a complete accounting system that grows as your business grows. VERSALEDGER II™ can be used as a simple personal checkbook register, expanded to a small business bookkeeping system or developed into a large corporate general ledger system **without any additional software**.

- VERSALEDGER II™ gives you almost unlimited storage capacity (300 to 10,000 entries per month, depending on the system),
- stores all check and general ledger information forever,
- prints tractor-feed checks,
- handles multiple checkbooks and general ledgers,
- prints 17 customized accounting reports including check registers, balance sheets, income statements, transaction reports, account listings, etc.

VERSALEDGER II™ comes with a professionally-written 160 page manual designed for first-time users. The VERSALEDGER II™ manual will help you become quickly familiar with VERSALEDGER II™, using complete sample data files supplied on diskette and more than 50 pages of sample printouts.

SATISFACTION GUARANTEED!

Every VERSABUSINESS™ module is guaranteed to outperform all other competitive systems, and at a fraction of their cost. If you are not satisfied with any VERSABUSINESS™ module, you may return it within 30 days for a refund. Manuals for any VERSABUSINESS™ module may be purchased for \$25 each, credited toward a later purchase of that module.

To Order:

**Write or call Toll-free (800) 431-2818
(N.Y. residents call 914-425-1535)**

- * 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

DEALER INQUIRIES WELCOME

All prices and specifications subject to change / Delivery subject to availability.



NEW **THE MARKET PAC** From
COMPUTRONICS

STOCK MARKET ANALYSIS PACKAGE

For TRS-80* Models I, II & III, IBM PC*, Apple* II & III, & CP/M* Computers

*TRS-80 is a trademark of Tandy Corp. - *IBM PC is a trademark of IBM Corp. - *Apple is a trademark of Apple Corp. - *CP/M is a trademark of Digital Research

**Portfolio Valuation • Trend Analysis • Bond Calculations
Money Market Analysis • Portfolio Bookkeeping • Future Projections**

This collection of programs aids both financial professionals and individuals in the evaluation, selection, and management of investment portfolios. It features: coverage of stocks, bonds, convertible securities, options, warrants and annuities; realistic treatment of taxes and commissions; and portfolio selection methods. The clearly written user's manual makes it easy to quickly learn how to use all of the programs in the package, even if you've never used a computer before. With the STOCK MARKET ANALYSIS PACKAGE, you'll soon find that your microcomputer is an indispensable tool, performing all of these instant calculations:

- Annuity Analysis
- Computation of Alpha and Beta Values for Security
- Option Valuation and Hedge Ratio using the Black-Scholes Method
- Bond Valuation — Yield to Maturity & Other Values
- Future Net Worth and Present Value of Projected Investment Schedule
- Compound Interest Computations
- Estimate of Future Earnings Per Share
- Date Computations (Number of Days Between Any Two Dates)
- Option Writing Computation
- Portfolio Listings with Various Profit/Loss Analyses
- Portfolio Selection by Sharpe's Method
- Rate of Return — Variable Inflow
- Valuation of a Share of Stock
- Value of a Warrant
- Investor's Rate of Return on a Convertible Bond
- Dilution Analysis
- Arbitrage Computations
- Future Price Estimation with Inflation
- Seasonal Quantity Indices
- Financial Ratios
- Merger Analysis
- Value of a Right
- Depreciation vs. Cash Flow
- Time Needed for Money to Double, Triple, etc.
- Time Series Analysis — Linear Trend
- Time Series Analysis — Moving Average Trend
- Brokerage Commissions
- Margin Account Computations
- Advanced Option Strategies
- Money Market Computations
- Forecasting Cash Flows
- Leverage Analysis

\$99.95

*** ALL PRICES AND SPECIFICATIONS SUBJECT TO CHANGE ***
DELIVERY SUBJECT TO AVAILABILITY

DEALER INQUIRIES WELCOME

NEW TOLL-FREE
ORDER LINE
(OUTSIDE OF N.Y. STATE)
(800) 431-2818



24 HOUR ORDER LINE
(914) 425-1535

COMPUTRONICS
MATHEMATICAL APPLICATIONS SERVICE™

50 N. PASCACK ROAD
SPRING VALLEY, NEW YORK 10977

30 DAY MONEY-BACK GUARANTEE

- All orders processed within 24 hours
- 30-day money back guarantee
- Add \$3.00 for shipping in UPS areas
- Add \$4.00 for C.O.D. or Non-UPS areas
- Add \$5.00 to Canada or Mexico
- Add exact postage to all other countries

FREE business software directory

- Radio Shack's Model I, II, III.
- Heath's MBASIC and HDOS
- CPM: Xerox, Alto...
- IBM Personal Computer

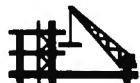
"IDM2 is GREAT!" - publisher of 80-US

"(GL) superior to either the Osborne (SBSG & Taranto) or Radio Shack... MAIL-X has a greater capacity... more flexible than (R.S.)"

- columnist of 80-microcomputing

"imperceptively fast...(DBMS) is a good and reliable workhorse" - publisher of Interface Age

Data base manager, integrated accounting package (AR, AP, GL & Payroll), inventory, word processing, and mailing list. Compare and be selective!



Micro Architect, Inc.
96 Dothan St., Arlington, MA 02174
(617)643-4713

TRS-80™ OWNERS... Enter the World of **POWERSOFT**

Nationally Acclaimed Programs for the TRS-80 Computers

Now from the authors of **SUPER UTILITY+** a complete line of software to increase the use and capabilities of your system.

While **SUPER UTILITY+** won recent honors as Utility Program of the year by the 200,000 readers of *80 Micro*, the quality and consistency carries through the complete Powersoft line.

Professionally written and completely documented, Powersoft programs are accepted as industry standards among TRS-80 enthusiasts.

SUPER UTILITY PLUS	74.95	PowerMAIL+	150.00
SCRIPPLUS	39.95	PowerDRAW	39.95
TOOLBOX FOR LDOS	69.95	PowerDOT	49.95
MASTER MECHANIC SET FOR LDOS	39.95	PowerDRIVER	29.95
BASIC/S COMPILER SYSTEM	49.95	PowerTERM	29.95
MAKE/80	19.95	INSIDE SU+	19.95
SUPERMOVE XFER SYSTEM	700.00	SU+ TECH MANUAL	14.95
DOSPLUS II	250.00	SU+ SPECIAL EDITION	50.00



BOOT UP WITH POWERSOFT...
THE WORLD'S MOST POWERFUL SOFTWARE VENDOR!

WE SUPPORT LDOS! BUY ANY PRODUCT AND
GET LDOS FOR \$99!

SEND FOR OUR COMPLETE CATALOG TODAY!
DEALER INQUIRIES INVITED

AVAILABLE THROUGH SELECTED DEALERS EVERYWHERE



PRODUCTS FROM BREEZE/QSD, INC.

11500 Stemmons Fwy.
Suite 125
Dallas, Texas 75229
Info: (214) 484-2976
Orders Only 800-527-7432

ADVERTISING DIRECTORY

4 ABC Data Products	1-800-854-1555
7 Anitek Software Products	305-259-9397
5 Bealin Corp.	301-490-2744
Cover 2 The Business Division	305-830-8194
61 Computer Plus	1-800-343-8124
6 Computer Services of Danbury	203-743-1299
41 Computer Shopper	1-800-327-9920
25 Cosmopolitan Electronics Corp.	1-800-392-3785
13 Data Systems	305-788-2145
8 EAP Co.	817-498-4242
63-67 H & E Computronics	1-800-431-2818
33 JKR Engineering	408-263-7139
Cover 4 Leading Edge Products, Inc.	1-800-343-6833
Cover 3 Memory Merchant	415-483-1008
68 Micro Architect	617-643-4713
9 Micro Images	212-445-7124
3 Micro Systems Software	1-800-327-8724
8 Nodville Software	203-431-6449
68 Powersoft	1-800-527-7432
10 S. D. C. & S. Co., Inc.	212-849-8600
10 Solutions, Inc.	802-229-0368
6 Triple-D Software	801-546-2833
8 Virginia Micro Systems	703-491-6502

CORRECTIONS

Two articles were accidentally omitted from the Table of Contents of last month's issue, #59. These were "Introduction to Fractions and Spellbound" by Airdrie Ferguson, which started on page 29, and "Cobol Primer #2" by Hubbard C. Goodrich, which began on page 47. We regret any inconvenience to these authors which may have been caused by these omissions.

Put 64K CP/M® 2.2 in your TRS-80 Model III and tap into 2,000 business programs.

Now you can run programs such as WordStar, dBASE II, SuperCalc, MailMerge and virtually thousands of other CP/M-based programs on your TRS-80 Model III.

CP/M 2.2 is the industry standard operating system that gives you access right now to over 2,000 off-the-shelf business programs.

Our plug-in Shuffleboard III comes with 16K of RAM, giving your Model III the power of full 64K CP/M 2.2 without interference of the ROM or video memory. In fact, the Shuffleboard will appear transparent in the TRS-80 mode and will not interfere with any DOS operation.

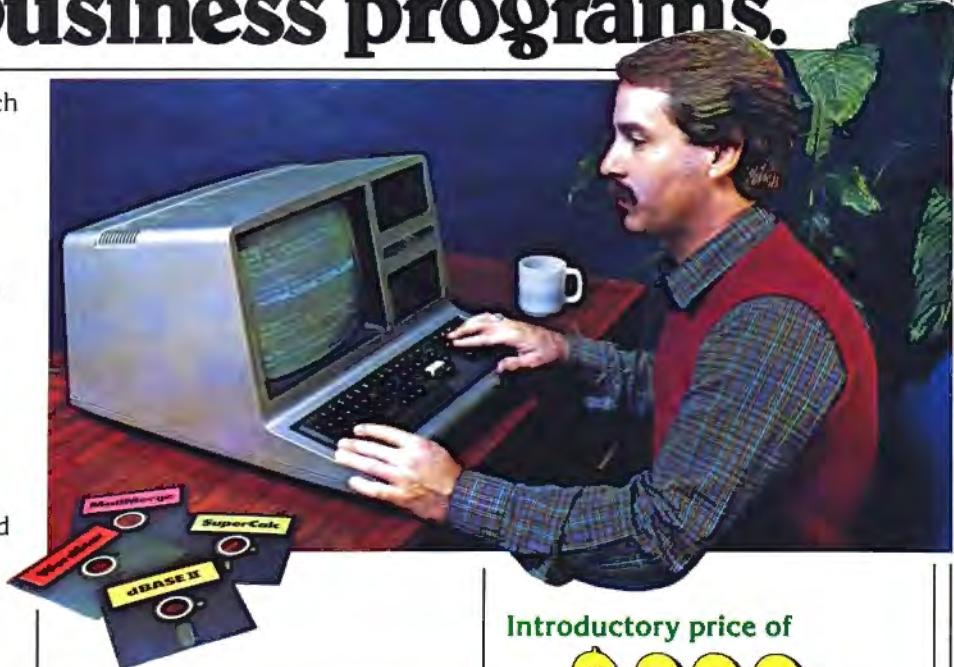
READ and WRITE Osborne, Xerox and IBM personal computer software plus many more popular formats.

Unfortunately, there is no standardized CP/M format for 5 1/4" diskettes. But we have developed a way to READ/WRITE and RUN standard programs under the following single-sided formats: Osborne 1 S/D, Xerox 820 S/D, IBM PC* D/D for CP/M 86 only, Superbrain D/D, Kapro II D/D, HP 125 D/D and TeleVideo D/D.

*Will Read and Write Only.

Easy plug-in installation.

It's so simple. The Shuffleboard III plugs into two existing sockets inside your Model III. There are no permanent modifications, no cut traces and no soldering. You'll be up and running in minutes.



New Products.

80x24 VIDEO BOARD: Features dual intensity screen, programmable cursor control for block, underline & blink rate, on-board bell with audible keyclick, battery-operated real time calendar/clock, full ASCII character set plus 256 special character graphics, dual RS-232 outputs and composite video output.

FLOPPY DISK CONTROLLER: Now you can access 5 1/4" and 8" floppy disk drives in any combination up to 4 drives of S/D density, S/D sided. Tap into a wealth of CP/M software which comes on 8" IBM 3740 format or Pickles & Trout CP/M for the Model II.

SOFTWARE: Additional CP/M software programs are available. Call or write for details.

Introductory price of

\$299.

The Shuffleboard III comes fully burned-in and tested complete with 64K CP/M 2.2 and MBASIC 80 interpreter, plus software manuals and a first class user's manual — with a 1-year limited warranty and 15-day no-risk free trial — for only \$299.

See the Shuffleboard III at your dealer's now.

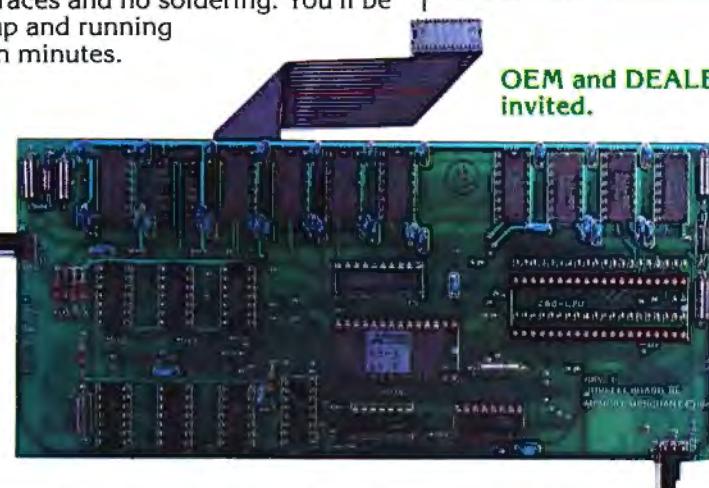
Once you see what the Shuffleboard can do for your Model III you'll want one at once. If your dealer does not yet stock the Shuffleboard have him give us a call. Or send check, money order, VISA or MASTERCARD number (sorry, no COD's) plus \$5 shipping per board (\$17 outside the USA & Canada)* directly to the address below. Cal. residents please add sales tax. Credit card purchases can be phoned in directly and we'll ship from stock.

(415) 483-1008

*Air mail shipments to Canada & all other countries.

Memory Merchant™

14666 Doolittle Drive San Leandro, CA 94577
(415) 483-1008



WordStar & MailMerge are trademarks of MicroPro.
SuperCalc is a trademark of SORCIM.

dBASE II is a trademark of Ashton-Tate.
CP/M is a trademark of Digital Research.

TRS-80 is a trademark of Tandy Corporation.
IBM is a trademark of IBM Corporation.

TRUNKS FOR THE MEMORIES.



Introducing the most logical place to store Elephant Memory Systems® (or lesser brands of disks): The Trunk.

With its alphabetized library index, you can file or retrieve up to 60 disks, instantly.

The Trunk is made of durable molded plastic with a hinged, one-piece lid, to keep disks safe from dust, dirt, and other detriments which disks despise.

And, it's portable. Because the lid doubles as a carrying handle so your Elephant Memory Systems® disks can go anywhere you do.

There's a model for 5 1/4" and 8" floppies, as well as a cassette-and-game file and a special Atari® version.

So if you're looking for the best disk storage system on the market ...

The Trunk is an open-and-shut case.

THE TRUNK. ENDORSED BY ELEPHANTS.

Elephant Memory Systems® Disks

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 for over a life-time of heavy-duty use.

Marketed exclusively by Leading Edge Products, Inc., 225 Turnpike Street, Canton, Massachusetts 02021.
Call: toll-free 1-800-343-6833; or in Massachusetts call collect (617) 828-8150. Telex 951-624.