

The Official **ZENITH** /Heath Computer Users Magazine

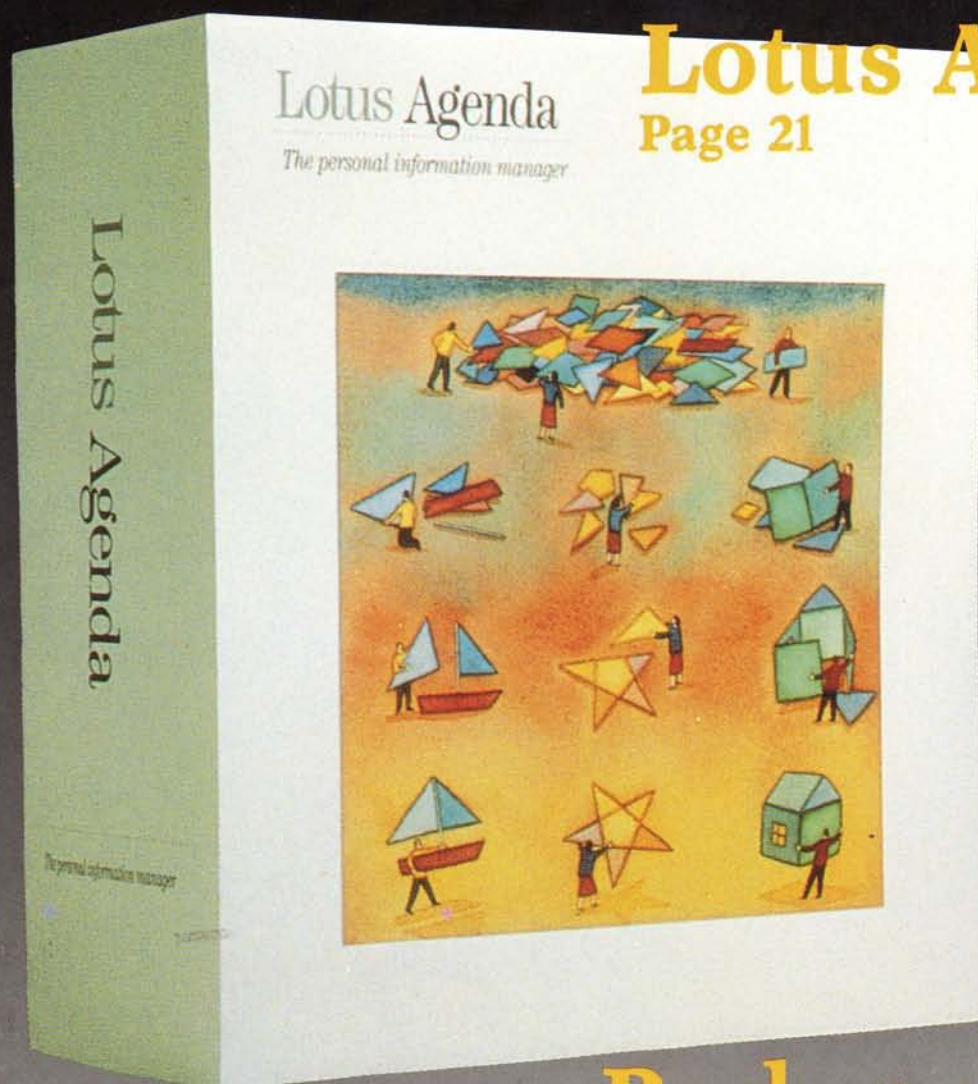
# REMark<sup>®</sup>



September 1989

## Goodbye to an Old Friend

Page 25



## Lotus Agenda

Page 21

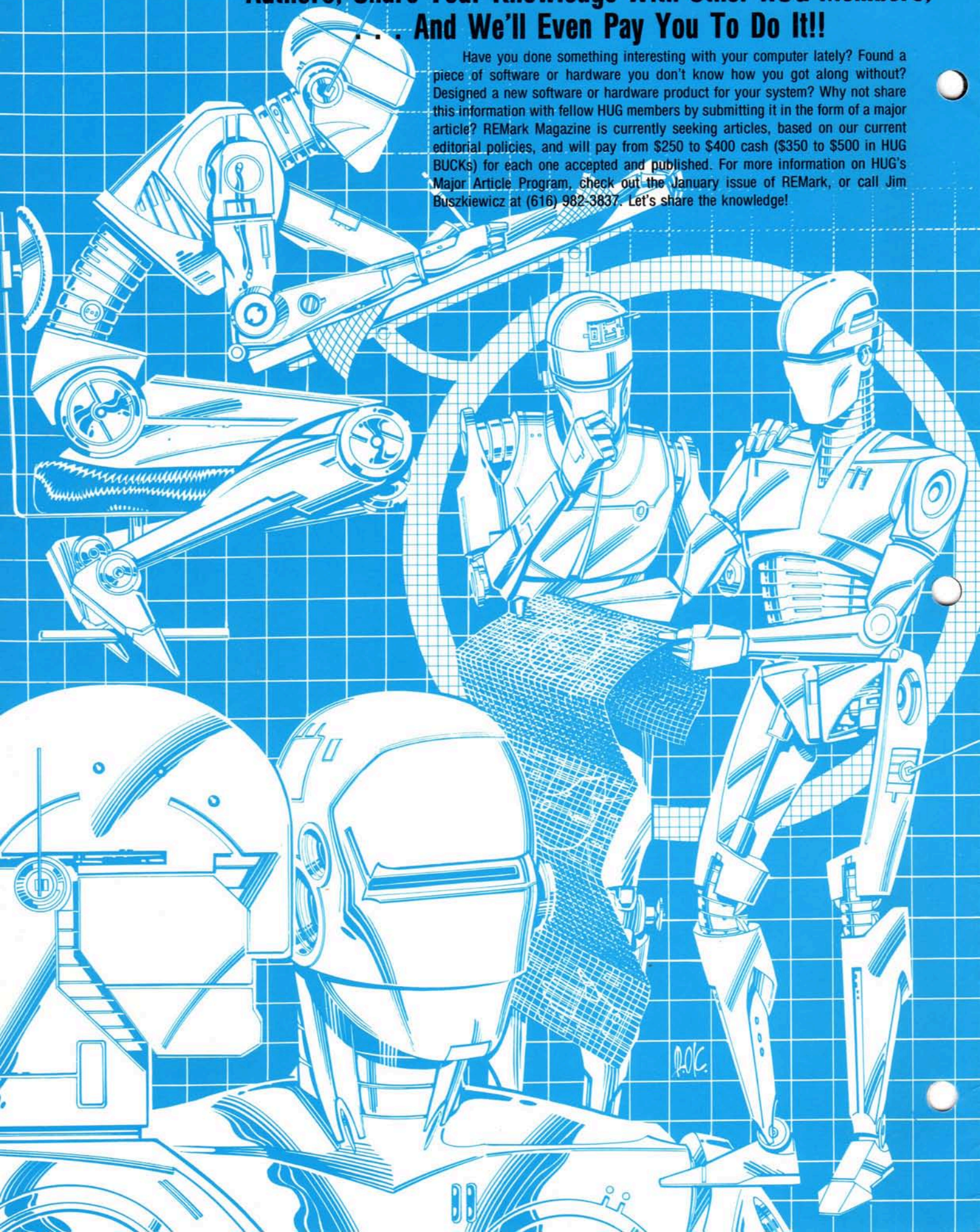
## Prolog and AI

Page 30



# Authors, Share Your Knowledge With Other HUG Members, And We'll Even Pay You To Do It!!

Have you done something interesting with your computer lately? Found a piece of software or hardware you don't know how you got along without? Designed a new software or hardware product for your system? Why not share this information with fellow HUG members by submitting it in the form of a major article? REMark Magazine is currently seeking articles, based on our current editorial policies, and will pay from \$250 to \$400 cash (\$350 to \$500 in HUG BUCKs) for each one accepted and published. For more information on HUG's Major Article Program, check out the January issue of REMark, or call Jim Buszkiewicz at (616) 982-3837. Let's share the knowledge!





# **REMark**®

Volume 10, Issue 9 • September 1989

## PC Compatible

### Powering Up

William M. Adney ..... 15

### Lotus Agenda

Kevin R. Grantham ..... 21

### Taking a Look at Prolog

Gregory D. Elder ..... 30

### Even the Kitchen Sink — Part 2

Richard J. O'Connor ..... 43

## H/Z-100 and PC Compatible

### On the Leading Edge

William M. Adney ..... 25

### Getting Started With . . .

#### WordStar Professional Release 5

Gregory D. Elder ..... 35

## H/Z-100 Only (Not PC)

### Z-100 Survival Kit #7

Paul F. Herman ..... 39

## Resources

HUG Price List ..... 2

Heath/Zenith Product Sale ..... 4

Buggin' HUG ..... 9

Classified Ads ..... 13

## PC Compatibles

All models include the following series of computers: H/Z-130, 140, 150, 160, 170, 180, H/Z-200 and 300.

## Reader Service No.

Reader Service No.	Page No.
196 Covox, Inc.	13
197 DMA Technologies, Inc.	12
104 FBE Research Company, Inc.	29
105 First Capitol Computer	12
191 First Capitol Computer	34
137 Jay Gold Software	11
136 Lindley Systems	10
114 Micronics Technology	22
107 Paul F. Herman	24
117 Payload Computer Services	42
130 Quikdata, Inc.	14
121 Scottie Systems, Inc.	24
199 Traceable Standard Labs	22

## WANTED:

### Quality Assurance Engineer

The Zenith Computer Group is continuing to experience unprecedented growth. We currently are seeking a Quality Assurance Engineer with a BSCE degree and strong background with assembly code or "C" on Zenith machines. A BSCS with strong hardware background will be considered.

For immediate consideration, send your resume to: Ms. Pat Kauffman, Zenith Computer Group, Hilltop Road, St. Joseph, MI 49085. The Zenith Computer Group is a group of wholly-owned subsidiaries of Zenith Electronics Corporation. An Affirmative Action Employer.

# HUG

Managing Editor ..... Jim Buszkiewicz  
(616) 982-3837

Software Engineer..... Pat Swayne  
(616) 982-3463

Production Coordinator ..... Lori Lerch  
(616) 982-3794

Secretary ..... Margaret Bacon  
(616) 982-3463

HUG Bulletin Board ..... (616) 982-3956  
(Modem Only)

HUG Parts Ordering ..... (616) 982-3463

Contributing Editor..... William M. Adney

Advertising.... Rupley's Advertising Service  
Dept. REM, 240 Ward Avenue  
P.O. Box 348  
St. Joseph, MI 49085-0348  
(616) 983-4550

Printer ..... Imperial Printing  
St. Joseph, MI

	U.S. Domestic	APO/FPO & All Others
Initial	\$22.95	\$37.95*
Renewal	\$19.95	\$32.95*
	*U.S. Funds	

Limited back issues are available at \$2.50, plus 10% shipping and handling — minimum \$1.00 charge. Check HUG Product List for availability of bound volumes of past issues. Requests for magazines mailed to foreign countries should specify mailing method and appropriate added cost.

Send Payment to: Heath/Zenith Users' Group  
P.O. Box 217  
Benton Harbor, MI 49022  
(616) 982-3838

Although it is a policy to check material placed in REMark for accuracy, HUG offers no warranty, either expressed or implied, and is not responsible for any losses due to the use of any material in this magazine.

Articles submitted by users and published in REMark, which describe hardware modifications, are not supported by Heath/Zenith Computers & Electronics Center or Heath Technical Consultation.

HUG is provided as a service to its members for the purpose of fostering the exchange of ideas to enhance their usage of Heath/Zenith equipment. As such, little or no evaluation of the programs or products advertised in REMark. The Software Catalog, or other HUG publications is performed by Heath Company, in general, and HUG, in particular. The prospective user is hereby put on notice that the programs may contain faults, the consequence of which Heath Company, in general, and HUG, in particular, cannot be held responsible. The prospective user is, by virtue of obtaining and using these programs, assuming full risk for all consequences.

REMark is a registered trademark of the Heath/Zenith Users' Group, St. Joseph, Michigan.

Copyright © 1989, Heath/Zenith Users' Group

PRODUCT NAME	PART NUMBER	OPERATING		PRICE
		SYSTEM	DESCRIPTION	
H8 - H/Z-89/90				
ACCOUNTING SYSTEM	885-8047-37	CPM	BUSINESS	20.00
ACTION GAMES	885-1220-37	CPM	GAME	20.00
ADVENTURE	885-1010	HDOS	GAME	10.00
ASCIRITY	885-1238-37	CPM	AMATEUR RADIO	20.00
AUTOFILE (Z80 ONLY)	885-1110	HDOS	DBMS	30.00
BHASIC SUPPORT PACKAGE	885-1119-37	HDOS	UTILITY	20.00
CASTLE	885-8032-37	HDOS	ENTERTAINMENT	20.00
CHEAPCALC	885-1131-37	HDOS	SPREADSHEET	20.00
CHECKOFF	885-8010	HDOS	CHECKBOOK SOFTWARE	25.00
DEVICE DRIVERS	885-1105	HDOS	UTILITY	20.00
DISK UTILITIES	885-1213-37	CPM	UTILITY	20.00
DUNGEONS & DRAGONS	885-1093-37	HDOS	GAME	20.00
FLOATING POINT PACKAGE	885-1063	HDOS	UTILITY	18.00
GALACTIC WARRIORS	885-8009-37	HDOS	GAME	20.00
GALACTIC WARRIORS	885-8009-37	CPM	GAME	20.00
GAMES 1	885-1029-37	HDOS	GAMES	18.00
HARD SECTOR SUPPORT PACKAGE	885-1121	HDOS	UTILITY	30.00
HDOS PROGRAMMERS HELPER	885-8017	HDOS	UTILITY	16.00
HOME FINANCE	885-1070	HDOS	BUSINESS	18.00
HUG DISK DUPLICATION UTILITIES	885-1217-37	CPM	UTILITY	20.00
HUG SOFTWARE CATALOG	885-4500	VARIOUS	PRODUCTS THRU 1982	9.75
HUGMAN & MOVIE ANIMATION	885-1124	HDOS	ENTERTAINMENT	20.00
INFO. SYSTEM AND TEL. & MAIL SYSTEM	885-1108-37	HDOS	DBMS	30.00
LOGBOOK	885-1107-37	HDOS	AMATEUR RADIO	30.00
MAGBASE	885-1249-37	CPM	MAGAZINE DATABASE	25.00
MAPLE	885-8005	HDOS	COMMUNICATION	35.00
MAPLE	885-8012-37	CPM	COMMUNICATION	35.00
MICRONET CONNECTION	885-1122-37	HDOS	COMMUNICATION	16.00
MISCELLANEOUS UTILITIES	885-1089-37	HDOS	UTILITY	20.00
MORSE CODE TRANSCEIVER	885-8016	HDOS	AMATEUR RADIO	20.00
MORSE CODE TRANSCEIVER	885-8031-37	CPM	AMATEUR RADIO	20.00
PAGE EDITOR	885-1079-37	HDOS	UTILITY	25.00
PROGRAMS FOR PRINTERS	885-1082	HDOS	UTILITY	20.00
REMARK VOL 1 ISSUES 1-13	885-4001	N/A	1978 TO DECEMBER 1980	20.00
RUNOFF	885-1025	HDOS	TEXT PROCESSOR	35.00
SCICALC	885-8027	HDOS	UTILITY	20.00
SMALL BUSINESS PACKAGE	885-1071-37	HDOS	BUSINESS	75.00
SMALL-C COMPILER	885-1134	HDOS	LANGUAGE	30.00
SOFT SECTOR SUPPORT PACKAGE	885-1127-37	HDOS	UTILITY	20.00
STUDENT'S STATISTICS PACKAGE	885-8021	HDOS	EDUCATION	20.00
SUBMIT (Z80 ONLY)	885-8006	HDOS	UTILITY	20.00
TERM & HTDC	885-1207-37	CPM	COMMUNICATION & UTILITY	20.00
TINY BASIC COMPILER	885-1132-37	HDOS	LANGUAGE	25.00
TINY PASCAL	885-1086-37	HDOS	LANGUAGE	20.00
UDUMP	885-8004	HDOS	UTILITY	35.00
UTILITIES	885-1212-37	CPM	UTILITY	20.00
UTILITIES BY PS	885-1126	HDOS	UTILITY	20.00
VARIETY PACKAGE	885-1135-37	HDOS	UTILITY & GAMES	20.00
WHEW UTILITIES	885-1120-37	HDOS	UTILITY	20.00
XMET ROBOT X-ASSEMBLER	885-1229-37	CPM	UTILITY	20.00
Z80 ASSEMBLER	885-1078-37	HDOS	UTILITY	25.00
Z80 DEBUGGING TOOL (ALDT)	885-1116	HDOS	UTILITY	20.00

## H8 - H/Z-89/90 - H/Z-100 (Not PC)

ADVENTURE	885-1222-37	CPM	GAME	10.00
BASIC-E	885-1215-37	CPM	LANGUAGE	20.00
CASSINO GAMES	885-1227-37	CPM	GAME	20.00
CHEAPCALC	885-1233-37	CPM	SPREADSHEET	20.00
CHECKOFF	885-8011-37	CPM	CHECKBOOK SOFTWARE	25.00
COPYDOS	885-1235-37	CPM	UTILITY	20.00
DISK DUMP & EDIT UTILITY	885-1225-37	CPM	UTILITY	30.00
DUNGEONS & DRAGONS	885-1209-37	CPM	GAMES	20.00
FAST ACTION GAMES	885-1228-37	CPM	GAME	20.00
FUN DISK I	885-1236-37	CPM	GAMES	20.00
FUN DISK II	885-1248-37	CPM	GAMES	35.00
GAMES DISK	885-1206-37	CPM	GAMES	20.00
GRADE	885-8036-37	CPM	GRADE BOOK	20.00
HRUN	885-1223-37	CPM	HDOS EMULATOR	40.00
HUG FILE MANAGER & UTILITIES	885-1246-37	CPM	UTILITY	20.00
HUG SOFTWARE CATALOG UPDATE #1	885-4501	VARIOUS	PRODUCTS 1983 THRU 1985	9.75
KEYMAP CPM-80	885-1230-37	CPM	UTILITY	20.00
MBASIC PAYROLL	885-1218-37	CPM	BUSINESS	60.00
MICRONET CONNECTION	885-1224-37	CPM	COMMUNICATION	16.00
NAVPROGSEVEN	885-1219-37	CPM	FLIGHT UTILITY	20.00
REMARK VOL 3 ISSUES 24-35	885-4003	N/A	1982	20.00
REMARK VOL 4 ISSUES 36-47	885-4004	N/A	1983	20.00
REMARK VOL 5 ISSUES 48-59	885-4005	N/A	1984	25.00
REMARK VOL 6 ISSUES 60-71	885-4006	N/A	1985	25.00
REMARK VOL 7 ISSUES 72-83	885-4007	N/A	1986	25.00
SEA BATTLE	885-1211-37	CPM	GAME	20.00
UTILITIES BY PS	885-1226-37	CPM	UTILITY	20.00
UTILITIES	885-1237-37	CPM	UTILITY	20.00



# Price List

PRODUCT NAME	PART NUMBER	OPERATING SYSTEM	DESCRIPTION	PRICE
X-REFERENCE UTILITIES FOR MBASIC	885-1231-[37]	CPM	UTILITY	20.00
ZTERM	885-3003-[37]	CPM	COMMUNICATION	20.00

## H/Z-100 (Not PC) Only

ACCOUNTING SYSTEM	885-8048-37	MSDOS	BUSINESS	20.00
CALC	885-8043-37	MSDOS	UTILITY	20.00
CARDCAT	885-3021-37	MSDOS	BUSINESS	20.00
CHEAPCALC	885-3006-37	MSDOS	SPREADSHEET	20.00
CHECKBOOK MANAGER	885-3013-37	MSDOS	BUSINESS	20.00
CP/EMULATOR	885-3007-37	MSDOS	CPM EMULATOR	20.00
DBZ	885-8034-37	MSDOS	DBMS	25.00
ETCHDUMP	885-3005-37	MSDOS	UTILITY	20.00
EZPLOT II	885-3049-37	MSDOS	PRINTER PLOTTING UTILITY	25.00
GAMES CONTEST PACKAGE	885-3017-37	MSDOS	GAMES	25.00
GAMES PACKAGE II	885-3044-37	MSDOS	GAMES	25.00
GRAPHICS	885-3031-37	MSDOS	ENTERTAINMENT	20.00
HELPSCREEN	885-3039-37	MSDOS	UTILITY	20.00
HUG BACKGROUND PRINT SPOOLER	885-1247-37	CPM	UTILITY	20.00
KEYMAC	885-3046-37	MSDOS	UTILITY	20.00
KEYMAP	885-3010-37	MSDOS	UTILITY	20.00
KEYMAP CPM-85	885-1245-37	CPM	UTILITY	20.00
MAPLE	885-8023-37	CPM	COMMUNICATION	35.00
MATHFLASH	885-8030-37	MSDOS	EDUCATION	20.00
ORBITS	885-8041-37	MSDOS	EDUCATION	25.00
POKER PARTY	885-8042-37	MSDOS	ENTERTAINMENT	20.00
SCICALC	885-8028-37	MSDOS	UTILITY	20.00
SKYVIEWS	885-3015-37	MSDOS	ASTRONOMY UTILITY	20.00
SMALL-C COMPILER	885-3026-37	MSDOS	LANGUAGE	30.00
SPELL5	885-3035-37	MSDOS	SPELLING CHECKER	20.00
SPREADSHEET CONTEST PACKAGE	885-3018-37	MSDOS	VARIOUS SPREADSHEETS	25.00
TREE-ID	885-3036-37	MSDOS	TREE IDENTIFIER	20.00
USEFUL PROGRAMS I	885-3022-37	MSDOS	UTILITIES	30.00
UTILITIES	885-3008-37	MSDOS	UTILITY	20.00
ZBASIC DUNGEONS & DRAGONS	885-3009-37	MSDOS	GAME	20.00
ZBASIC GRAPHIC GAMES	885-3004-37	MSDOS	GAMES	20.00
ZBASIC GAMES	885-3011-37	MSDOS	GAMES	20.00
ZPC II	885-3037-37	MSDOS	PC EMULATOR	60.00
ZPC UPGRADE DISK	885-3042-37	MSDOS	UTILITY	20.00

## H/Z-100 and PC Compatibles

ADVENTURE	885-3016	MSDOS	GAME	10.00
ASSEMBLY LANGUAGE UTILITIES	885-8046	MSDOS	UTILITY	20.00
BOTH SIDES PRINTER UTILITY	885-3048	MSDOS	UTILITY	20.00
CXREF	885-3051	MSDOS	UTILITY	17.00
DEBUG SUPPORT UTILITIES	885-3038	MSDOS	UTILITY	20.00
DPATH	885-8039	MSDOS	UTILITY	20.00
HADES	885-3040	MSDOS	UTILITY	40.00
HELP	885-8040	MSDOS	CAI	25.00
HEPCAT	885-3045	MSDOS	UTILITY	35.00
HUG BACKGROUND PRINT SPOOLER	885-3029	MSDOS	UTILITY	20.00
HUG EDITOR	885-3012	MSDOS	TEXT PROCESSOR	20.00
HUG MENU SYSTEM	885-3020	MSDOS	UTILITY	20.00
HUG SOFTWARE CATALOG UPDATE #1	885-4501	VARIOUS	PROD 1983 THRU 1985	9.75
HUGMCP	885-3033	MSDOS	COMMUNICATION	40.00
HUGPBBS SOURCE LISTING	885-3028	MSDOS	COMMUNICATION	60.00
HUGPBBS	885-3027	MSDOS	COMMUNICATION	40.00
ICT 8080 TO 8088 TRANSLATOR	885-3024	MSDOS	UTILITY	20.00
MAGBASE	885-3050	VARIOUS	MAGAZINE DATABASE	25.00
MATT	885-8045	MSDOS	MATRIX UTILITY	20.00
MISCELLANEOUS UTILITIES	885-3025	MSDOS	UTILITIES	20.00
PS's PC & Z100 UTILITIES	885-3052	MSDOS	UTILITY	20.00
REMARK VOL 5 ISSUES 48-59	885-4005	N/A	1984	25.00
REMARK VOL 6 ISSUES 60-71	885-4006	N/A	1985	25.00
REMARK VOL 7 ISSUES 72-83	885-4007	N/A	1986	25.00
REMARK VOL 8 ISSUES 84-95	885-4008	N/A	1987	25.00
SCREEN DUMP	885-3043	MSDOS	UTILITY	30.00
UTILITIES II	885-3014	MSDOS	UTILITY	20.00
Z100 WORDSTAR CONNECTION	885-3047	MSDOS	UTILITY	20.00

## PC Compatibles

ACCOUNTING SYSTEM	885-8049	MSDOS	BUSINESS	20.00
CARDCAT	885-6006	MSDOS	CATALOGING SYSTEM	20.00
CHEAPCALC	885-6004	MSDOS	SPREADSHEET	20.00
CP/EMULATOR II & ZEMULATOR	885-6002	MSDOS	CPM & Z100 EMULATORS	20.00
DUNGEONS & DRAGONS	885-6007	MSDOS	GAME	20.00
EZPLOT II	885-6013	MSDOS	PRINTER PLOTTING UTILITY	25.00
GRADE	885-8037	MSDOS	GRADE BOOK	20.00
HAM HELP	885-6010	MSDOS	AMATEUR RADIO	20.00
KEYMAP	885-6001	MSDOS	UTILITY	20.00
PS's PC UTILITIES	885-6011	MSDOS	UTILITIES	20.00
POWERING UP	885-4604	N/A	GUIDE TO USING PCS	12.00
SCREEN SAVER PLUS	885-6009	MSDOS	UTILITIES	20.00
SKYVIEWS	885-6005	MSDOS	ASTRONOMY UTILITY	20.00
TCSPELL	885-8044	MSDOS	SPELLING CHECKER	20.00
ULTRA RTTY	885-6012	MSDOS	AMATEUR RADIO	20.00

The following HUG Price List contains a list of all products in the HUG Software Catalog and Software Catalog Update #1. For a detailed abstract of these products, refer to the HUG Software Catalog, Software Catalog Update #1, or previous issues of REMark.

Magazines everywhere, and no way to reference the wealth of information they hold? Not anymore! Now there's **MAGBASE**; a database designed specifically for referencing magazine articles. Don't let those one-hundred-and-some back issues of REMark, or C Users Journal, or Veterinary Medicine, (or any magazine) gather dust, use **MAGBASE**, and find that article you read two years ago! **MAGBASE** is available for **MSDOS HUG P/N 885-3050** or **CP/M (P/N 885-1249-[27])**.

**LAPTOP OWNERS** . . . don't feel left out! All of HUG's MSDOS software is available on 3-1/2" micro-floppies too! When ordering, just add a "-80" to the 7-digit HUG part number. For the standard 5-1/4" floppy, just add a "-37".

Make the no-hassle connection with your modem today! **HUGMCP** doesn't give you long menus to sift through like some modem packages do. With **HUGMCP**, YOU'RE always in control, not the software. Order **HUG P/N 885-3033-37** today, and see if it isn't the easiest-to-use modem software available. They say it's so easy to use, they didn't even need to look at the manual. "It's the only modem software that I use, and I'm in charge of the HUG bulletin board!" says Jim Buszkiewicz. **HUGMCP** runs on ANY Heath/Zenith computer that's capable of running MS-DOS!

## ORDERING INFORMATION

For VISA and MasterCard phone orders, telephone the Heath Users' Group directly at (616) 982-3463. Have the part number(s), descriptions, and quantity ready for quick processing. By mail, send your order, plus 10% postage and handling (\$1.00 minimum charge, up to a maximum of \$5.00) to: Heath Users' Group, P.O. Box 217, Benton Harbor, MI 49022-0217. VISA and MasterCard require minimum \$10.00 order. No C.O.D.s accepted.

Questions regarding your subscription? Call Margaret Bacon at (616) 982-3463.





# EXTREMELY LIMITED QUANTITIES!!

**Call TOLL-FREE Today!**

(Prices good from July 1 through October 1, 1989)

**1-800-253-0570**

**Use order code 217-063**

MODEL NUMBER	PRODUCT DESCRIPTION	OLD PRICE	SALE PRICE
A-200	CA DATA DISPLAY	\$1,199.00	\$699.00
AA-3000	MIG ABILITY PLUS 1.0	\$259.95	\$99.00
AAW-2500	ASSEMBLED 100W STEREO POWER AMPLIFIER	\$499.00	\$399.00
AA-2500	KIT 100W STEREO POWER AMPLIFIER	\$449.00	\$359.00
AB-3000	MIG ABILITY 1.2	\$99.95	\$49.00
ACW-2540	ASSEMBLED STEREO CASSETTE DECK	\$349.00	\$279.00
ADW-2530	ASSEMBLED COMPACT DISC PLAYER	\$349.00	\$279.00
AE-2501	STEREO EQUIPMENT CABINET	\$189.95	\$149.00
AJW-2520	ASSEMBLED AM/FM DIGITAL STEREO TUNER	\$249.00	\$199.00
AJ-2520	KIT AM/FM DIGITAL STEREO TUNER	\$229.00	\$180.00
AMA-1	FUNDAMENTALS OF FINANCE LEARNING PROGRAM	\$100.00	\$60.00
AMA-2	ASSERTIVENESS TRAINING PROGRAM	\$100.00	\$60.00
AMA-3	TECHNICAL REPORTS LEARNING PROGRAM	\$100.00	\$60.00
AMA-4	PROJECT MANAGEMENT LEARNING PROGRAM	\$130.00	\$75.00
APW-2510	ASSEMBLED AUDIOPHILE STEREO PREAMPLIFIER	\$399.00	\$319.00
AP-2510	KIT AUDIOPHILE STEREO PREAMPLIFIER	\$349.00	\$279.00
ARA-355-1	EV DYNAMIC CARDIOD MICROPHONE	\$69.96	\$49.95
ARA-355-2	ALL WEATHER COLUMN SPEAKER	\$149.95	\$99.95
ARA-355-3	8' CEILING LINE SPEAKER W/BAFFEL	\$29.97	\$12.95
AR-355	35W PA MIXER AND RECEIVER	\$299.95	\$199.00
ASW-1082	ASSEMBLED 2-WAY BOOKSHELF STEREO SPEAKER	\$159.95	\$119.00
ASW-1230	ASSEM. 3-WY FLOOR STANDING STEREO SPEAKR	\$399.00	\$309.00
AS-1082	KIT JBL 2-WAY BOOKSHELF SPEAKER	\$129.95	\$99.00
AS-1230	KIT 3-WAY FLOOR STANDING STEREO SPEAKER	\$349.00	\$259.00
BC-100X-LT	100 Ch 11 Band Hand Scanner	\$219.95	\$174.88



MODEL NUMBER	PRODUCT DESCRIPTION	OLD PRICE	SALE PRICE
BC-175X-L	16 Ch 11 Band Scanner	\$169.95	\$144.88
BC-227	X-10 WALL RECEPTACLE MODULE	\$19.99	\$14.99
BC-2458	remote open door indicator for BC-6000-1	\$24.99	\$13.99
BC-2481	extra remote transmitter for BC-6000-1	\$34.99	\$19.99
BC-2700	X-10 TELEPHONE RESPONDER	\$49.95	\$29.95
BC-2807	X-10 THERMOSTAT CONTROLLER	\$19.99	\$12.99
BC-284	X-10 BURGLAR ALARM	\$39.99	\$29.95
BC-55-XLT	10 Ch 10 Band Hand Scanner	\$134.95	\$104.88
BC-580X-LT	100 Ch 11 Band Base/Mobile Scanner	\$249.95	\$178.88
BC-6000-1	Stanley garage door opener	\$189.95	\$139.95
BC-760X-LT	100Ch 12Band Base/Mobile Scanner	\$319.95	\$234.88
BH-110	DIGITAL FEVER THERMOMETER	\$9.95	\$4.95
BH-1735	BLOOD PRESSURE METER	\$49.95	\$34.95
BH-1750	BLOOD PRESSURE METER WITH PRINTER	\$99.95	\$79.95
BH-2200	DIGITAL SCALE	\$39.95	\$19.95
BH-8500	DIGITAL SCALE W/INFRA-RED REMOTE	\$59.95	\$39.95
BP-1234	NI-CAD BATTERY CHARGER	\$21.99	\$17.95
BW-400	PC Wx Card	\$399.95	\$299.95
CGW-1562	AUTOMOTIVE G-METER	\$39.95	\$36.95
CI-2065	AUTO CHARGING SYSTEM TESTER	\$24.95	\$19.95
DC -1000	DG DOUBLECOM	\$149.95	\$99.00
EA-100	MICROPROCESSOR TRAINER ACCESSORY BOARD	\$750.00	\$100.00
EB-1010	ELECTRONICS LEARNING DICTIONARY	\$15.00	\$5.00
EB-1030	ELECTRONICS READY REFERENCE GUIDE	\$15.00	\$6.00
EC-1126	ARTIFICIAL INTELLIGENCE COURSE	\$70.00	\$40.00
EC-1127	TI ARTIFICIAL INTELLIGENCE SOFTWARE	\$500.00	\$150.00
EC-1305	AUTOCAD BASIC FOR Z-100	\$300.00	\$75.00
EC-1305-PC	AUTOCAD BASIC FOR PC'S	\$300.00	\$75.00
EC-1307-PC	AUTOCAD WITH ADE 1 & 2	\$1,400.00	\$400.00
EC-1310-PC	AUTOCAD AUTO SKETCH FOR PC'S	\$79.95	\$30.00
EE-1002	TRANSISTOR CIRCUIT DESIGN COURSE	\$44.95	\$29.00
EE-1900	INTELLIGENT MACHINES COURSE	\$70.00	\$25.00
EE-1901	ELECTRONICS FOR AUTOMATION COURSE	\$100.00	\$50.00
EE-1903	INDUSTRIAL ROBOTICS & AUTOMATION COURSE	\$80.00	\$50.00
EE-3140-A	CONCEPTS OF ELECTRONICS COURSE	\$55.00	\$40.00
EE-3820	6811 MICROPROCESSOR PROGRAMMING COURSE	\$100.00	\$60.00
EW -2000	MIG ENRICH 1.2	\$199.95	\$49.00
ETA-3600-1	BREADBOARDING BLOCK ACCESSORY	\$20.00	\$10.00
ETB-3300	BREADBOARD MODULE FOR BACKPACK ACCESSORY	\$100.00	\$40.00
ETB-6000	WIRE WRAP MODULE FOR BACKPACK ACCESSORY	\$40.00	\$25.00
ETB-6104	ELECTRONICS CIRCUITS BACKPACK BOARD	\$200.00	\$100.00
ETS-18-C	HERO I ROBOT WITHOUT ARM (KIT)	\$1,099.00	\$795.00
ETS-19-32	STANDALONE ROBOT ARM WITH BASE (KIT)	\$999.00	\$695.00
ETS-19-D	HERO 2000 ROBOT SYSTEM (KIT)	\$2,995.00	\$1,500.00
ETW-19-31	HERO 2000 CASSETTE INTERFACE	\$100.00	\$30.00
ETW-3200-B	DIGITAL TRAINER (WIRED)	\$200.00	\$50.00
ETW-3300-B	LABORATORY BREADBOARD TRAINER (WIRED)	\$194.00	\$100.00
ETW-3567	TRAINER BACKPACK ACCESSORY (WIRED)	\$200.00	\$130.00
ET-3567	TRAINER BACKPACK ACCESSORY (KIT)	\$130.00	\$100.00
EVM-2015-A	12" AMBER MONITOR	\$160.00	\$75.00
EWS-100-D	16-BIT MICROPROCESSOR TRAINING SYSTEM	\$1,599.00	\$999.00
EWS-100-E	ADVANCED MICRO TRAINER/ACCESSORY BOARD	NONE	\$599.00
EWS-18-A	HERO I ROBOT WITH ARM (WIRED)	NONE	\$1,495.00
EWS-19-32	STANDALONE ROBOT ARM WITH BASE (WIRED)	\$1,995.00	\$1,295.00
GDA-3196-1	16" X 20" FURNACE FILTER FOR GD-3196	\$99.95	\$74.95



MODEL NUMBER	PRODUCT DESCRIPTION	OLD PRICE	SALE PRICE
GDA-3196-2	16' X 25' FURNACE FILTER FOR GD-3196	\$99.95	\$74.95
GDA-3196-3	20' X 20' FURNACE FILTER FOR GD-3196	\$99.95	\$74.95
GDA-3196-4	20' X 25' FURNACE FILTER FOR GD-3196	\$99.95	\$74.95
GDP-1108	WELLER TEMP-CONTROLLED SOLDERING STATION	\$89.95	\$79.95
GDP-1109	REPLACEMENT TIPS FOR WELLER SOLD. STATION	\$17.95	\$12.95
GDP-3100	UNGAR IRON REPLACEMENT TIPS	\$8.95	\$6.95
GDP-3125	25W UNGAR SOLDERING IRON	\$12.95	\$8.95
GDP-3140	40W UNGAR SOLDERING IRON	\$17.95	\$9.95
GDS-1297	PORTABLE AIR CLEANER	\$249.95	\$179.00
GDZ-143	ZENITH UNIVERSAL REMOTE CONTROL	\$49.95	\$29.95
GD-1151-H	KIT ULTRASONIC CLEANER	\$99.95	\$79.96
GD-1701	KIT FLOOD ALARM	\$24.95	\$12.95
GD-3196	KIT FURNACE AIR CLEANER	\$149.95	\$79.95
GD-3610	heat actuated outdoor light controller	\$69.95	\$29.95
GHP-1270	WELLER ELECTRONIC TOOL KIT W/CASE	\$42.95	\$34.36
GRA-2035-1	TV/VCR VIDEO CENTER	\$79.95	\$55.97
GR-4500	45' SUPER SCREEN TELEVISION KIT	\$1,295.00	\$999.00
HBT-40-1	40 Mbyte tape cartridge, HBT-40 series	\$24.95	\$19.95
HBT-40-AT	internal 40 Mbyte back-up tape drive	\$389.95	\$299.95
HBT-40-H4	external adapter for HBT-40, H-40 series	\$199.95	\$149.95
HBT-40-HZ	external adapter for HBT-40, Heat/Zenith	\$199.95	\$149.95
HBT-40-SD	external adapter for HBT-40, IBM	\$139.95	\$99.95
HBT-40-YT	internal 40 Mbyte back-up tape drive	\$379.95	\$289.95
HCA-9	PRINT STAND FOR H-25	\$49.00	\$25.00
HCA-3000	co-ax connector/adaptor kit	\$69.95	\$69.95
HDP-1472	2 METER MAG ANTENNA	\$29.95	\$19.95
HDP-1474	2 METER GLASS ANTENNA	\$42.95	\$32.95
HDP-7800	SWL TRAP ANTENNA	\$64.95	\$48.88
HD-1274	117 VAC 12 outlet power strip kit	\$19.99	\$17.95
HD-4040	TNC-1	\$114.88	\$114.88
HP2-2706-B	HPDJ FONT: PRESTIGE	\$95.00	\$59.00
HP2-2706-P	HPDJ FONT: TMS/RMN	\$125.00	\$79.00
HP2-2707-E	HPDJ FONT: EPSON FX80	\$75.00	\$49.00
HP8-8390-B	HP MAC INTF: SCANJET	\$595.00	\$299.00
HSM-100	spreadsheet, word proc, database, 5 1/4"	\$100.00	\$37.99
HSM-100-3	spreadsheet, word proc, database, 3 1/2"	\$99.95	\$37.99
HS-2526-A & HWD-20-AT	286/12, large case, w/20 Mbyte, kit	\$2,138.00	\$1,949.00
HS-2526-A & HWD-4028	286/12, large case, w/40 Mbyte, kit	\$2,358.00	\$2,249.00
HS-3860	386/12 laptop w/40 Mbyte, kit	\$4,999.00	\$4,369.00
HS-3860-M	386/12 laptop w/40 Mbyte, modem, kit	\$5,249.00	\$4,589.00
HS-386-C	386/16, large case, kit	\$2,999.00	\$2,699.00
HS-386-C & HWD-4028	386/16, large case, 40 Mbyte, kit	\$3,658.95	\$3,049.00
HS-40-A & HWD-420	286/8, mini case, 20 Mbyte, kit	\$2,138.00	\$1,849.00
HS-40-A & HWD-440	286/8, mini case, 40 Mbyte, kit	\$2,298.00	\$1,999.00
HS-42 & HWD-420	286/12, mini case, 20 Mbyte	\$2,338.00	\$2,049.00
HS-42 & HWS-440	286/12, mini case, 40 Mbyte	\$2,498.00	\$2,149.00
HWD-20	20 Mbyte disk & controller for XT	\$299.00	\$269.00
IDS-4801	EPROM programmer system	\$279.00	\$179.00
ID-4804	byte probe	\$29.95	\$24.95
IHO-2201	IN-HOUSE ACCOUNTANT	\$99.99	\$49.00
IM-2203	line voltage monitor kit	\$39.00	\$29.95
IM-2260	3 1/2 digit bench DMM kit	\$139.95	\$119.95
IM-5217	VOM with rugged case	\$39.95	\$39.95
IN-6000	MIG IN-HOUSE ACCT 2.1	\$199.95	\$49.00
LM-2000	80,000 word dictionary, plus thesaurus	\$159.95	\$99.95

MODEL NUMBER	PRODUCT DESCRIPTION	OLD PRICE	SALE PRICE
LU-500	video transeiver for telephones	\$289.95	\$179.95
MD-550-1	VERBATIM 5' STD DISKS	\$9.99	\$6.00
MD-600	VERBATIM 5' HD DISKS	\$19.99	\$11.00
MF-360	VERBATIM 3.5' DISKS	\$19.99	\$11.00
MIA-2020-1	Transom Transducer	\$54.95	\$49.95
MIA-2020-2	Thru hull Transducer	\$129.95	\$99.95
MI-2020	20/20 Video Fish Finder	\$265.00	\$199.95
MI-2040	Color Video Fish Finder	\$445.00	\$299.95
MM-5	AMARAY MEDIAMATE 5	\$12.99	\$6.00
MM-5-XL	AMARAY MEDIAMATE 5XL	\$24.99	\$9.95
MO-5500	MIG POCKET MODEM IBM	\$159.95	\$99.00
MO-7500	MIG POCKET MODEM MAC	\$149.95	\$99.00
MP-4	AMARAY MEDIAPACK 4	\$11.99	\$6.00
NE-2112	KIT HEAT SNIFFER	\$19.95	\$14.95
PA-430	MICROSOFT BOOKSHELF	\$349.95	\$149.00
PC-122	INTEL ABOVEBOARD PS	\$220.00	\$50.00
PC-131	USR PCM-5 MODEM (PC)	\$205.00	\$79.00
PC-132	ZOOM MODEM PC-1200XL	\$199.95	\$99.00
PC-136	ZOOM PC2400XL MODEM	\$349.95	\$129.00
PC-146	INTEL ABOVE BOARD	\$199.00	\$199.95
PCS-110	DG ANAL. PORT & CLOCK	\$79.95	\$49.00
PD-500	CAM VOICEMAIL	\$349.99	\$229.00
PD-510	CPC FAX BOARD	\$499.00	\$249.00
PD-520	CPC HAND SCANNER	\$249.95	\$149.00
PD-2424	2400B EXTERNAL MODEM	\$199.95	\$129.00
PF-132	BUSH MONITOR PLATFORM	\$17.95	\$9.00
PF-156	NEW GENERATION PRNTR STAND	\$49.00	\$39.00
PF-157	NEW GENERATION 2-DRWR CAB	\$59.00	\$55.00
PF-161	ARTISAN HUTCH	\$120.00	\$49.00
PF-167	ARTISAN FILE DRAWER	\$69.00	\$49.00
PI-23	KRAFT MICROMOUSE	\$69.99	\$39.00
PI-5000-1	AMDEK CD ROM	\$999.00	\$599.00
PK-3-A	rf probe for oscilloscopes, kit	\$9.95	\$4.95
PM-100	DISKETTE TRAY	\$19.99	\$12.00
PM-125	RS 232 M/M GENDER CNGR	\$11.99	\$7.00
PM-126	RS 232 F/F GENDER CNGR	\$11.99	\$7.00
PM-160	HYPERACCESS SFTWARE PK	\$149.00	\$79.00
PM-160-3	HYPERACCESS 3.23, 3.5'	\$149.95	\$79.00
PM-162	HYPER/ACCESS: Z-100	\$149.95	\$39.00
PM-1493	PEC OUTLET STRIP GP	\$9.99	\$7.00
PM-2100	KEN MASTERPIECE	\$129.95	\$79.00
PM-2212	KEN KYBRD SLIDEAWAY	\$39.99	\$29.00
PM-2250	KEN PRINT MUFFLER 80	\$39.99	\$29.00
PM-3269	ICI FLIP SORT 5.25'	\$7.95	\$5.00
PM-8326	NAS 3.5' DS DDKS: 10	\$24.99	\$11.00
PM-8703	NAS VIDEO BRICK	\$7.00	\$19.95
PM-8705	NAS CAMERA BRICK	\$69.95	\$35.00
PM-9125	NAS 5.25' DSHD: 10	\$24.99	\$11.00
PMK-121	25 line data switchbox, kit	\$49.95	\$39.95
PO-310	SYQUEST CART: 10MB	\$139.95	\$99.00
PP-105	C. ITOH PROWRITER JR.	\$189.00	\$149.00
PP-111	SAFT 200VA SPS	\$199.00	\$199.00
PP-115	SAFT SPS0400 350W UPS	\$549.95	\$319.00
PP-221	EPSON LX-86	\$199.00	\$149.00
PP-230	EPSON PRINT RX-100	\$329.00	\$99.00



MODEL NUMBER	PRODUCT DESCRIPTION	OLD PRICE	SALE PRICE
PP-243	EPSON LQ-1000	\$999.95	\$599.00
PP-256	EPSON EX-800	\$699.00	\$449.00
PP-400	KOALAPAD TOUCH TABLET	\$69.95	\$20.00
PP-511	IOMEGA A210H 10M/2DR	\$2,799.00	\$500.00
PP-714	CANON PC-14 COPIER	\$625.00	\$399.00
PP-2239-A	HP RW SHEET FEED	\$250.00	\$149.00
PP-3443	1M RAM BD LASERJET II	\$595.00	\$399.00
PP-3444	2M RAM BD LASERJET II	\$1,195.00	\$699.00
PP-8290	HP PC INTF: SCANJET	\$595.00	\$299.00
PP-9190	HP SCANJET SCANNER	\$1,495.00	\$699.00
PPA-220-2	EPSON LX-80 TRAC. FEED	\$39.95	\$15.00
PPA-243-2	TRACTOR: LQ-1000	\$69.99	\$19.95
PPA-246-1	RIBBON: LQ-800	\$9.00	\$4.95
PPA-246-2	TRACTOR: LQ-800	\$59.99	\$19.95
PPA-400-1	KOA SPEEDKEY SOFTWARE	\$29.95	\$10.00
PR-120	RIBBON LQ-2500	\$9.00	\$4.95
PR-130	RIBBON: LQ-950	\$12.00	\$6.95
PR-609	PRM WALL PLUG/SURGE	\$19.99	\$14.95
PS-104	pocket sized 80,000 word speller	\$59.95	\$34.95
P-350-2	2 ea 350 MHz 'scope/counter probes	\$99.95	\$79.95
QM-57	QUME CRYSTAL PRINT WP	\$1,169.10	\$995.00
SA-2550	REMOTE ANTENNA TUNER	\$99.95	\$64.88
SBS-1400-1	HF TRANSCEIVER, POWER SUPPLY, & SPEAKER	\$1,129.90	\$899.90
SCD-1	SUPERCONDUCTOR DEMONSTRATION KIT	\$24.95	\$17.95
SCD-1-1	EXTRA SUPERCONDUCTOR PELLET	\$14.95	\$11.95
SD-4801	EPROM PROGRAMMER	\$249.00	\$150.00
SD-4803	EPROM ERASER	\$100.00	\$49.95
SD-4850	50 MHZ DIGITAL SCOPE	\$750.00	\$495.00
SE-100	Mr. CIRCUIT 30-IN 1 ELECT. EXPERIMENTER	\$24.95	\$19.96
SE-101	SOLAR POWER 150 EXPERIMENTER LAB	\$29.95	\$23.96
SK-103	ELECTRIC MOTOR SPEED CONTROL	\$39.95	\$24.95
SK-201	moden surge protector, kit	\$14.95	\$6.97
SK-202	RS-232 surge protector, kit	\$14.95	\$6.49
SK-203-H	printer buffer--speeds throughput, kit	\$199.95	\$169.95
SK-204	RS-232 cable protocol switch, kit	\$29.95	\$24.95
SK-205	baud rate analyzer	\$49.95	\$27.95
SK-210	RS-232 patch board	\$14.95	\$6.97
SK-99	STARTER KIT CABINET ACCESSORY	\$6.95	\$4.95
SL-5410	HEATH/ZENITH MOTION SENSOR LIGHT CONTROL	\$39.97	\$19.97
SMD-1	SURFACE MOUNT EXPERIMENTER KIT	\$14.95	\$10.95
SMT-1	SURFACE MOUNT TOOL KIT	\$39.95	\$24.95
SM-2440	1.3 GHz counter, deluxe	\$399.95	\$329.95
SO-4251	50 MHz dual trace, d'lyd sweep oscscope	\$749.95	\$689.95
STS-4750	16 CHANNEL LOGIC ANALYZER	\$2,195.00	\$1,500.00
TD-1089	PROGRAMABLE ELECTRONIC DOORBELL	\$49.95	\$35.95
TE-37	TECHMAR GRAPHICS MASTR	\$545.00	\$100.00
TE-200	OPEN-AIR DESIGN STEREO HEADPHONES	\$59.95	\$45.00
TE-400	CLOSED BACK DESIGN STEREO HEADPHONES	\$69.95	\$55.00
TPI-5000	20 coax cables, various connectors	\$49.95	\$49.95
VAC-618	CARRY CASE FOR VM-6180 CAMCORDER	\$49.95	\$24.95
VM-6180	ZENITH VHS-C CAMCORDER	\$849.00	\$749.00
VRE-200	ZENITH 4-HEAD DOUBLE AZMITH MIDI VCR	\$329.00	\$309.00
V-617	PICTURE/VIDEO TRANSFER	\$79.95	\$49.95
WM-1200	pocket speller, synonyms, definitions	\$79.95	\$47.95

# BUGGIN' HUG

## Smetana on Networking

Dear Mr. Buszkiewicz:

I'm guessing that just about everyone with some contact with the computer world realizes we're in the middle of a revolution to network everything together. Personally, I was never too keen on the idea. The networking hardware, the lines, and servers end up costing more than the machines they are linking. And what does all this "connectivity" buy you? The ability to move a lot of data around quickly, grab programs or compilers from remote machines, send mail or program results to others on the net without leaving your seat. Myself, I always welcomed the opportunity to get up and stretch a bit if I needed to go see a colleague and if I left my mail in my mailbox, I didn't have to think about responding to all the requests for this or that because I didn't know about it yet.

Work stations on networks are supposed to make your job easier because they have multitasking capabilities. You can do three or four things at one time — if your mind works that way. I guess I must be an anachronism because mine doesn't. When I debug code I've written, I use the time during compile and link to think about what I've just done and what other steps I may need to take. Or, I can close my eyes or look off into space and give my eyes a little rest.

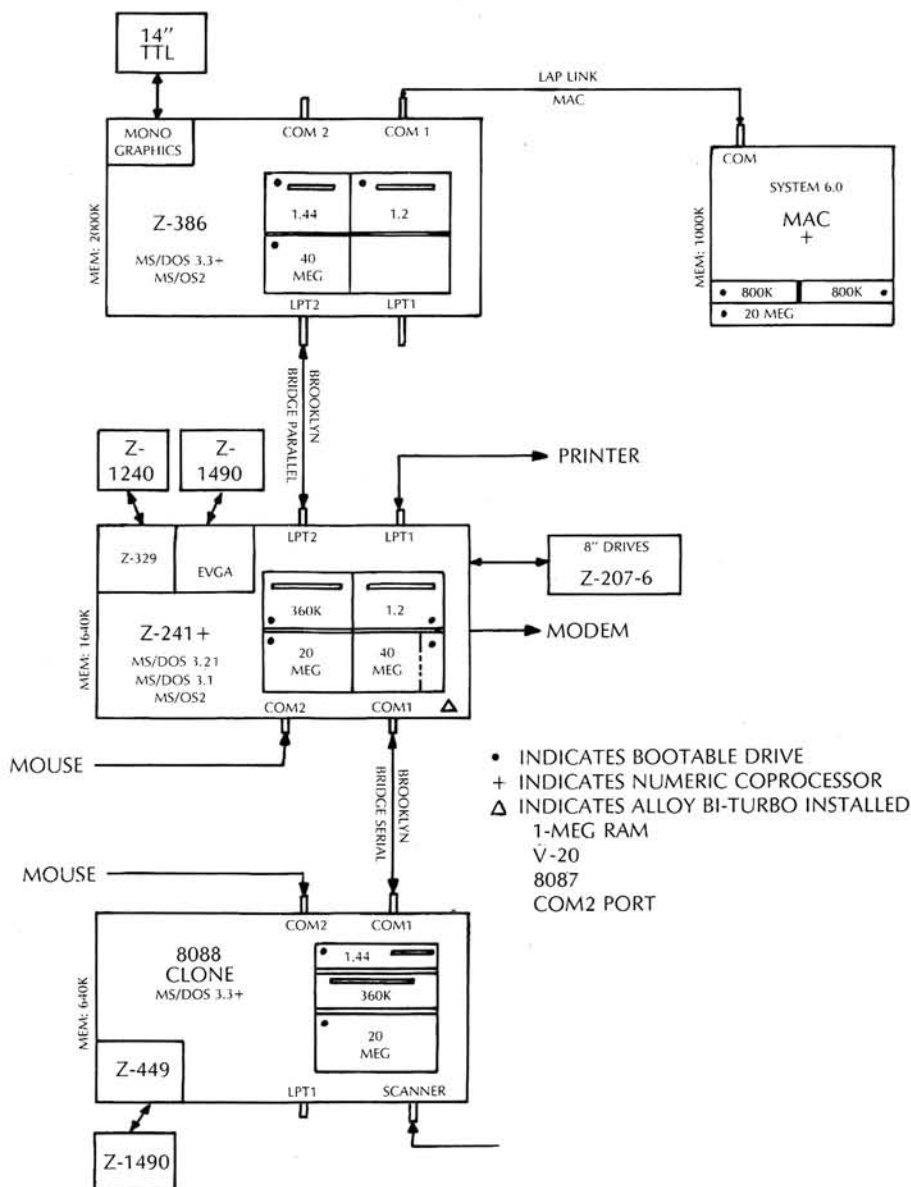
Given this mindset, you wouldn't think I'd be interested in getting involved in networking at home, too. Originally, I wasn't. But then something interesting happened. I ran out of slots on my computer. That's not all that unusual for people who like to try a lot of different hardware. Other engineers to whom I've spoken tell me that's why they like flip-top computer cases so much; these cases allow one to get in there and change boards more easily than does a slide-on case. One engineer at Zenith Data Systems to whom I spoke told me he hardly ever has the cover ON his machine, he is changing things so often. I didn't find either of these two solutions all that appealing to my higher aesthetic sense. What I needed was some way to add more slots.

That's when reality reared its ugly head. I found that an expansion chassis — just backplane, power supply, and case — often cost more than a whole clone computer. Then there are problems with the logic of the receiver-transmitter process that slows down operations, address con-

flicts between two popular add-ons, etc. So without really considering who I was asking, I posed the question of how to solve my slot shortage to our local systems guru. He immediately responded: network!

This was just about the time that 3.5" diskettes were introduced. I got an eZpc demo unit at the office. I needed some way to move files from my Z-148 to the eZpc and back since neither would read the other's disks. While I was at spring COMDEX in 1987, I had looked at both Lap-Link and Brooklyn Bridge. The thing that tilted the decision to Brooklyn Bridge was their RUN utility. This lets you remotely start a program on the slave machine and then do another on the host at the same time. I have been interested in parallel processing for some time and I thought of this as one crude, but possibly effective, way to get started.

I looked for a good mail order price, called, and within a couple of days had the two-headed cable, diskettes, and manual. Zenith assured me that the mouse port on the eZpc would probably work. I hooked everything up as per directions, but the host machine kept telling me "failure to establish communication". In its from-the-package form, Brooklyn Bridge tries to communicate at 115,200 baud. At this point, I knew I was in over my head so I called White Crane for technical support. They told me to try a slower speed. I tried the two slower speed patches — 57,600 baud and 38,400 baud — provided on the disk, but no dice. When I reported this to White Crane, I was told to open the patch file and change a byte value from 2 to 6. The person to whom I spoke told me that was what was being divided into 115,200. 115,200 divided by 6 would give me a



THE NETWORK



communication rate of 19,600 baud. If this didn't work, I could always try 9600 baud. Well, it did work.

Brooklyn Bridge consists of an executable program on the slave machine which can be invoked when desired and a device driver which is loaded at boot time on the host machine. Thus, when one types the command BRIDGE on the slave, one must then reboot the host to establish communication. The drives on the slave are treated as additional drives on the host. For example, if the slave has a floppy and a hard drive and the host has a floppy and a hard drive (my configuration) the host machine thought it had six drives: A, B, C, D, E, and F. If drive D was 3.5" and drive A 5.25", getting data from A to the D is as simple as COPY MYFILE D: <CR>. Additional software device drivers are included so you can temporarily attach the slave's printer port, console, and other COM port (if it exists) to the host machine. Except for the fact that transferring large files was a little slow, everything worked as advertised. I could have big jobs running simultaneously on both machines, if I wanted to.

Then, last summer I got a call from a fellow on the West Coast who said he would like to publish an updated version of a book I had done on computer programs for aircraft performance, stability, and control. He thought his market would

demand that the programs be in BASIC for the Macintosh. I winced at the prospect of recoding 4000 lines of FORTRAN programs into BASIC and he finally agreed that he would take them in FORTRAN. I bought a Macintosh Plus demo from our local college bookstore at a reduced price and set about learning how the darn thing works. (In addition to being anachronistic, I must also be dense because I found the Mac way of doing things not so easy or all that intuitive. The documentation borders on non-existence.) The first program I bought for it was Lap-Link Mac so that I could transfer the programs for my new book from my ZW-241 to the Mac. Lap-Link Mac works at 57,600 baud, so after 19,200 baud, I was impressed.

Lap-Link Mac always uses the IBM-compatible as the host and the Mac as the slave. It has a split screen so you can see the contents of the IBM directory on the left and that of the Mac folder on the right. That was a godsend. Just highlight the file you want to send (in either direction), press C for COPY, hit <CR>, and away it goes. So now I had, in effect, two, two-computer "networks": one at home and one at the office.

While doing the book I had the opportunity to test a beta version of AOX's Z-Master 386 speed-up board for the Z-241 and Z-248. (Imagine my surprise at the June issue of REMark which came to-

day to see Pat Swayne's review of the Z-Master board. It was fascinating to compare his response to the board with mine although my testing involved a Z-241 and his a Z-248. I had never looked into the differences between a Z-241 and a Z-248, but AOX told me there were quite a few besides the obvious change in clock speed.) My test of the Z-Master taught me the visual and functional superiority of Zenith's BIOS-based setup menu and the advantage of being able to boot from any drive. This is something I use frequently and I was surprised to find that everyone didn't use it. The AOX board has a Phoenix BIOS which is said to be "more compatible" with IBM, though I'm not sure there's much advantage to that. As Pat said, it's pretty plain Jane and I haven't found much that I'm interested in that doesn't run on the Zenith.

The version I tested did not have the final production version of the BIOS in it, so problems I had may not be present in the version now shipping. For me, at least, it complained about the keyboard and the video card, though it used both without a problem. (I was using an Everex EVGA video card with a Z-1490 monitor because the Z-449 card was not available locally when I bought my monitor.) It would not recognize my second hard drive or my extended memory. (It would also not work with Zenith's version of OS/

## PC89LINK

This new product allows you to transfer files between an H/Z-89/90 (running HDOS 2 or 3) and a PC compatible computer at baud rates up to 19200. (It also works between two '89s or two PC's.) Only a simple serial cable is needed. HDOS and MSDOS wildcards are supported, so sending "\*" sends the entire disk or subdirectory — this eliminates the main inconvenience of most file transfer programs. Order #150, \$25.00 postpaid (two disk set, one HDOS one MSDOS).

## HDOS 3 Printer Drivers

Our Ultimate Driver now supports HDOS 2.0, 3.0 and 3.02. New features include **programmable macros** and support of new Epson compatible escape codes. Under HDOS 3 and higher, supports UNLOAD and other new HDOS commands. The UPC driver, which prints H-19 graphics (also inverse, double and triple width and/or height), has also been updated for HDOS 3 support. If you already have one these drivers, send \$3 for one update, \$5 for both updates. If you don't already have our drivers, send \$25 for a **special combination** of BOTH drivers (you must mention this REMark ad!) Please specify hard/soft sector diskettes, and printer type: Epson/Compatible, NEC/C.Itoh, Okidata, MPI. Call or write if you have any questions.

## EGAD

Graphics and Text Screen Print Package for VGA, EGA, and CGA displays.

- Print any part of the screen ('crop box' lets you use the cursor keys to select any rectangular area).
- Enlarge graphics for emphasis (1, 2, 3, or 4 times).
- Color text and Color graphics print in color on color printers. On monochrome printers, graphics print in black and shades of gray (depending on printer resolution used).
- SET program lets you determine a print color (or gray shade) for each screen color. A different choice may be made for each text and graphic mode. (Pick 16 printer colors in Text mode, 4 colors in CGA 320x200 graphics, 2 colors in CGA 640x200, 16 colors in 640x350 and 640x480 EGA and VGA modes.)

The distribution disk includes drivers for these printers: Epson and compatibles, Star NX-1000 Rainbow and Epson JX-80 color, Xerox 4020, Dataproducts 8020, NEC-8023 and C. Itoh 8510. A version for the HP LaserJet series will be available in September 1989. Order # 270, \$25.00 postpaid.

**Lindley Systems 4257 Berwick Place,  
Woodbridge, VA 22192 (703) 590-8890**

Call or Write for Catalog and Product Information.

2 which I have installed as the alternate operating system on my C: drive. I notice that this dual boot feature is so well thought of that its being sold as a separate utility for IBM and Compaq machines by a third party vendor, although it was part of Microsoft's OS/2 development kit. I have my Seagate ST-251 partitioned as drive D: for MS-DOS 3.21 plus a boot drive partition as drive C: under MS-DOS 3.1. The original C: drive then becomes D: drive. Doing things this way allows me to boot either of three operating systems from the hard drives without using an ASGNPART in the AUTOEXEC.BAT file.) On the Norton SI test, the Z-Master gave the same results as a Z-386, but tests with real programs showed this to be true only as long as the code would fit within the 8K onboard cache and no disk accesses or video writes were necessary, both of which have to use the existing 6 MHz buss.

While trying to decide whether to upgrade my ZW-241 by this route, I was able to acquire a used Z-386 and, presto, I had need for more networking. I found that White Crane Systems had just announced a parallel version of Brooklyn Bridge and I, as the registered owner of the serial version, could buy the software and parallel cable for \$45. With fast machines and bidirectional parallel ports, this setup is capable of transmission rates of 50K bytes/sec. It has all the same utilities as the serial version in addition to which you can hook another machine to the serial port and thereby make a three-machine "network".

That situation came upon me sooner than I had expected. I have had another textbook in extended gestation and its publisher has now demanded that I supply him with machine-readable copy. Since the original was done on a now-extinct dedicated word processor, I desperately looked at Optical Character Recognition to avoid retyping. I found a hand scanner and OCR software package for \$300.00 (The DFI HS-3000 scanner and Carets By Hand software.) I knew that this package required lots of hard disk space, more than 512K of available RAM, and not much else on the buss since its address conflicts with LPT2 and it uses IRQ 3. I therefore cannibalized my Z-148 for its drives, keyboard, and memory and added these to a motherboard, case, power supply, I/O-disk controller card, and video card and, voila!, another computer on the network.

I had originally wanted to avoid having two keyboards and two monitors on the desk with the Z-241 and the Z-386. I finally managed to find an inexpensive switch that would switch both a keyboard and a monitor at the same time. So I tried it. It worked as advertised — except that the internal shielding must not have been as good as was needed since the picture

on my ZVM-1240 monitor had waves in the center on some software packages and, except that, after a week's use in this mode the Z-386 had to go to the shop for what turned out to be a blown fuse on the backplane, due, I suppose, to the transients created by switching the keyboard from machine to machine while both were active. A little rearrangement of the hardware and some carpentry gave me enough room for two keyboards side by side, two TTL monitors, and a Z-1490 VGA monitor on top of the stacked Z-386 and Z-241. (The Z-241 has two video cards.) Behind one keyboard is a Z-207-6 subsystem consisting of a pair of Shugart 860 1/2-height 8" floppy drives. This is operated by a second floppy disk controller in the Z-241. A Compaticard II which had driven an external 3.5" floppy from the Z-241 was replaced by an I/O card and the 3.5" drive was installed in the Z-386 as drive B:. The other keyboard sits in front of the stacked computers. An additional 3.5" drive was added to the OCR machine so that it has a 1.44 Meg drive, a 360K drive, and a 20 Meg hard drive. This machine sits on a table behind the swivel chair which is in front of the desk. All I need to do OCR is turn my chair 180 degrees. I have put a keyboard drawer under the OCR machine so that when I'm scanning I can easily push the keyboard out of the way.

If I want a feeling of real meglomania

I can have them all — all 14 drives — (including the vdisk in the extended memory of the Z-241) talking to each other on the same virtual machine. I can also execute Lap-Link Mac which is now connected to one of the serial ports of the Z-386 from the Z-241 using Brooklyn Bridge. Now if I can only get the networked IBM PC/RT that now sits on my desk in the office working as I'd like . . .

Last summer I got a copy of Microsoft's Fortran Version 4.1 at a special academic price. I had intended to see if I could take advantage of its virtual machine capabilities (when run under OS/2's protected mode) to run some of the big programs I had on the University's mainframe. With the rush job to do the book, I didn't get around to it. Now comes an announcement from Microsoft of version 5.0 with new bells and whistles. Then when you read the fine print you find

1. It will run in protected mode only under Presentation Manager (\$339 list from Zenith).
2. Microsoft wants \$100 for the upgrade.
3. You need at least 2.5 megabytes of memory to run PM.

Here's the real dilemma: My Z-386 has 2 Megabytes and, as Pat Swayne mentioned, Zenith memory boards are very expensive (university price of more than \$1800 for the 4 megabyte board for the Z-386). I don't know of third party memory boards that will work in this machine.

## Most home finance programs have serious limitations.

Like the dodo bird, they're slow, awkward and not very smart. You'll waste eons of time, only to realize they're dead ends.

HFS-III, the Home Finance System, is fast, loaded with features and intuitively smart.

- Easily manage up to 100 checking, saving, CD, IRA and other accounts.
- Print checks on any form.
- Organize financial summaries and activity reports based on 100 expense and 15 deposit codes you define.
- Flag tax-deductible or medical expenses.

In short, control your finances to save money, save time and simplify taxes.

Find out why so many leading computer authorities and users are calling HFS-III the "superior species." See it today at participating Heath/Zenith Computers and Electronics Centers, or call toll free for more info.



### Only \$49<sup>95</sup>

Add \$3.50 handling & shipping.  
Iowa residents add 4% sales tax.

Requires: DOS 2.0 or higher, IBM® PC/XT/AT or compatible or Zenith® Z100 computer; 2 disk drives or hard disk, 256K RAM.

Reader Service #137



Jay Gold Software, Inc.  
P.O. Box 2024  
Des Moines, IA 50310  
(800) 541-0173



# Z-386 (16MHz) Users

## Soon You'll Have a Choice!

It's been a long time in coming! *Finally*, an inexpensive approach to upgrading the memory in your Zenith Z-386 computer system is coming!

RamTop-386 is a *new* daughterboard from the people who brought you the RamPal and RamTop products for other Heath/Zenith computer systems. When plugged into a Z-505 card (the 1Meg RAM card which is already in your system), RamTop-386 will allow you to install higher capacity RAM chips to convert the board to either a 4Meg (in the basic configuration) or a monster 16Meg (with add-on) RAM card - without losing a single buss slot!

You say you've already bitten the bullet and paid the cost of a Z-515 4Meg add-on board? *No problem!* You can use RamTop-386 on your Z-515 to convert THAT card to a 16Meg RAM card as well (with the optional add-on)!

Just imagine! You can fully populate the RAM in your system using just one board - and no buss slots! Or... you can add a few megabytes now, and finish filling it later on, when memory is cheaper. The choice is all yours! Either approach is bound to be cheaper than before!

For more information, call or write us at the location below. And, while you're at it, ask us for a FREE copy of our unique Disk-Based Catalog.

## RamTop-386

*The Latest Best-Seller from First Capitol Computer*



#16 Algana Drive  
St. Peters, MO 63376  
Orders: 1-800-TO-BUY-IT  
Tech: 1-314-447-8697

Reader Service #105

## REMOVABLE HARD DISK DATA STORAGE



**DMA**  
TECHNOLOGIES

P.O. Box 1236  
601 Pine Avenue  
Goleta, California 93116

### FOR YOUR ZENITH Z-150, Z-248, Z-286, Z-386 MICROCOMPUTER SYSTEMS

- RANDOM ACCESS
- 10MB OR 20MB HARD DISK CARTRIDGES
- MAXIMUM DATA SECURITY
- IMAGE BACK-UP
- 100% MEDIA INTERCHANGEABILITY
- SCSI OR ST506 INTERFACE
- NOVELL COMPATIBLE
- EXTENDED 24 MONTH WARRANTY

For product information see your local Zenith dealer or call 1-805-964-0733.

In California call 1-800-654-8590.

FAX inquiries: 1-805-964-0734.

VISA & MASTERCARD accepted.

Also available on GSA contract GSOOK-88-AGS-5087.

Reader Service #197

Third party boards will work in the Z-241, according to recent correspondence with Bill Adney. OK boards can be had for \$139 and up from mail order houses. The latest prices I've seen for 1 Meg DRAMS is about \$17 per 120 nanosecond chip. Since you need at least two banks of chips on a board regardless of type, that means an expenditure of at least \$306 for 2 Meg of 1 Megabit chips or \$262 for 1 Meg of 256K bit chips. Even with university prices on OS/2 version 1.1, this is still well over \$600 just to try the new version. I'm afraid a decision on that one will be a while.

As your readers probably know by now, Zenith has introduced new Z-386/25 and Z-386/33 machines which have fundamentally different architectures from the 16 MHz Z-386. They have sent around to selected "partners" promotional and technical literature on the machines and have included a 1.44 Meg diskette which holds a graphics demonstration and a description of the features of the machine (one disk for the 386/25 and one for the 386/33.) You're told at the beginning that you need a 1.44 Meg drive and VGA to view the disks. What you are NOT told is that the video card CANNOT be a Z-449! Unfortunately, all the Zenith machines we have acquired at the University through last January at least that have 1490 monitors also have 449 video cards. They are especially impressive graphics demos which I would love to use to brag on Zenith machines. Unfortunately, I can't have everyone out to my house where my Everex EVGA card will display it.

Finally, let me compliment you on your zapping Bugs column. I read last month about the wavering image problem with the Z-1490 just at the time it happened to one of my two 1490's. The prior knowledge of what was wrong gave me a little more time to accept the \$94 university hardware services charged to replace the power supply.

Frederick O. Smetana, Professor  
Mechanical and Aerospace Engineering  
North Carolina State University  
Raleigh, N. C. 27695-7910

## A Z-100 for the Classroom

Dear HUG:

By September 1989, I will be teaching 5th through 8th grade science and I am in need of a Z-100 or Z-150 system (computer, printer, and monitor) for student use in the classroom. The computer will allow the students to write programs, store and chart data, and run different simulation programs. I am sure other uses will surface as the class progresses.

If you care to donate, please call (313) 651-3859 or write. I will gladly ac-

cept portions of any system as I might be able to put a working system together from various sources. I will provide you with tax deduction numbers and any information you will need to claim the donation.

Thanks very much,  
Edward A. Byrnes  
412 Taylor  
Rochester, MI 48063

## PC Tools Review

Dear HUG:

I was very much interested in the review of PC Tools Deluxe in the July '89 issue of REMark. I've been a very satisfied user of PC Tools for a few years now and would second the strongest possible recommendation of it. I was particularly interested in the discussion of one of the included programs, PCFormat, because I've never had it work correctly. Right now, I've got PC Tools Deluxe 5.1, and I'm very pleased with it, but I'd sure like to have PCFormat work.

The program will run normally until just after it displays the "Format Complete" message. Then, rather than displaying the total and usable amount of space on the disk, the program hangs my computer. This happens whether I'm format-

ting 3.5" or 5.25" disks, running the CPU at high or low speed, or with a full set of memory resident programs and drivers or a clean boot from a floppy. I've written Central Point about this several times and I've always gotten responses. However, I've never gotten a fix.

My computer is an H-151 with the Wildfire turbo kit, including the NEC V-20 processor, 3.5" and 5.25" floppy drives and a Seagate ST225 hard drive with an OMTI controller, MS-DOS 3.3+ and the 3.0D ROM monitor chips. However, as I said, the problem is not new. PCFormat didn't work for me in PCTools 4.3 (or older versions, though I can't remember when it was introduced), with MS-DOS 3.21 or with older ROM chips.

If anyone has an idea, I'd very much appreciate the help.

Thank you very much.

Sincerely,  
Lt. Col. Michael Hansen  
5901 JFK Boulevard, Apt. 5625  
North Little Rock, AR 72116



## CLASSIFIED ADS

**Z-100 LOW-PROFILE**, 768K, full video RAM, 10MB, 5-1/4", Monitor, Documentation, Software. \$1000 or Make Offer. Evenings (703) 371-3416.

## VOICE MASTER KEY<sup>®</sup> VOICE RECOGNITION SYSTEM FOR PC/COMPATIBLES & TANDY 1000 SERIES A FULL FEATURED VOICE I/O SYSTEM

GIVE A NEW DIMENSION TO PERSONAL COMPUTING. . . The amazing **Voice Master Key System** adds recognition to just about any program or application. Voice command up to 256 keyboard macros from within CAD, desktop publishing, word processing, spread sheet, or game programs. Fully TSR and occupies less than 64K. Instant response time and high recognition accuracy. Voice recognition tool-box utilities are included. **A genuine productivity enhancer!**

**SPEECH RECORDING SOFTWARE.** . . Digitally record your own speech, sound, or music to put into your own software programs. Software provides sampling rate variations, graphics-based editing, and data compression utilities. Create software sound files you can add to macros for voice recognition verification response. **a complete, superior speech and sound development tool.**

**SOFTWARE CONVERSION CODES.** . . The **Voice Master Key System** operates a growing list of third party talking software titles using synthesized phonetics (text - to - speech) or digitized PCM, ADPCM, and CVSDM encoded sound files. **Voice Master Key system does it all!**

**EVERYTHING INCLUDED.** **Voice Master Key System** consists of a plug-in card, durable lightweight microphone headset, software, and manual. Card fits any available slot. External ports consist of mic inputs and volume controlled output sockets. **High quality throughout, easy and fun to use.**

**ONLY \$149.95  
COMPLETE**

**ONLY \$89.95 FOR TANDY 1000 SL/TL MODELS  
SOFTWARE ONLY—REQUIRES TANDY BRAND  
ELECTRET MICROPHONE.**

**ORDER HOTLINE: (503) 342-1271 Monday-Friday, 8AM to 5 PM Pacific Time**

Visa/MasterCard, company checks, money orders, CODs (with prior approval) accepted. Personal checks subject to 3 week shipping delay. Specify computer type and disk format (3 1/2" or 5 1/4") when ordering. Add \$5 shipping charge for delivery in USA and Canada. Foreign inquiries contact Covox for C & F quotes. **30 DAY MONEY BACK GUARANTEE IF NOT COMPLETELY SATISFIED. ONE YEAR WARRANTY ON HARDWARE.**

**CALL OR WRITE FOR FREE PRODUCT CATALOG**



**COVOX INC.**  
675 Conger Street, Eugene, OR 97402  
TEL: 503-342-1271 • FAX: 503-342-1283

Reader Service #196



# QUIKDATA - 12 YEARS OF H/Z SUPPORT!

For all your H/Z 8-bit and PC/XT/AT needs

## ACCELERATE YOUR PC!

From Sota Technologies, Inc., the fastest and most proven way to breath new life in your Heath/Zenith PC/XT computer, giving it -AT compatible speeds! Turn your turtle into a -286 rabbit with a 12.5 Mhz 80286 accelerator board. Complete with 16K on-board CACHE RAM for dramatic speed increases.

There are other ways to speed up your H/Z PC/XT computer, but you spend too much and get too little. The **286i** is the effective solution, making your H/Z150/160/150/158/159 series of computers, or any general PC/XT computer faster, in many cases, than a standard IBM AT type computer! Simple half-card plug-in installation with switchable speed control. **You won't believe your stop watch!**

**EXP-12 - \$379**

## MEMORY UPGRADES

Note: All memory upgrades come without memory chips. Call for current chip pricing.

**Z150MP - \$19** Will allow you to upgrade your H/Z150/160 to up to 704K on the main memory board, using up to 18 256K DRAM chips.

**MEGARAM - \$43** Upgrades your H/Z150/160 series with up to 704K of main memory, and about 512K for RAMDRIVE memory. Includes documentation, software RAMDRIVE disk, PAL and jumper wire. For the full 1.2 megs total memory, 45 256K DRAM chips are required.

**ZMF100 - \$53** Will allow you to upgrade your H/Z110/120 (old motherboards; with p/n less than 181-4918) to 768K system RAM. Requires 27 256K DRAM chips.

**Z100MP - \$76** Similar to ZMF100 above, but for new motherboards with p/n 181-4918 or greater.

## WINCHESTER UPGRADE KITS

**PCW20 - \$295** Complete winchester setup for a H/Z150, 148, 158, 159, 160, PC etc. Includes 21 meg formatted half-height Segate ST-225 65ms drive, WD WX1 winnie controller board, cable set, documentation.

**PCW30 - \$380** 32 meg with 38ms Segate ST-138.

**PCW40 - \$449** 42 meg with 37ms Segate ST-251.

**PCW80 - \$695** 80 meg with Segate ST-4096 full size drive.

**EWIN20 - \$555** Complete 21 meg winchester package for the Z100. Includes S100 host adapter card, WD winchester controller, Segate ST-225 winchester drive, cables, software and documentation. We also have this in 32 megs (\$640), 42 megs (\$689), and 80 megs.

## BARE WINCHESTER DRIVES

**ST-225 - \$229** 65ms half height bare winchester with 21 megabyte storage capacity.

**ST-251 - \$395** 37ms autopark half height bare winchester with 42 megabyte storage capacity. Good choice for -AT compatibles.

**ST-251-1 - \$429** Same as ST-251 but 28ms speed.

**ST-4096 - \$619** 28ms autopark full height bare winchester with 80 megabyte storage capacity. Best choice for -AT compatibles.

## ANY DRIVE IN YOUR PC/XT/AT

With the CompatiCard, you can install up to four additional drives, of any type in your PC/XT/AT computer. Add a 1.2 meg 5" floppy, or a 1.44 meg 3.5" floppy, or any other drive, including 8" to your system. The CompatiCard I will handle up to two drives, and the CompatiCard II will handle up to 4 drives. Additional cables and external enclosures may be required.

**CCARD1 - \$89**

**CCARD2 - \$109**

## DOES YOUR COMPUTER KNOW THE TIME?

We carry the SMART CLOCK for any PC/XT computer, including the Z100 series. Simply plug it in the system ROM socket, and plug the ROM into the clock module. Self contained sealed lithium battery for 5-10 year life. Includes software for the clock/calendar.

**SMARTCLK - \$35** (add \$2 for Z100 installation for spacers)

## ZENITH COMPUTER SAMPLE

**Zenith AT STARTER.** Start with this basic unit and build your own. Includes 12Mhz Z248 (80286 processor) with two serial ports, one parallel port, 1MB RAM, 5" 1.2 meg floppy, no video card. If you desire we can install a 42mb 28ms hard disk for an additional \$395, or a 80MB 28ms hard disk for an additional \$595. 16-bit paradise VGA card for \$319, Paradise EGA card for \$189.

**Z24812S1 - \$1595** (starter system)

**Zenith Z159** 8Mhz, 640K RAM, 101 keyboard, EGA video, dual floppies.

**Z159-S1 - \$1295** (\$1549 for single floppy 21mb hard disk unit)

**Zenith Laptop 184 SuperSport** with 20MB hard disk - \$2395

Call or write for a quote on any Zenith computer system.

## OTHER STUFF

Quikdata also carries spike protection filters, backup power supplies, modems, printers, disk drives, drive enclosures, cables and connectors, laptop batteries, monitors, memory cards, memory chips and ICs, joysticks, accessory cards, serial and bus mouse, the complete Zenith line of computers, parts and accessories, a wide variety of useful and most popular software and much more!

## 8-BIT

We carry a full line of hardware and software products for the H8/H89/90 computers. This includes diskettes, printers, H89 working boards and parts, some H8 parts, etc. Call or write to obtain a catalog of all our 8-bit goodies. Here's a sample.

**H37 SOFT SECTOR CONTROLLER** \$179

**H8 SOFT SECTOR/MINNIE CONTROLLER PACKAGE** \$275

**TM100-4R** Tandon DS DT 96 TPI refurb disk drive \$89

**SIEMENS DRIVES** (refurbs) for H8/H89 \$49

**Z47 REMEX DRIVES** slaves and masters - special \$89

## H-SCOOP

Of course, don't forget about the only independent source of Heath/Zenith related information you can obtain - our monthly newsletter, **H-SCOOP!** Just \$24 for a 12-issue year (\$28 Canada, \$35 foreign), it will help you get the most from your computer investment. Get sound technical advice, helpful hints, find out what the problems are, fixes, reports, reviews, information from other subscribers, classifieds and much more.

Call or write in to place your order, inquire about any products, or request a free no obligation catalog. VISA and Master Card accepted, pick up 2% S&H. We also ship UPS COD and accept purchase orders to rated firms (add 5% to all items for POs). All orders under \$100 add \$4 S&H. Phone hours: 9AM-4:30PM Mon-Thu, 9AM-3PM Friday. Visit our bulletin board: (414) 452-4345

## QUIKDATA, INC.

2618 PENN CIRCLE  
SHEBOYGAN, WI 53081  
(414) 452-4172

*William M. Adney*

*P.O. Box 531655*

*Grand Prairie, TX 75053-1655*

*Copyright © 1989 by William M. Adney. All rights reserved.*

# Powering Up Vol. 2

This article begins the continuation of the "Powering Up" series that will help you get more out of your computer system. The primary objective of the first group of articles published in this series was to discuss some of the various topics you need to know to use and upgrade your computer system. Although this series was originally intended to help beginning users understand the need-to-know non-technical information about their computers, many advanced users have found that the articles provide a good reference for information not available in the standard documentation. In case you missed one or more articles in the original series, the complete set of the original "Powering Up" articles is available in a 150-page book from HUG (Heath/Zenith Users' Group), p/n 885-4604.

Like its predecessor, *Powering Up* (Volume 2) is intended to provide a series of practical "How-to" articles related to using your PC compatible computer, especially a Zenith or Heath computer system with Zenith MS-DOS. Some articles, like this one on the DOS EDLIN command, will focus on specific commands and software supplied with virtually all versions of DOS (MS-DOS and PC-DOS), regardless of manufacturer. Other articles, such as the next one on "How to Use the Zenith ROM Commands", will focus on some unique hardware capabilities that you can use with a Zenith (or Heath) computer. And additional articles will explain how to connect serial devices to your computer, how to choose a programming language, and what a data base is and how you can use it.

All articles in this continuing series will assume that you understand comput-

er system basics discussed in the original *Powering Up* book. In this series of articles, Chapter numbers will refer to the original *Powering Up* book in which you will find additional information about a topic. For example, this article mentions batch files and CONFIG.SYS, and information about these files can be found in Chapter 7 about "Batch files and CONFIG.SYS" in *Powering Up*.

This article will help you learn how to use the DOS EDLIN command. Many people shy away from learning how to use EDLIN because of the way it is presented in their DOS manuals. Actually, EDLIN is pretty easy to use, and it has the added advantage that it is supplied with virtually all versions of DOS, not to mention OS/2. But what is the real advantage of learning how to use EDLIN?

## Why You Need to Know About EDLIN

For technical reasons, some DOS files, such as batch files (e.g., AUTOEXEC.BAT) and CONFIG.SYS (Chapter 7 in *Powering Up*), must be ASCII files. And because an ASCII file MUST contain only certain types of characters (generally called "printable" characters for the DOS PRINT command), you may find it difficult or impossible to use a word processor to create or change an ASCII file because many of those programs add unique characters to a file for special formatting, such as bolding or underlining or other special features. From a user perspective, an ASCII text file generally only contains the printable characters like A-Z, a-z, numbers, and digits (0-9). If you forget to save a batch file or a configuration file in an ASCII format, you will find that, in most cases, the file simply cannot be read by

DOS, and whatever you intended will not work in the system. In extreme cases, a batch file or configuration file that contains non-ASCII characters can even cause a system freeze during the boot process or during execution. Although some word processing programs can work directly with ASCII files, such as WordStar's non-document mode, a number of today's word processors have so many features that it is sometimes difficult to work with plain ASCII files. On the other hand, EDLIN is designed to specifically work with ASCII files. To show how important a basic knowledge of EDLIN can be, consider the following situation.

Because you have a personal computer at home or on your desk at work (or both), you are generally viewed as the resident computer "expert," whether you are or not. A friend asks you to help set up a new computer system with some batch files, and you know you will also have to make some changes to the CONFIG.SYS file. Unfortunately, you are not able to easily help your friend because you don't know how to use the "free" word processor, called Wazoo-Write, that was included with the computer. If you know EDLIN, setting up some batch files is really easy, no matter what brand of computer or DOS version you are using. But before we get into the specifics of EDLIN, let's review one other topic that is essential to successfully using the EDLIN command.

## The Function Keys

In some versions of the DOS manuals, you will find the editing features of the function keys referred to as "Intraline" editing commands which means that

# How to Use the EDLIN Command

these keys can only be used to edit the characters within a single line, such as the DOS command line (Chapter 4). Aside from their use in editing the command line, function key commands are also an essential part of using the EDLIN command, and they work within EDLIN just like they do on the DOS command line. Although a brief tutorial on the use of function keys was presented in the first *Powering Up* book (Chapter 4), let's go through a brief review of how you can use these keys on the DOS command line, as well as within EDLIN (and the DEBUG command too).

Table 1 contains a list of the Editing Function Keys for your PC compatible computer. In case you are not familiar with these functions, I suggest that you might find it helpful to power up your computer and practice along as we proceed.

Function	Key
Copy one character	F1
Copy up to character	c F2c
Copy remaining characters	F3
Skip to character	c F4c
Skip one character	DEL
Insert mode	(toggle)
Enter	INS
Exit	INS
Define new template	F5
Void template changes	ESC
End-of-file character	F6 (or Ctrl-Z)

**Table 1**  
**Editing Function Keys**

Assuming that you have your computer running and you can see the command prompt (e.g., A>), type the following: 1234567890. Then press RETURN to store the data in the template (Chapter 4), and you will see a "Bad command or file name" message that can be ignored.

Press the F1 key once. You will see the 1 appear on the command line. Press F1 again (to copy the next character), and you will see 12 on the command line. Press the F3 key once, and all of the remaining characters will appear on the command line so that you will see the 1234567890 that you originally typed. Press RETURN to clear the command line, and the "Bad command or file name" error message will appear again. Because this error message will occur every time you press RETURN, I won't mention it again.

The next copy function is slightly different. Press F2, type a 6 on the main keyboard, and 12345 will be displayed on the command line. That's how the "Copy up to character c" function works — it copies all characters up to, but NOT including, the character stored in the template. Press F3 to copy the remaining characters, and press RETURN. So much for the copy functions.

Now let's try the skip functions. Press DEL (to skip one character), then press F3. You should see 234567890 since we "skipped" the first character with the DEL key and copied the remaining ones from the template to the command line. Press RETURN.

The "Skip to character c" F4 key works just the opposite of the F2 key. Press F4, then type a 6, and you will see that nothing happens yet. Now press F3 to copy the remaining characters, and you should see 67890 on the command line. Press RETURN to get back to the DOS prompt. Now you have the characters 67890 stored in the template (memory). You can prove that by pressing the F3 key (followed by a RETURN) to display them.

At this point, you should have only the command prompt (e.g., A>) displayed on your terminal. We'll use the INS key to activate the insert mode and insert the digits 1 to 5 in front of the template. First, press the INS key one time. Then, type 12345, press F3 to copy the remaining template characters, and you will see the entire 1234567890 sequence displayed. Now press RETURN to store the complete sequence in the template.

One note about the insert mode — you automatically exit the insert mode when you press any other editing key, such as F3 in this example. Because the INS key can "toggle" the insert mode on and off like a light switch, you can also exit the insert mode by pressing INS again to perform more complicated editing functions. In this example, you did not have to press INS to copy the remaining characters on the line because pressing the F3 key automatically ended the insert mode.

All of these editing key functions are easy if you remember that you are making changes to the template, which is just a special area of the computer's memory reserved for this specific purpose. Some documentation may refer to the template as a "command line buffer", which is the same for purposes of this discussion.

To be sure we're all at the same place, press F3 and you should still see the 1234567890 that was saved as part of the insert mode example. If not, type those numbers in again and press RETURN to store the line in the template.

Press F3 again to display the number sequence, and type the word VOID. Now try the "Void template changes" (i.e., "ignore" all changes) function by pressing the ESC key. The system should print a slash (/) at the end of the line, jump to the beginning of the next line, and wait for input. Press F3 again, and you will see the original 1234567890 number sequence again without the word "VOID" following it. The void function of the ESC key essentially tells your computer to NOT store any changes that you typed, and the original template contents are retained.

The easiest way to think of the "Define new template" function key F5 is to remember that it saves whatever you type in the template (like the RETURN key), but it does NOT cause command execution (unlike the RETURN key). This key is particularly useful in editing a complex line to help you see where you are. When you press the F5 key, the system will print an "at-sign" (@) to indicate that the displayed line is stored in the template, jump to the beginning of the next line, and wait for input. The F5 key is normally used with the other editing keys, such as F1 (copy one character), F3 (copy remaining characters), and the INS or DEL keys. To see how this works, let's take a simple example by creating the original 1234567890 number sequence in small steps.

Beginning at the command prompt, type 12290. Wait — don't press anything yet. The first problem is the "extra" 2, so let's get rid of it. Press F5 to store that line. The system will store the 12290 characters, print an at-sign at the end of the line, and jump to the beginning of the next line. Press F1 to copy the first digit (the 1), press DEL one time to skip the first 2, and then press F3 to copy the remaining characters. If you have followed these instructions exactly, you should see 1290 on that line. Now press F5 to store that line. Press F3 again to verify that 1290 was stored in the template, and press F5 again to begin a new line.

The rest is easy. Press F1 two times to copy the first two characters (i.e., 1 and 2). Press the INS key to activate the insert mode, and type 345678. Press F3 to copy the remaining characters, and press F5 to store the template. To prove that your template is now correct, just press F3, and you should see the complete 1234567890 sequence. Press F5 to return to the DOS command prompt.

Perhaps you will notice that a discussion of the "End-of-File" F6 (or CTRL-Z) key has not been included here. It makes more sense to look at the uses for that key when we look at EDLIN. You may be wondering what all of this has to do with EDLIN. Well, these keys are not only useful for editing the DOS command line, but knowledge of them is also required for the effective use of EDLIN, and that's the reason for this brief review.

### Using EDLIN Quickly

Most people find that the easiest way to learn software is to use it. If you haven't already, you will probably want to power up your computer to follow along with this example. Even if you are not familiar with EDLIN, you should be able to do this little exercise in 10 minutes or less. We'll create a simple batch file, called STARTUP.BAT that contains the following commands:

```
DATE
TIME
```



To make things easy, create the file in the current subdirectory on the current drive. In this example, I have assumed that you are using drive A so that you will enter the following command line:

```
A:\ ==>EDLIN STARTUP.BAT (Press RETURN)
```

EDLIN will display a "New file" information message followed by the asterisk (\*) or star prompt as it is sometimes called. The entire display looks like:

```
A:\ ==>EDLIN STARTUP.BAT
New file
*-
```

The star (\*) is the EDLIN command prompt and indicates that EDLIN is in the COMMAND mode which will accept the subcommands. The cursor position is shown as the underline character following the star prompt. All commands and information you need to type are shown. Create the STARTUP.BAT file by typing I (Insert subcommand), and press RETURN to execute it. You should see the following:

```
A:\ ==>EDLIN STARTUP.BAT
New file
*I (Press RETURN)
```

The 1 followed by the colon is the line number in the file. It is slightly indented as shown, and the line number indicates you are in the EDIT mode. Lines are automatically numbered by EDLIN, and the colon is used as a separator. Line numbers are not stored in the file like they are in BASIC. The asterisk or star (\*) following the line number is the current line number indicator that will be discussed later in the article. The cursor shows that EDLIN is ready for input data to be typed. Now enter the DATE and TIME commands. If you make any mistakes, simply press the Backspace key to correct them. Press RETURN at the end of each line. Your display should now look like the following:

```
A:\ ==>EDLIN STARTUP.BAT
New file
*I
  1:*DATE (Press RETURN)
  2:*TIME (Press RETURN)
  3:*_ (Press CTRL-C)
```

That's the end of the file, and you do not want anything on line 3. All you need to do to exit back to the EDLIN star prompt is press CTRL-C or CTRL-Break. In case you are not familiar with this convention, a hyphenated command, such as CTRL-C, means to press and hold the CTRL key while pressing the next key — the letter C, in this case.

To verify what you entered, type an L (List subcommand) at the star prompt followed by a RETURN. Because you must always press RETURN to execute an EDLIN subcommand or save a line in the file, it will not be shown in the remaining examples. To save the data, type an E (End subcommand), and press RETURN. EDLIN will exit, the STARTUP.BAT file will be saved, and you will see the DOS com-

mand prompt. If you followed these steps correctly, you should see the following:

```
A:\ ==>EDLIN STARTUP.BAT
New file
*I
  1:*DATE
  2:*TIME
  3:*C
*I
  1: DATE
  2: TIME
*E
A:\ ==>_
```

That's it. You have created your first file with EDLIN. You started EDLIN with a command line, created a file with the Insert (I) subcommand, added the DATE and TIME commands, exited the insert mode with CTRL-C (or CTRL-Break), and reviewed the file with the List (L) subcommand. Then you saved the file to disk with the End (E) subcommand. Now that you generally know how to create a file, we will look at some additional details of EDLIN and its subcommands.

### More on the EDLIN Command

Refer to Figure 1 for the most general form of the EDLIN command.

```
EDLIN [d:] [\path] filename.typ[/B]
```

**Figure 1**  
**EDLIN Command Syntax**

When you use the EDLIN command, you MUST include a file name on the command line as shown. If you do not include a file name on the command line, then EDLIN will display the "File name must be specified" error message and return you to the DOS prompt. Like many commands, you can also precede the file name with an optional drive letter (d:) and path name (\path) as indicated by the brackets. If you do not enter an optional drive letter and/or path name, EDLIN will create or edit a file in the current directory on the current drive. Current MS-DOS versions (i.e., 3.2 and later) also allow you to edit binary files (e.g., COM or EXE) with the use of the optional /B parameter. This option is not particularly useful for most people, and it is only recommended for advanced users.

Before you use EDLIN for more complicated editing, there are a few things you should know. First, the maximum line length allowed by EDLIN is 253 displayed characters. Although the maximum is really 255, a carriage return/line feed sequence (2 characters) is required for each line of text.

For all Zenith and Heath PC compatible computers, lines containing more than 69 characters of text will "wrap around" to the next line on the screen, just like a word processor. You will find that you can enter up to 253 characters until you hear a beep of complaint from your computer.

### Editing and Fixing an Existing File

To return to the STARTUP.BAT file example, let's enter an additional line with the PROMPT command in the form of: PROMPT \$P \$Q\$Q\$G. Start EDLIN with the file name, and your display should look like:

```
A:\ ==>EDLIN STARTUP.BAT
End of input file
*-
```

Because you are editing an existing file, EDLIN displays the information message "End of input file", which means that EDLIN has loaded the entire file into memory.

Use the L subcommand to display the lines in the file, and your entire display should look like:

```
A:\ ==>EDLIN STARTUP.BAT
End of input file
*L
  1:*DATE
  2: TIME
*-
```

EDLIN has displayed your file with an asterisk (\*) or star following the line number on line 1. This star indicates the *current line number*, and you must keep in mind that it is not the same as the command prompt star. We look at the line number in more detail later, but now let's use a variation of the Insert (I) subcommand to insert the PROMPT command as line 3. Simply enter 3I (which means insert line 3) at the EDLIN command prompt so that your screen looks like:

```
A:\ ==>EDLIN STARTUP.BAT
End of input file
*L
  1:*DATE
  2: TIME
*3I
  3:*_
```

Type the PROMPT \$P \$Q\$Q\$G command on line 3, and press RETURN. EDLIN will display the next line number (line 4). Use a CTRL-C to return to the EDLIN command prompt, and your screen should now look like:

```
A:\ ==>EDLIN STARTUP.BAT
End of input file
*L
  1:*DATE
  2: TIME
*3I
  3:*PROMPT $P $Q$Q$G
  4:*C
*-
```

Whenever you exit the insert mode and return to the EDLIN command prompt star with CTRL-C, I suggest you always use the List (L) subcommand so that you can see exactly where you are in the file. But let's say that you do not want the equal signs (\$Q parameter) to display in the command prompt. That is, you want the prompt to look like "A:\>," not "A:\ ==>". To change that, what you need to do is enter a 3 (to edit line 3) at the EDLIN command prompt. If you did not enter the L subcommand, your display should look like:

```
A:\ ==>EDLIN STARTUP.BAT
end of input file
*L
1: *DATE
2: TIME
*3I
3: *PROMPT $P $Q$Q$G
4: *^C
*3
3: *PROMPT $P $Q$Q$G
3: *
```

There are two line 3s displayed. The first line shows the existing data on that line, and the second line is waiting for your input. Although there are any number of ways to edit this line, I like to use the F1 (copy 1 character) key to keep things simple. Press the F1 key repeatedly until your cursor is in position just before the dollar sign in the first \$Q parameter. Press the DEL four times to skip the four \$Q\$Q characters; then press F3 to copy the rest of the line. Now press RETURN to go back to the EDLIN command prompt.

Now let's assume that you really did not want to change the file and add the PROMPT command after all. Just enter a Q (Quit subcommand) at the EDLIN star command prompt and press RETURN. Because you will lose everything you typed during an EDLIN session, the program will display an "Abort edit (Y/N)?" message. If you type a Y, EDLIN will exit to the DOS command prompt, and all changes made during that editing session will be lost and NOT saved in the file. If you type an N, you will see the EDLIN star command prompt for additional commands.

You now know enough to use EDLIN effectively to create and change a file. In this section, you saw how to edit an existing file, how to use line numbers to edit an existing line, how to insert a new line in a file with the I subcommand, how to see the contents of a file with the L subcommand, how to save a file with the E subcommand, and how to abandon changes made to a file with the Q subcommand. These basic EDLIN subcommands are summarized in Figure 2.

#### Subcommand Description

<b>n</b>	Edit line number n
<b>E</b>	End editing and save file
<b>[n]I</b>	Insert line(s) in file [before line n]
<b>[begin-n] [,last-n]L</b>	List (display) line in file
<b>Q</b>	Quit editing and do not save changes

**Figure 2**  
**Basic EDLIN Subcommands**

As you can see, some of the subcommands, such as Insert and List, have optional parameters shown in brackets. When a line number is not entered, EDLIN begins any command at the current line number which you can see by

using the L subcommand by itself. The way that EDLIN uses line numbers is so important that it is worth looking at in more detail.

#### The Current Line Number

EDLIN is a line editor which is different than the "full-screen" editors that you are probably used to in a word processing program. Because EDLIN works with lines, it must maintain some kind of "line pointer" so that you and the program can keep track of which line is being edited (i.e., the current line).

EDLIN displays an asterisk or star (\*) following the current line number and colon to help you identify the current line. That can be very useful for some subcommands because EDLIN always begins an operation (such as a Search) at the current line number (indicated by the asterisk) unless otherwise specified. Use the L subcommand to see what the current line number is.

The current line number is especially important when you are using the I subcommand to insert a new line in an existing file. Let's say you have a file containing five lines and you want to insert a new line between lines 2 and 3. To do that, you would use the 3I subcommand because when the optional line number is specified, it tells EDLIN to insert a new line BEFORE an existing line that has the same number. In other words, the inserted line has the line number that precedes the I subcommand (e.g., 3, in this example). Some other subcommands — Copy, Move, and Transfer — have a destination line *dest-n* that is specified the same way.

It is important to remember that, even though EDLIN displays line numbers, they are not actually stored in the file. EDLIN always displays sequential line numbers regardless of the operation that you perform (e.g., delete a line). The way that EDLIN displays the line numbers which are not stored in the file, and "renumbers" them automatically when an operation is performed, is usually called dynamic line numbering.

You can edit any line in the file by simply typing the line number and pressing RETURN at the command prompt. The current line is always indicated by a star (\*) following the line number, and you can always check the current line number with the L subcommand. You can also edit the current line indicated by the star by typing a period (.) followed by a RETURN. And you can easily edit the line following the current line number by simply pressing RETURN at the command prompt. These are little shorthand tricks that you can use, and now let's take a look at some additional ways to see a file.

#### More Ways to See a File

In order to use a line editor, like

EDLIN, easily, it is important to know various subcommands that you can use to see the contents of a file. Although we have already used the List (L) subcommand in its most basic form, the most general List subcommand syntax is:

```
[begin-n][,last-n]L
```

If you wish to see a specific range of lines, you can use the optional beginning line number *begin-n* and the last line number *last-n* in the subcommand. The List subcommand does not change the current line number, so you can use it frequently to see what the current line number is.

When you want to scroll or page through a large file, you can use the Page (P) subcommand. Although the Page subcommand was introduced in MS-DOS version 2, it has not been included in the Zenith EDLIN documentation until version 3.2. The general syntax for the P subcommand is:

```
[n]P
```

or

```
[begin-n][,last-n]P
```

The Page subcommand is similar in usage to the List subcommand. If you wish to see a specific range of lines, you can use the optional beginning line number, *begin-n*, and the last line number, *last-n*, in the subcommand. Unlike the List (L) subcommand, the Page subcommand DOES change the current line number. You will find that the current line indicator (\*) is always the last line displayed on the screen.

#### Block Subcommands

Like most editors, EDLIN has the standard Copy, Delete, and Move block subcommands. Even though all three subcommands were introduced in MS-DOS version 2, Copy and Move have not been included in the Zenith EDLIN documentation until version 3.2.

The Delete subcommand allows you to delete a single line or a range of lines. The general syntax for the Delete subcommand is:

```
[n]D
```

or

```
[begin-n],last-nD
```

Be sure to always use the L subcommand first (to see what the current line is) if you use D by itself because the current line will be deleted when you don't enter an optional line number. When you want to delete a range of lines, the optional beginning line number *begin-n* defaults to the current line number if not specified. The comma preceding the last line number *last-n* is always required if used.

The syntax for the Copy and Move subcommands is quite similar. The general syntax for the Copy subcommand is:

```
,,dest-nC
```

or

```
[begin-n],[last-n],dest-n[,count]C
```

Similarly, the general syntax for the Move subcommand is:



```
,,dest-nM
or
[begin-n],[last-n],dest-nM
```

You can use the first general form of the Copy and Move subcommands to copy the current line number to a destination line number *dest-n*. In the second general form of each subcommand, the range to be copied or moved can be specified just like the Delete subcommand.

The Copy subcommand also has a *count* option. If a number is not specified for *count*, the subcommand default is to copy the line (or range of lines) one time. If you enter a number for *count*, then the Copy subcommand will copy the line or range of lines that number of times — a useful feature if you want to create a large file quickly.

### The Search and Replace Subcommands

EDLIN's Search and Replace subcommands can also be used to make changes in a file. The general syntax for the Search subcommand is:

```
[begin-n][,last-n][?][?]Soldstring
```

If a range of lines (i.e., *begin-n* to *last-n*) is not specified, the Search subcommand will begin the search at the current line (indicated by the \*) and search for the first occurrence of *oldstring* until it is found or the end of the file is reached, whichever comes first.

The optional question mark (?) is used to request a prompt in the form of "O.K.?" after the display of the line containing the *oldstring*. Since I generally want to search the entire file for something, my preferred form for using this subcommand is:

```
1, #? Soldstring
```

That subcommand example allows me to search the entire file for whatever the contents of the *oldstring* is. But what is that strange pound sign (#) doing in the middle of the subcommand? That is an EDLIN "shorthand" symbol which means a line number larger than the last one in the file — in effect, the entire file. You could accomplish the same thing by entering a line number larger than any in the file, say 999. I use the # only because it requires fewer keystrokes.

The use of the question mark in the Search command line allows you to control the search. As the Search progresses and an exact match is found, the line containing the match will be displayed with the "O.K.?" prompt on the following line. If you enter a Y, the Search stops; if you enter an N, the Search continues to look for the next exact match of the *oldstring*.

For example, let's say that I was looking for ECHO in a batch file. I can use the following subcommand: 1, #?SECHO. I type an "N" response to the "O.K.?" prompt until I find the exact line I am looking for; then I type a "Y" to abort the Search. Once you get the hang of doing

this, it really is pretty easy.

After you have once entered the entire Search subcommand with the *oldstring*, you can also just type an S to Search for the next occurrence of the *oldstring*.

The Replace subcommand is quite similar to Search. The general form of the subcommand is:

```
[begin-n][,last-n][?][?]RoldstringCTRL-Znewstring
```

The Replace subcommand will replace an occurrence of the *oldstring* with the *newstring*. Each of the text strings is separated by a CTRL-Z as shown. This subcommand is shown in the most general form although you also can press the F6 key to generate the CTRL-Z character.

Aside from the replace function itself, the Replace subcommand works exactly like the Search subcommand. If a range of lines is not specified, the Replace subcommand will begin the search (and replace) at the current line (indicated by the \*) and search for the first occurrence of *oldstring* until found or the end of the file is reached, whichever comes first. Then the *oldstring* text will be replaced by the *newstring* text.

The optional question mark (?) is used to request a prompt during the replace process and functions just as described for the Search subcommand. The use of the question mark prompt is particularly recommended for the Replace subcommand since it gives you complete control of the replace process.

After you have once entered the entire Replace subcommand with the *oldstring* and *newstring* text, you can also just type an R to search for the next occurrence of the *oldstring* and replace it with the *newstring*.

### The Transfer Subcommand

It is sometimes useful to be able to read or "merge" one file into another. For example, you may already have another batch file that contains the basic skeleton or other subcommands that you want to include in a new batch file. The Transfer subcommand allows you to do this quite easily.

The general form of the Transfer subcommand is:

```
[dest-n][T[d:]filename.typ
```

If the destination line number *dest-n* is not specified, the contents of the specified file (*filename.typ*) are transferred (i.e., merged) into your current file at the current line number indicated by the \*. When the destination line number is specified, the contents of the specified file are transferred to the current file beginning at that line number. Although you can specify an optional drive letter for the file, the Transfer subcommand still does not allow you to specify a path. If you specify a path, EDLIN does not recognize it as such and displays a "File not found" error message.

### Some Other EDLIN Tricks

One of the reasons that people don't seem to like EDLIN is because a few of the really practical tricks in using it are not documented very well, if at all, in any of the manuals. Some of the clues on these tricks are there, but they are very not explicit as to how you can really use them.

The first trick is how to change the current line number without affecting the contents of any line of the file. All you have to do is type the line number *n* at the command prompt and press RETURN twice. The first RETURN executes the subcommand in the command mode and displays the line in the edit mode. The second RETURN sends you back to the command mode.

You can also continue to press RETURN, alternating between the command mode and the edit mode, and you will see that this "cycles" through the lines in sequential order. In other words, you can edit sequential line numbers (and change the current line number) by simply pressing RETURN, and you really don't need to use the L subcommand in this case. This brings up an important point about how the RETURN key can be used.

When you are in the edit mode (with a line number displayed), pressing RETURN at the BEGINNING of the line does NOT CHANGE the contents of that line. For this feature to work properly, you must not use any of the Function Keys or type anything in the edit mode — just press the RETURN key to return to the command mode. If you do ANY editing on a line and then press RETURN, that line will be stored as displayed, as you expect.

You can also use this feature to move around quickly in a file that has more lines than can be displayed on a single screen. For example, you can always get to the top of the file by typing a 1 in the command mode and pressing RETURN twice. Or, you can get to the bottom of the file by typing the shorthand pound sign (#). Then use the L subcommand to display the lines in both cases. How to really use the RETURN key and these subcommands is not obvious if you read the DOS manuals.

Although it may not be obvious, you also need to know that the Function Keys work the same way in EDLIN for both the command mode and the edit mode. The Function Keys also perform identical editing functions on a DOS command line, and Function Keys can also be used in the DEBUG command too.

One other trick of how and when to use the CTRL-C (or CTRL-Break) may also help you, especially when you are learning how to use EDLIN. When you are editing a line, sometimes it gets so messed up that it is really easier to start over. You can press CTRL-C anywhere within a line (including the beginning) to abort those



changes and return to the command mode. Then use the dot (.) shorthand command to "re-edit" the current line. In other words, CTRL-C will abort changes made in the edit mode, as well as aborting commands (e.g., Search, Replace or Insert) in the command mode.

All of these tricks will help you use EDLIN to quickly perform simple editing tasks on ASCII files. Take a few minutes to try them out on your own system to actually see how they work.

### EDLIN Subcommands — A Summary

The purpose of this article was to give you a basic introduction to EDLIN, including its subcommands, and some of its capabilities. If you are a new computer user, I encourage you to sit down at your computer and try these examples. You will find that EDLIN is not nearly as difficult to learn as you thought.

I have not attempted to cover every subcommand variation or all of the options available in the EDLIN subcommands. It is not intended to replace your word processor since EDLIN simply does not have the features that most people want in a word processor. And like most software, it takes a little practice to become familiar with the capabilities and features of EDLIN. It is admittedly not the best editor around, but it is useful for small tasks that you want to do quickly.

Figure 3 contains a complete list of all EDLIN subcommands. I have not discussed the use of the Append or Write subcommands in this article, and my reason is that I do not believe that either subcommand should ever be needed. The Append subcommand allows you to edit files which are larger than the available memory (EDLIN uses 64K of memory), and it is normally used with the Write subcommand.

```
EDLIN [d:][\path]filename.typ[/B]
```

Even though EDLIN is fast and easy to use for small files, it gets to be very clumsy and cumbersome if you are trying to edit a large file. My personal preference is to limit the use of EDLIN to files of 100 lines or less. Even so, I still use WordStar's non-document mode to edit ASCII files when I need to perform complicated editing functions.

In order to get the most out of EDLIN, I suggest using it for the editing of small files, such as batch files (e.g., AUTOEXEC.BAT and CONFIG.SYS). You will probably want to use your standard editor or word processor for most other chores.

### A Batch File Fix

Have you ever run a batch file and found that you had an extra blank command prompt displayed? If you ran the example STARTUP.BAT file, you might see something like the following:

EDLIN [d:][\path]filename.typ[/B]	
Subcommand	Description
n	Edit line number n
„dest-nC or [begin-n],[last-n],dest-n[,count]C	Copy
[n]D or [begin-n],last-nD	Delete
E	End editing and save file
[n]I [begin-n] [,last-n]I	Insert line(s) in file [before line n]
[begin-n] [,last-n]L	List (display) lines in file
„dest-nM or [begin-n],[last-n],dest-nM	Move
[n]P or [begin-n],last-nP	Page
Q	Quit editing and do not save changes
[begin-n],[last-n][?]{RoldstringCTRL-Znewstring	Replace
[begin-n],[last-n][?]{Soldstring	Search
[dest-n]T[d:]{filename.typ	Transfer
[n]A	Append (not discussed)
[n]W	Write (not discussed)

**Figure 3**  
**EDLIN Command Summary**

```
A>DATE
...
A>TIME
...
A>
A>_
```

The fourth display line is where your cursor finally stops, but the third line is blank. Why? Even though you don't see it when you're editing, there is an "extra" set of carriage return and line feed characters in the file. If you don't like this extra blank line, there is an easy way to fix it.

After you finish making changes to a batch file with EDLIN, enter a pound sign (#) to get to the end of the file. Then enter the List (L) subcommand to see the last line. Edit the last line by typing in the line number. Press F3 (to copy the entire line), then press F6 (to enter a CTRL-Z character at the end of the last line), and press RETURN to store the line and get back to the EDLIN command prompt. Finally, save the file with the E subcommand. The CTRL-Z character (actually displayed as ^Z) is used to indicate the end of an ASCII text file, and you have simply inserted an "End-of-file" marker before the extra carriage return/line feed characters, which eliminates the blank command prompt line.

### The CTRL-Break Feature

One of the more interesting things about MS-DOS is the use of the CTRL-C or "Break" feature that allows you to abort DOS commands, EDLIN subcommands or BASIC programs that are executing. Nearly all of the MS-DOS manuals and BASIC documentation available implies that you absolutely must use the CTRL-Break key sequence to accomplish the abort function. In most cases, that is not true. You can also use a CTRL-C to abort a DOS command, exit to the BASIC Ok prompt or abort EDLIN (or DEBUG) subcommands. In these cases, most programs that accept a CTRL-Break will also accept a CTRL-C as an abort command. How do you know which of these programs will also accept a CTRL-C? Test them in a situation where the failure to abort will not cause a problem. To be sure that most commands will stop as soon as possible, you may want to add the BREAK=ON command line to your CONFIG.SYS file. You may also want to refer to Chapter 7 in the first POWERING UP book for additional information on CONFIG.SYS.

I prefer the CTRL-C key sequence simply because I think it is easier to type than the CTRL-Break. Perhaps you will find it easier too.

### Next Time

The next article deals with the Zenith ROM and the various commands that can be used with it. We will look at what any microcomputer ROM really does, some of the Zenith ROM commands, and how you can use them.

If you have any questions about anything in this column, be sure to include a self-addressed, stamped envelope (business size preferred) if you would like a personal reply to your question, suggestion or comment.

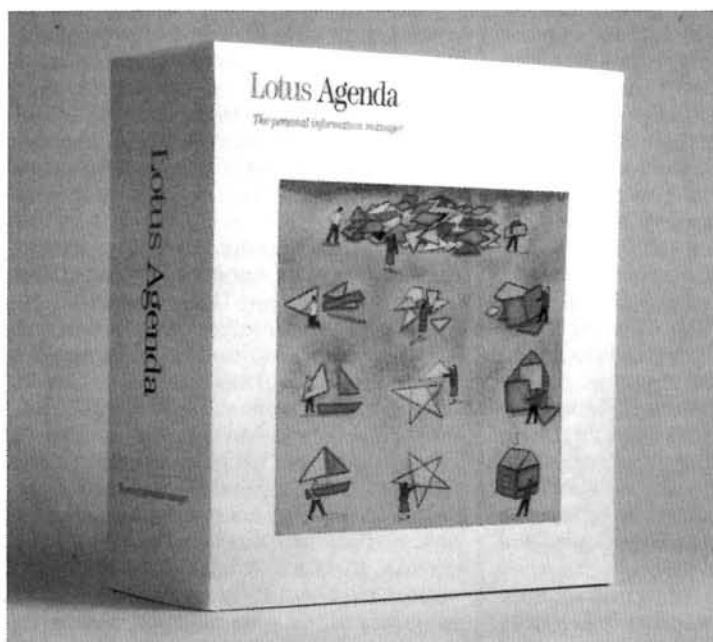
### Products Discussed

HUG Software  
Powering Up (885-4604) ..... \$12.00  
Heath/Zenith Computer Centers  
Heath/Zenith Users' Group  
P.O. Box 217  
Benton Harbor, MI 49022-0217  
(616) 982-3463

### Software

MS-DOS Version 3.3 Plus  
..(OS-51-3) ..... \$149.00 (List price)  
Mail order with update card only) .. 49.00  
Heath/Zenith Computer Centers  
Heath Company  
Hilltop Road  
St. Joseph, MI 49085  
(800) 253-0570





# Lotus Agenda

**Kevin R. Grantham**  
**Suite 606**  
**2701-C West 15th Street**  
**Plano, Texas 75075**

## Finally, a product that does for ideas and text what 1-2-3 did for numbers charts!

There is a new class of application software available that promises to change the way we work with our personal computers. The term "Personal Information Manager" (PIM) was coined by Lotus Development Corporation, whose Agenda product is the first of this new breed. So fundamental is the change that Agenda cannot be described in the context of any existing application type, neither word processor, outliner nor data base.

Recently, a number of computer industry magazines have attempted to compare and rate software products which claim to be PIMs. Most of the packages reviewed belong to other software categories, although their makers are trying to cash in on the interest in PIM by pointing out their PIM-like features. Agenda is a completely new product, and is as significant, in my opinion, as the original Visicalc or 1-2-3 in days past.

I will attempt to describe Agenda, both in terms of what it is and what it is not, and describe some applications to which Agenda has been applied.

The personal computer became popular when professionals understood the value of decision support computing power on their desktops. Because so many people work with numbers as part of their professions, it is not too surprising that numerical processing applications like Visicalc and later Lotus 1-2-3 were the catalyst in the original personal computer revolution. When word processing software matured to compete with stand-alone machines, the personal computer

took to the desktops of the clerical staffs, as well. And as storage technology advanced, the personal computer became more and more an important repository of corporate information.

Personal computers became successful because they allowed managers and professionals to analyze and evaluate information in their own way, using tools of their own choosing. A spreadsheet is a blank piece of paper just waiting for numeric doodling. And for those with a more structured task at hand, templates are available for the "fill in the blank" crowd.

Missed in the rush for new data bases, spreadsheets and word processors was the other side of a professional's life: textual information analysis. Most professionals are paid to know or to get answers for decision makers. And most decision makers are paid to filter the chaff, get down to the facts, and make a good decision for their organization. The information dealt with is not necessarily numeric. It does not fit into the ordered rows and columns of the relational data base. It needs more structure than the word processor or notepad. This information, the "personal" information with which a manager or professional does his job, is often buried within other forms of expression. The sources include the trade press, reference manuals, electronic or paper mail systems, meeting notes and conversations. It is often related to time, with due dates or coordination dates.

To address this need, the time management companies are making millions

of dollars selling personal "organizers". In addition to the usual diary, these small notebooks keep notes of meetings, action item lists, and provide sections for filing the results of telephone calls, visits, expenses and the like. Many professionals have found that these organizers require too much time to maintain, however. A phone call may encompass five different subjects, interwoven in the thread of conversation; a visit to a customer likewise. A note made about one idea may later turn out to have relevance to some completely different project.

The organizers also assume that one knows ahead of time where information is to be filed. If a category develops later on, there is no good way to find all the bits and pieces and log them in again, under the new topic. As a result, the professional either spends too much time creating indexes and cross references, or just gives up.

Personal Information Management (PIM) software is designed to fill the gap, to give the professional or manager a tool to organize this random and interwoven information, presenting it to him when he needs it in a form he can use. Since Lotus coined the term last year, a number of products have appeared claiming to be PIMs. These range from popup notepads to outliners to data base packages. Most simply try to apply an existing software package to the problem, hoping that it will be close enough to be successful. Many attempt to address the problem of capturing the information, but fall short when organizing or presenting the infor-

## Z-100/Z-248 Rebuilt Power Supplies

Exchange plus Core . . . . . \$150  
Unit shipped when Core received.  
COD Only.

**Traceable Standard Labs**  
1686 Jennifer Drive  
Macon, GA 31206  
(912) 788-6077

Reader Service #199

## Quality Heath/Zenith Enhancements

- H150 Speed Mod \$34.95, 8 MHz DMA and Controller \$34, MT PAL 704k RAM \$19.95, Super PAL 1.2 meg RAM \$29
- Smartwatch clock module for all H/Z Computers \$29
- H89: WIN89 20 meg Hard Disk \$399, Speed Mod \$34.95, NZCPR \$59, ZSDOS (includes Date Stamp) \$65
- Complete Line of EVEREX Products and Computers. 20 Meg Hard Disk \$259, 30 meg HD \$279, 40 meg \$399 H-248 3 meg RAM Card \$95, H-150 2 meg RAM Card \$85 VGA Card \$289, Int. Modem \$149, Ext. Modem \$189
- External 5 1/4 Drives for Z-181 \$181, Z-183 \$189
- H-148 1 1/2 Card Expansion Bus \$89

## Micronics Technology

(205)-244-1597 BBS: (205)-244-0192

Suite 159, 54 Dalraida Road  
Montgomery, AL 36109  
Call or Write for Our Free Catalog

Reader Service #114

mation collected.

Lotus Development Corporation also produces word processors, spreadsheets, data bases, popup notepads, etc. But Lotus took a closer look at the problem, and decided to create an entirely new product expressly designed for personal information management. Called Agenda, the package was first demonstrated at Comdex in 1987. Lotus then signed several large organizations to test and comment on the package, including several departments of the US government. It was applied to personal time management, of course, but it was applied in novel and fascinating ways, as well. One tester, a bank, used it to analyze news from newswire sources such as Associated Press and Reuters. This application filtered and categorized news relating to third world countries who had loans with the bank. Some US government departments used Agenda for "low-level intelligence" processing.

Agenda is an entirely new concept in information management. I will try to describe Agenda, then explain how it is different from the other software types.

Agenda maintains a text data base. This is unlike document management sys-

tems, or like text data bases such as askSam and Q&A, because Agenda is at the same time more structured and more analytical. It is unlike text search programs such as Gofer, because it does not attempt to look at documents all over the disk. Keep in mind that Agenda is unique, that it is difficult to describe, and that trying to compare it to other products on the market is inaccurate at best, at least until real competition shows up.

Within Agenda there are four primary concepts. An Item is text up to 350 characters long. It might be something like "Spot market sugar prices are expected to fall 5 points by March."

A Category is a heading. Categories are arranged hierarchically, in outline fashion. Agenda, however, is not an outliner package, and many Agenda users also use other commercial outliners such as PC-Outline and Grandview. They have little in common with Agenda.

Each item is assigned to one or more categories. It may be assigned to the category under which it was entered, but it may also be assigned automatically or manually to other categories as well. Automatic assignments can occur based on the text of the item. In the above example, the item might have been entered under the category "Forecasts." It might also automatically be assigned to the categories "Spot Markets," "Sugar", and assigned a "When" date of 1 March.

A category acquires items via other techniques as well, although text matching is the one most new users will be most comfortable using. A category definition can include "conditions" which declare other characteristics an item must meet before it can be assigned to the category. Conditions may include a test to see if the item is (or is not) assigned to a combination of other categories; whether the When, Entry or Done dates fall within certain ranges, or whether other text is contained in the item.

A category may also have an action assigned, which will be performed on any item which becomes assigned.

A Note is text, up to 10,000 characters long. A note can be attached to an item or to a category, or not attached at all. A note can be internal to the data base, or can be an external ASCII file.



Figure 1

A View is an arrangement of categories and items, cross referenced in whatever form makes sense to the user. Agenda supports as many views as one wishes to define. The view consists of a category as a section header, and all the items assigned to the category shown under it (see Figure 1). There may be more than one section in a view, arranged vertically.

To the left and right of the section may be cross reference columns, called Category Columns. These may be formatted as "Yes/No" and "X", showing that the particular item is or is not assigned. They also may be formatted as "Actual", which will show the child category if assigned. For example, a parent category "People" might have child categories of "Bob", "Sue", and "Tom". If an item appeared in a view, and a category column was showing "People" in that section, then the category "Bob" would appear next to the items that had been assigned to Bob (see Figure 2).



Figure 2

An item may be entered in any category. Agenda will examine the text of the item, and assign it to other appropriate categories as needed. In addition, categories can be assigned manually.

Let's take an example: Suppose you entered the item "Call Perkins next Wednesday at the Pentagon about the new hard-rubber B-52 tires we ordered." Agenda could assign the item to Calls, Pentagon, B-52, Tires and Orders automatically. If you had a cross-reference column called Persons in the view, you would see that Perkins name did not show up, since you had never defined a category called Perkins. Just move over to the blank spot in the column, and type in Perkins. Agenda will automatically create a category called Perkins, under the Persons branch, and assign the item to it. The next time you enter an item with the name Perkins, Agenda will assign it automatically. Agenda will also give this item a When date of next Wednesday (whatever that date happens to be). You might also have a family of categories called Priority (Urgent, High, Medium, Low). If "High" had a condition attached to include all items with When dates from tomorrow to next Saturday (a sliding window) then this item would also be assigned to High Priority automatically.



Agenda evaluates dates every morning, as well as when new items are entered, so next Wednesday, when you call up any view with priority in it, you will find that this item has moved from High to Urgent priority.

When someone wants to know what is going on with spare parts, you could pull up (and print, if desired) a view by Spare Parts, under which Tires could be one category. The item you entered would be there under tires, and the cross reference could show B-52 as aircraft type. Views can be created quickly with just a few keystrokes, and filtered, sorted, printed, etc.

Filtering and sorting views adds to the versatility of the product. Filters can exclude items which are not important at the moment, and sorting can place items in alphabetical or category order. For example, if your Persons category had its child categories sorted by military rank (and theoretical importance), then views that contained Persons could be sorted the same way, even though the names were not alphabetical.

One of Agenda's other features is its understanding of dates. Much of the information one deals with daily is time related. Agenda automatically keeps track of three dates for each item. The Entry Date takes on the DOS system date when the item was entered, although it can be assigned some other date. The When date is derived from the text of the item. The Done date is set when an item is marked as Done (more about that later).

When date setting in Agenda has been hinted at already. Agenda will look in the text of the item and can understand terms such as "next Friday", "last Tuesday", and "first of May". This capability, not without its limitations, makes Agenda extremely useful as a task organizer and project tracking tool.

Items can also be marked as Done. Done items can be filtered from active views, if desired, but retained in specified categories or exported to external files. For example, one might want to keep track of completed items for reference purposes, and so retain them in the data base for a while. Another user might export them to a file, so that they are clear of the data base, but could be imported again if desired. Also, the external file could be used as the basis for an activity or status report to management.

If all this sounds pretty amazing, it is. This is a tool to put the personal back in personal computers. It goes great with laptops and field work, as well as with desktop machines. However, it is not designed for "workgroups", and it is not something that you can hand your secretary and have her set up for you. It is complicated to learn but easy to run, although no more so than a spreadsheet. The difference is that it doesn't fit a good real life

metaphor, in other words, it doesn't look like an accountants pad (spreadsheet), a typewriter (word processor), etc. It also requires a big chunk of memory (it will work on a 512K machine, but only if you are not running other popup or network software. It needs 430K free and clear). It positively insists on a hard disk for maximum performance. It is not a popup, and it doesn't use Expanded Memory (interestingly enough, since Lotus was one of the companies that invented the LIM-EMS specification for expanded memory).

In order to describe this new software category, it may be valuable to describe what Agenda is not, to differentiate it from the competing claims and incomplete magazine reviews.

Agenda is not an outline processor. The best outline processor available today is considered to be Grandview. Combining a good word processor with a good outliner, and providing some limited ability to file entries under different subject headings makes Grandview the choice of professional writers. Grandview, and other outliners such as Sidekick Plus (Borland) and those built into word processors are not really competitors of Agenda, because the products do fundamentally different things.

It is true that Agenda files items of information in different categories, and that categories are structured hierarchically. The difference is that the same item can be filed in more than one branch of the hierarchy automatically. This allows Agenda to organize information as it is entered, and reorganize it at any time. Of course, outliners do not offer the cross referencing and date manipulation of Agenda, nor the automatic assignment to categories, among other things.

An outliner typically places an item of information (a series of paragraphs, say) under one heading in the outline. An outliner allows the headings (and associated items) to be moved around at will, and allows heading levels to be compressed to make reorganization easier. The outliner requires a structure to be developed, to which is added items of information. Agenda does not require a structure to be constructed, but will build the structure dynamically as information is added.

This means that an item of information can be added and automatically assigned to a project, a responsible person, a product, and an account, even though each topic is in a different branch of the outline.

Agenda is not a free form text data base. These data bases are excellent for people who must manage large volumes of textual information, and need a way to find passages and references easily. They usually employ some boolean search command to find information, and may use indexing schemes to speed up re-

trievals. Unfortunately for some, they do not provide any categorization of information. They generally are free form, so building a useful "application" with them is akin to programming in a traditional language. They are very good at what they do; they just don't do what Agenda does.

Agenda does allow one to enter free form text, in the form of Items (up to 350 characters) and Notes (up to 10,000) characters. Agenda finds key words and phrases, and dates, and assigns the item or note to the correct categories. Agenda allows Views to be created, which show items assigned to categories (a group of items assigned to a category and displayed in a view is called a Section). Agenda views also contain cross reference columns showing other categories to which items are assigned. Some of this capability can be obtained from the freeform text data base by writing complex macros and report definitions. Agenda does that automatically.

Agenda is not a traditional relational data base. Agenda is not organized in rows and columns, with key fields. Agenda has two different data structures, with links tying them together. There is the information structure, which is a free form data base containing Items and Notes. Then there is a hierarchically structured Category arrangement. Items and Notes are assigned to appropriate Categories based on their content and on other criteria. The physical layout of the data base can be difficult to visualize since it is not the outline structure entirely nor is it a relational flat structure. On the other hand, the view mechanism means that the data base structure does not need to be considered, as it does with relational data bases.

Agenda is not a spreadsheet, although it has been called a "Spreadsheet for Ideas," a sentiment I agree with completely. Agenda does not understand numbers. It understands information and dates, and unlike a spreadsheet does not require the "template" to be built before it can be used. Of course, if you have templates, and there are companies now specializing in developing Agenda templates, so much the better.

Agenda is not a word processor. Definitely not. One of Agenda's weak points is its built-in editor. However, Agenda can organize text in ways you never thought possible, and make it available to a word processor or external outliner.

Agenda is not a Project Management system, although Agenda understands and can process dates. If you entered an item such as "Call Frank on Friday about the new maintenance program," Agenda would assign the item to "Frank", "maintenance", "Calls" and give it a When date of this coming Friday. Agenda understands phrases such as "First Monday in January", "Next Tuesday", "every Mon-

day", etc. Agenda keeps track of the Entry Date for each item (which can be used to offset from a prerequisite item), the When, or due, date, and the Done date. It does not do all the load leveling, Gantt and Pert analysis, critical path planning and the like. However, I have found it a valuable adjunct to a project management system such as Timeline. Agenda is not as complicated to set up and maintain as a full scale project management system, yet provides an analytic tool to monitor project status that structured project management systems cannot.

So what is Agenda? It is an idea or information data base with automatic categorization. It allows independent Views

of the data base to be constructed without regard to the storage of information. It is a personal information organizer, an information analyzer, a to-do list manager, a task planner, a filing system, and more. It is a personal tool that thinks and acts the way most of us work, with random bits of important information. It files and saves the things we need to know, and makes no demands on us as to how we need to pull them back out.

It is also version 1 of a fascinating new wave of personal computing, and needs some additional features. Its users have found fascinating applications for it, from organizing their writing, to evaluating patient symptoms and treatments, from

marketing to legal case planning, from scheduling and project control to intelligence analysis, from Real Estate sales to public relations campaigns.

Lotus, realizing that there is no way that a two page advertisement can explain what Agenda is all about, has decided to offer either a demonstration disk (in 3.5 or 5.25 inch formats) or a videotape, for 15 dollars. Lotus also supports the package on the CompuServe Information Service (GO LOTUSB), where the developers are available to answer questions. For the price of the package, though (around \$250 street price) it's worth a look. \*



## IBM XT in an H/Z-100

Scottie Board W/ZPC \$149.

### OPTIONS AVAILABLE

- PC Compatible Serial Port W/Cable \$50
- 2nd Port W/Cable \$45.
- Clock/Calendar \$45.
- No Solder H/Z-100 MOD Kit \$5.

Requires H/Z-100 MS-DOS 768 of RAM, H/Z-100 Modification.

### FACTORY NEW...THE REAL THING

Z-217-1 W/DOCS, HDWE, CABLES, INSTRS

- With 20 Meg ST-225 Drive \$599.
- With 40 Meg MS3650 Drive \$729.

### \*\*NEW\*\*

Pair of AT Drives - External, Self Contained, Complete  
Uses 1.2 Meg AT Floppy Drives • W/Power, Cables,  
34 pin to 50 pin adapter, Instructions - \$299.  
34 pin to 50 pin adapter alone - \$23.95

IBM XT is a registered trademark of IBM Corp. • ZPC is a product of HEATH USER GROUP.

### PC/XT/AT and WORKALIKES

LOGITECH	Scanman 4" x 10" Scanner -	\$199.
	C7 Serial Mouse -	\$ 79.
	HI RES BUS Mouse -	\$ 89.
	Complete PC Hand Scanner -	\$169.
	GRAVIS Joystick -	\$ 39.

### HARD DRIVES - All Makes/Models at BEST PRICES

ADD ON CARDS, VIDEO, Controllers, Adapters

Tape Backup 40/60 MEG, 2 MEGS/minute.

Colorado Memory Systems DJ-10 \$299.

VCR Tape Backup "IMAGER" \$199.

### BEST PRICES ON-

TURBO XT, 80286, 80386 Computers -

BUYING SERVICE - any item you wish us to purchase COST + 10%  
GIVE US AN OPPORTUNITY TO BID ON YOUR NEEDS

### Scottie Systems, Inc.

1609 S. Main St.  
Milpitas, CA 95035  
(408) 262-5021

VISA & MC ACCEPTED, ALL PAYMENTS SUBJECT TO APPROVAL

Reader Service #121

# Z-100 Graphics Software

DOODLER-V Graphics Package **\$99.00**  
with D-MOUSE Mouse Driver **\$109.00**  
with Logitech C7 Mouse & Driver **\$189.00**

Font Library Disk for DOODLER-V **\$29.95**  
*44 Additional Ready-to-use fonts*

Texture & Symbol Library Disk **\$29.95**  
*Hundreds of pictures, symbols, textures*

ScreenPro Utility **\$59.00**  
*The ultimate screen print/capture utility*

**FREE** Catalog available on request

*From the Leader in Z-100 Graphics...*



3620 Amazon Drive  
New Port Richey, FL 34655



Order toll-free...

**800-346-2152**

Or in Florida... 813-376-9347  
Unconditional money-back guarantee  
if your not satisfied!



Reader Service #107

## Goodbye Sextant, Computer Service, Microsoft Word 5.0, Word Companion Disk, HUG Laptop Utilities

Some articles are more difficult to write than others, and this is one of those. I always dislike having to say goodbye to a friend, especially a permanent goodbye, but this time there is no choice.

### Goodbye Sextant

Charlie Floto, publisher of *Sextant* magazine and the *Buss* newsletter, has just informed me that: "Sextant and Buss have ceased publication." Charlie also says that: "The most recent issues of each came out in March." For those of you who subscribed to *Sextant*, my file indicates that the last magazine was issue number 40 with a cover date of March 1989 (Late Winter). The last *Buss* newsletter I received is #159 with a date of March 7, 1989 in the lower left-hand corner of the cover page.

The next obvious question is: "What will happen to existing subscriptions to both publications?" When I talked to Charlie, he told me that there had been no decision, and I also just received a note from him stating that this question was "not yet resolved." His note also stated that: "The one thing clear is that there can be no cash refunds."

At this point, Charlie also tells me that he is talking to several publishers who might be interested in picking up these subscriptions, but there is no definite decision as of the time of this writing. Perhaps there will be a final resolution of the question by the time you read this.

Most of you probably know that I have written a number of articles for *Sextant*. And as you might guess, I have a personal interest in the subscription question since I have several years remaining for both my *Sextant* and *Buss* subscriptions. When I get more information from Charlie about the situation, I will let you know.

I found *Sextant* to be an interesting magazine, and I thought that it was a useful publication for users of Zenith and Heath computers. I will miss a different

viewpoint, and although I did not always agree with Charlie on some of the editorial statements, I still respected his opinion. That's as it should be, and the passing of *Sextant* into history is something I regret. For my part, I wish Charlie Floto and the members of his staff well in the future.

I thought one of the comments that Charlie included in his note to me was particularly appropriate in that he said: "We're all aware of the changing conditions since the 'good old days'." That is an important point because it illustrates how much Heath, Zenith, Veritechnology (i.e., the Heath/Zenith Computer Centers), and the computer market have changed over the last seven years.

### The Good Old Days

Whether or not the "good old days" were especially good is one of those issues that is especially open to debate. Some major companies that are now quite well-known did not even exist 10 years ago — Lotus, for example. Other major companies, like Microsoft, were barely a gleam in a prospective founder's eye. And of the few major companies that did exist, Micropro International has been one of the basic software suppliers for microcomputers with their WordStar program that was originally released in 1979.

Many of you are probably not aware of the fact that Heath Company has been designing and selling computer kits for years. Heath was doing this long before anyone ever thought of Zenith Data Systems, and there are many people today that believe that the 8-bit H-8 computer kit — developed and originally sold by Heath in the fall of 1977 — was and still is one of the best systems ever made. I've seen one of these computers, and it is dif-

ficult to disagree with that assessment. Heath also introduced the 16-bit H-11 computer kit in that same catalog, and I mention that in case you thought that IBM started that idea. By the way, you could also get an operating system called HDOS — Heath Disk Operating System — as well as Benton Harbor BASIC. In later years, the CP/M operating system also became available.

Then Heath developed a computer "terminal" for the H-8 called the H-9 that included a keyboard, as well as a CRT. The H-9 terminal had some shortcomings, and Heath had a working prototype of a new terminal, called the H-19, working about a year later in September 1978. This was an extremely popular unit, and some clever Heath engineers modified the terminal "slightly" to make it a full-fledged microcomputer kit known as the H-89. As I recall, both the H-19 terminal and the H-89 computer were officially "introduced" in mid 1979. After Heath was purchased by Zenith (also in 1979), you could buy an assembled version called the Z-89 or Z-90. The H-89 used the now-famous Zilog Z-80 MPU (Micro Processor Unit) which ran at a modest clock speed of 2 MHz.

All in all, 1982 was a busy year for a lot of people, including me. In case you are wondering just how long I have been involved with Heath and Zenith computers, I happen to have the very first issue of *Sextant* that was published in Spring 1982. I bought an H-89 computer kit that year and joined HUG shortly thereafter. The Z-100 computer was announced and released, and IBM also released the PC that year. Both computers ran at a screaming clock speed of 4.77 MHz, but the Z-100 was especially innovative because it included two MPUs — an 8-bit 8085 (generally compatible with the Z-80) and a 16-bit 8088. The IBM PC only had a single 8088 MPU. More than a few HUG members were dismayed to learn that the Z-100 was not compatible with the IBM

**William M. Adney  
P.O. Box 531655**

**Grand Prairie, TX 75053-1655**

**Copyright © 1989 by William M. Adney. All rights reserved.**



PC, and I eventually wrote a couple of articles about it.

In April 1983, I bought a Z-100 Heathkit and began finalizing manuscripts for my first REMark articles and books, which later became the FlipFast series. My first "On the Leading Edge" article was published in November 1983, and it was originally written on my H-89 with the old Magic Wand word processor that some of you may know now as PeachText.

From that point on, Zenith and Heath quickly learned that compatibility was the name of the game, and the Z-100 PC series was introduced, followed by a number of other 8088-based systems. The Z-200 series with an 80286 was released as an alternative to the IBM AT, and the Z-386 was released when the 80386 MPU became available in production quantities. Today, the latest word is that a lot of people are waiting to see what the 80486 MPU will be able to do.

During this time, and especially in the last few years, there has been a significant change in the overall computer market. Large businesses still buy powerful mainframe computers, but the general trend is to buy smaller computers, particularly microcomputers, to give many people access to various facilities and software. The microcomputer, which was the totally-owned realm of the hobbyist in the 1970s, has now become an accepted business tool, like an electric drill or saw in the construction trade. Some people even say that the microcomputer has become a commonly available "commodity" because of the number of generally compatible systems available from different manufacturers. As I mentioned last month, Zenith, in particular, has not done a very good job in keeping up with this trend because of the apparent, and successful, concentration in the large government market. Things are not at all like they were in the "good old days."

All of us at HUG have watched this trend with interest, and we have recognized more and more HUG members are not using computers as a hobby, but as a business tool. For many of us, the use of computers represents something we use in a vocation, as well as an avocation. For others, especially new HUG members, a computer is nothing more than a tool to get a job done, such as word processing or using a spreadsheet for budgeting. The heterogeneous composition of the HUG membership — a mix between technically knowledgeable hobbyists and new computer users — makes it difficult to strike a balance between technical information and topics that new users need to know about. HUG has recognized that new users need additional information that is not available in the manuals supplied with the Zenith and Heath computer systems; therefore, we have articles for the beginner, such as my "Powering

Up" column and book, as well as articles for the more advanced user and hobbyist relating to special software and programming.

I was talking to Margaret Bacon (HUG secretary) the other day, and she has observed (like I have) that many new HUG members have a preoccupation with computer model numbers for some reason. I guess that many new members expect to see a specific article about the Z-159, Z-148, eaZy PC or whatever, but I am still puzzled by the fact that some new users spent a wad of money on a computer and apparently don't understand that these are PC compatible systems. While it's true that each model has some unique features, which is, of course, the reason for different model numbers, most of these features are somewhat "hidden" to new users, and in many cases, not as important as they would be to an "advanced" user. Still, the good old days are gone, and I think one of the major changes has been that computer users are becoming less interested in the technical details of what goes on inside a computer. That is quite evident from your response to the Powering Up book, and it is the main reason I have decided to continue the series beginning this month with a more advanced set of articles.

#### Microcomputer Hazards

One of the particular hazards of using a computer is that there is a considerable amount you have to learn and know. In a mainframe data processing operation, there are all kinds of job skills required to keep the system up and running. That includes: application programming, system programming, computer operator, data entry clerk, tape librarian, repair technicians, installation personnel, and all the tasks that these people perform. This totally ignores the vast number of people who "use" the computer, typically with a 3270 terminal. The introduction of the microcomputer has changed all that. When the microcomputer was introduced, a hobbyist did not seem to worry about the fact that there was a large volume of information that had to be learned just to run the system. The important point is that each user of a microcomputer must have a wide variety of skills and knowledge in order to cope with the kinds of problems that can occur. If you have a micro, you must be familiar to some extent with the operating system (MS-DOS), as well as the various applications you need. If you do not have someone knowledgeable in your local area, that can be a particular problem because you may not know what to do. The unfortunate trend seems to be to always blame the manufacturer of a product, whether it's a computer or an automobile. And I think that's where many problems occur.

Time is at a premium for most people, and many people simply don't have the time or the interest in learning about the details of how a computer works. How it works is just not important so long as it does work. In itself, that is not a bad approach because I have long maintained that a computer is nothing more than a tool — a complicated one, but nevertheless a tool — to accomplish a task. More importantly, today's overall market perception is that the microcomputer is a business tool, not just, or only, a plaything for a hobbyist. For the Heath market, in particular, this is a significant change because Heath computers began primarily as a kit for hobbyists, and now Zenith (and Heath) is trying to cope with the transition between the hobby world and the business world. In the hobby-oriented world, a computer kit was an end in itself. In the business world, a computer is just a tool that is used to obtain results without any particular concern as to the details used to reach that end. Most businesses generally have a microcomputer "specialist" available to solve common problems.

I have been seeing (in your letters) and hearing that some people perceive that the Heath/Zenith Computer Centers (HZCC) are not providing adequate service. More and more people are complaining about service, especially on Zenith systems, as I mentioned last month. Although there are some really obvious cases where Zenith and/or the HZCCs may have fallen short in the service area, there are some other considerations. As far as I can tell, there are several reasons for this, and some of them are quite interesting.

#### Service Issues

From an informal survey, there seem to be three major types of service issues. The first is primarily related to getting help setting up or using a computer after it is purchased, usually from someone on the sales staff. The second and third issues involve some kind of "repair" to the computer by a store's service department. The discussion of these issues is arranged by the frequency that they seem to occur, from most frequent to least frequent.

If you look at very many computer publications, you will find that there are many ads that list all kinds of Zenith computer hardware. Most of the ads I have seen focus on the very popular and highly rated SupersPort laptops, and a lot of ads also feature the excellent Zenith FTM monitor (ZCM-1490). Based on price, many people buy these units from a discount mail-order business, and when they have problems with the unit, they expect immediate and top priority from a LOCAL dealer, such as an HZCC. That scenario is nearly as ridiculous as buying a Chevrolet and expecting a Cadillac dealer to provide all of the "standard" service benefits, in-

cluding a free loan car. In one of the local Heath stores, I saw one customer get REAL upset because he could not get any detailed help on a SupersPort that he bought at a local discount computer store. Apparently, this customer wanted a lengthy tutorial session on how to load a hard disk while I was waiting to get some help on a printer problem that I did BUY at that store. As a paying customer, I think that most people would understand that I got top priority in that situation. Unfortunately, that customer probably went away complaining to all his friends about the terrible service in the HZCC. There is no easy cure for this problem, but perhaps Zenith should consider a requirement that ALL dealers (including discount stores) selling computer equipment must also provide appropriate support for that hardware. By the way, I should mention that any authorized service center should provide warranty service on a system, but it is obviously unrealistic to expect to get top priority if you did not buy the system from that store. Lest you have any doubts, dealers who are authorized to sell Compaq computers must provide support, and that's why you rarely see heavy discounts on those systems.

The second issue is not quite as clear cut. This occurs when a user takes a computer for service, and after it has been checked, the user takes the computer home only to find that the problem still occurs. I have received an amazing number of letters talking about the "terrible" service department in an HZCC, but believe it or not, I have "fixed" more than a couple of these computers as a result of these letters with no hardware changes. To understand what happened, it is important to know that the primary function of a service department is to diagnose hardware problems and fix them. Service technicians are expected to be hardware experts as part of their job, and a wide experience in software is not part of that job. If software is causing a problem, such as a system freeze, that cannot be diagnosed as a hardware problem, and no service technician will be able to "fix" it by replacing hardware. It should not surprise you to learn that a lot of service departments use the Zenith Disk Diagnostics to check out computers, and technicians also rely on experience and technical knowledge to verify that the hardware is functioning correctly. For computers that contain non-Zenith hardware (e.g., memory boards), the checkout is tricky because the Zenith Disk Diagnostics may not be able to test it. But if the problem is software related, such as conflicting device drivers or other memory-resident programs, the hardware may be working fine, and a technician may never find the problem. To show how elusive this kind of troubleshooting can be, let's look at an example of what can happen, even

though it does not involve the service department.

One Huggie wrote to me that he had installed a ROM upgrade (from version 2.9 to 3.1C) in his heavily modified '151, and "all sorts of things went wrong." If he had taken the computer to a store, it is more than likely that a service technician would have reported that there were no hardware problems (or firmware either) — the system hardware was perfectly functional. It would also be easy to blame Zenith for a bad ROM, but that wasn't the problem either. Fortunately, he had the good sense to install the old ROM, and although his system worked fine again, he did not know what the problem was until he read my May article. As it turns out, he was also using an old version of Borland's SuperKey program that was simply not compatible with the new ROM (with the 101-key keyboard support added) as I mentioned in that column. After reading it, he ordered a new version of SuperKey (and the Northgate keyboard), and in his letter he told me: "I am now in seventh heaven." In case you are wondering exactly who caused this problem, it is a fact of life that IBM did when they introduced the 101-key keyboard and ROM support for it. That new keyboard has caused a lot of heartburn for a number of manufacturers, including Zenith, Compaq, IBM, and Borland.

The most important point about this second issue is that you need to take the time to be sure that apparent hardware problems are really caused by an actual hardware failure. Admittedly, this is sometimes difficult, especially for new computer users. I have included tips for troubleshooting problems in Chapter 12 (How to Select Utility Programs) and Chapter 15 (Maintaining Your Computer System) in the *POWERING UP* book available from HUG. You also may be able to save a lot of time (and possibly money) if you run the Zenith Disk Diagnostics yourself or run the Zenith ROM-based TEST command. In short, make sure that you really do have a hardware problem before you take your computer into the store.

### How Do YOU Spell Relief?

For computer problems, I spell relief: P-A-T-S-W-A-Y-N-E! Our own intrepid HUG Software Engineer Pat Swayne has done another outstanding programming job. For those of you unfamiliar with Pat's work, the difficult can be done immediately (as in this case), and the impossible takes a little longer, such as Pat's legendary ZPC program. If you never heard of ZPC, Pat performed the nearly impossible task of getting the non-PC compatible Z-100 computer to run PC compatible software using only a software emulation program called ZPC. As a matter of fact, it actually is impossible to develop a program to run PC software on a Z-100 (ask

any PC hardware expert), but fortunately for us, someone forgot to tell Pat. Pat also clearly did not know that it was impossible to run PC software with an emulation program using the standard Zenith Z-100 MS-DOS that was also not PC compatible.

In this case, I have mentioned the problem of the disappearing cursor on my SupersPort 286, and Pat reads my articles before you do. There is nothing like a challenge to generate a fix for a problem, and Pat has fixed the disappearing cursor, along with a few other problems that I have noticed, but did not mention. Pat has developed a set of programs, called the HUG Laptop Utilities, and this disk contains a number of really neat programs that solve all kinds of laptop problems, not to mention that a few of them are useful for desktop computers, too. I'll only mention a few of my special favorites, but you can see the complete list of included programs on page 4 of the July 1989 REMark.

Fixing the disappearing cursor problem is easy when you have the CURSOR program. On my SupersPort 286, the "CURSOR B" command increases the size of the cursor to about half of a standard character — MUCH easier to see. This program even works with Samna Word IV, and if it works with that program, it should work with anything. I even tried the program on an old Toshiba 1200, which has got to have the worst display available in a laptop, and it works fine on that too. You still can't read the screen very easily, but at least you can find the cursor. This program is, of course, memory resident.

As the disk documentation says, some graphics programs may look like photographic negatives on a laptop. The GEM Desktop program and other GEM applications (e.g., GEM Draw) are particular offenders here, and there is not much you can do to fix it, unless you have the REVSCRN program. When I am going to use GEM, I use the REVSCRN command to load the memory-resident program. Then I start the GEM Desktop as usual, and it really does look like a negative which is easily fixed with CTRL-SHIFT-R to reverse it.

Although the SupersPort's keyboard is generally pretty good for a laptop, it has the usual problem that the CTRL and CAPS LOCK keys are in the "wrong" location. CAPCON fixes that, but I found that it does not work when I have Borland's SuperKey version 1.16 loaded. That really wasn't too much of a surprise because of the way SuperKey has to take over the keyboard. If you use SuperKey, you can also swap the CTRL and CAPS LOCKS keys with that program.

The DTEST program is a non-destructive test that can be used to check out a floppy or hard drive on any computer. It works fine with MS-DOS 3.3 Plus regardless of partition size (i.e., no 32 MB limit



problems) on both the SupersPort and desktop computers.

The SEE program is the best way to quickly look at a file — much easier to use than the TYPE command, plus you can scroll backwards and forwards, search for a string of text, and print selected text. A nice utility.

These are just a few of the programs included in the HUG Laptop Utilities disk. As usual, the assembler source code is included on the disk in case you want to modify a program or learn assembler. A highly recommended set of utilities primarily for laptops, but most of them can be used on a desktop computer, too.

### Microsoft Word 5.0

I just received the version 5.0 update to Word, and it is clearly one of the best word processing programs available. Perhaps you have seen the latest Microsoft ads that begin with the statement: "If you want better word processing, don't settle for Perfect." Microsoft is clearly trying to take over the lead from number-one-selling Word Perfect, and it will be interesting to see the response to this ad. In case you didn't know, Word Perfect has occupied the number-one sales position for word processors for several years with Word usually showing up as number two. I think Word could have been in the top spot today if Microsoft had not been so reluctant to remove copy protection from their programs. Word Perfect has never been copy protected.

Virtually all of the best word processing programs — Word Perfect, Word, and WordStar — have about the same features, and choosing among them is mostly a matter of personal preference as I have said before. All of these programs are so powerful that it is really doubtful that any user will exercise and exhaust all of their capabilities, no matter what you do. For example, I never did exhaust all of the capabilities of WordStar 3.3, and I now have the latest version 5.5. There is no chance that I will ever need all of the power that is available in Word 5.0, but I agree that it is nice to have those features available, just in case.

I have used WordStar in various versions for over seven years, and I've used Word for about three, beginning with version 3.1. When I can choose, I normally use WordStar because I can do most tasks faster due to its keyboard commands that are particularly easy for touch typists. I am also more familiar with WordStar which probably has something to do with speed of use, too. Still, I like to choose a tool, such as a word processor, based on the job I have to do, and there are a number of times that Word is clearly the best tool for the job. Although you can use just about any of the latest word processors I have mentioned to do a job, some are easier to use for specific tasks. Unfor-

tunately, I have found that many users only know the one application program that they bought, and any criticism of that package usually becomes an emotional argument, not a rational one. But Word clearly has one feature that provides a special capability Word Perfect does not have.

All of the latest Word versions have the best support for a mouse that you can find in any word processor worthy of the name. That really is no surprise because Microsoft has been selling various types of mice for years, and you would expect good mouse support from any Microsoft program. Depending on what kind of writing you do, especially if you work with documents longer than a couple of pages, good mouse support may be essential, and Word has it.

If you typically work with documents that are only a few pages long, you probably won't find much need for a mouse or other pointing device in Word or any other word processing program. Even for documents up to 10 pages or so, I agree that a mouse may not be necessary or even desirable because you can keyboard any command without a mouse. Articles that I write for REMark are about the shortest documents that I write (10-12 pages using 78-character lines and 10 point type on my printer). Other documents range from 20-50 pages for a report to sometimes as long as 100 plus pages for a chapter manuscript for something like the MS-DOS course I wrote. For documents of that length, I have found that the capability to use a mouse with a program has gone from the "nice-to-have" category to the "must-have" category. Here's why.

For a short document, a mouse has no real advantage and may even slow you down. Consider how you might write a long document. I have found that the best way for me to write a long document is to simply sit down and write it as fast as I can think and type. My objective is to get some words recorded about the subject, without worrying too much about all of the editing, spelling, and other details. The basic document contains paragraphs and headings, but I don't worry much about print enhancements, such as bold or underline font changes, until I begin editing one of these large documents. That's when I find good mouse support is a clear advantage. Word Perfect's mouse support is essentially non-existent because it was not originally designed to support a mouse.

For some reason, I have never cared much for Word Perfect's menus. That is, as I've said, mostly a matter of personal preference, and there are obviously a LOT of people who disagree with me. I also have never cared much for the way that Word Perfect splits the screen to show print enhancement characters, but some people really like that feature, too. Lest

someone misunderstand these comments, there is nothing "wrong" with Word Perfect. I just prefer WordStar and Microsoft Word.

Based on comparing all of these programs, there is little doubt that Word is, by far, the best integrated word processor available. In this context, I am using "integrated" to mean that all of the commands seem to me to be far more consistent as a set. For example, WordStar includes a memory-resident outline program (PC Outline), but any relationship between the WordStar commands and the outlining commands is strictly an accident. On the other hand, Word's outlining commands are built-into the program in a way that I find is MUCH easier to use. If you set up your outline correctly in Word, you can quickly and easily reorganize an entire document by changing the outline. That can be an important feature if many of your documents are very large.

Word also has the spectacular online tutorial program to help you learn how to use the program. The LEARN tutorial is absolutely first-rate, and you won't find any better, even if you are a beginner. After reading the manual to be sure that Word was installed correctly, I have pretty much ignored the manuals because I spent the time going through the tutorial. You can even run the tutorial while you are in the middle of a document in case you want to review the procedure for doing something tricky. Aside from the obvious time savings, that is a particularly valuable advantage for me because I don't have (or want) to carry a lot of manuals with me when I travel. For my money, these are the kinds of things that I look for in a word processor, and that's the reason I bought the latest update to Word 5.0.

I have not even remotely attempted to provide any kind of "evaluation" or "review" of these word processors because other publications do an excellent job of reviews on common software. What I have tried to do is point out some of the special features of Word that I particularly like, especially the mouse support and outlining features that have been available for several years. If these kinds of things are important to you, perhaps you will find that this latest version of Word will help you, too.

### Word Companion Disk

There are lots of books you can buy about most popular application programs, and Word is no exception. For my money, the very best book for Word is Rinearson's *Word Processing Power With Microsoft Word*. Indeed, one could even argue that Rinearson's Word book is possibly the best available for any application program. If you use Word, this is a must-have reference, and you owe it to yourself to get a copy of this book, whether you are a beginner or an advanced user.



I bought my first copy of Word (version 3.1) several years ago when I began working on the update to the 544-page MS-DOS FlipFast book which was published in 1987. I also used Word to do the technical editing on the 408-page GW-BASIC FlipFast book published in the same year. I found Rinearson's Word book to be an invaluable reference for the common (e.g., learning), as well as some of the real tricky things, that I had to do while I was writing these books. When I received a notice that the Word Companion Disk was available for Word 5.0, I thought so highly of the book that I decided to order the disk by mailing the order form. Then I forgot about it, up to a point.

When I received my last MasterCard statement, I noticed that a charge from Rinearson Support Associates appeared with a posting date that was nearly a month old, but I still had not received the software. So I called to see if possibly my order had been "lost" during shipping. I was told that there had been some "glitches" in the programs (and apparently the manuals, too) and the software was "not shipped as scheduled," but the software "WAS shipped last Friday." I admit to being skeptical since I have played the "check's in the mail" game before, and I don't like it. I also admit that I gave the lady who answered the phone a hard time because I did not appreciate having a charge posted over a month before the software was shipped, regardless of the reason. When I order something like this, I expect that an item may be charged within a day or two of shipment, but I don't expect a vendor to "anticipate" a shipment by a month or more. Regardless of what happened, the good news is that I did receive the software less than a week after "last Friday," and it has turned out to be as good as I expected.

The Word Companion Disk includes all kinds of neat help screens (the manual says hundreds) and an incredible variety and number of macros. The software is offered with a money-back guarantee (within 30 days), and it is an excellent value for its \$49.95 price tag. Aside from the "Advanced Help Screens" (which are excellent, the software consists of well-thought-out and useful macros that can help you with all kinds of tasks. Some of them are really neat (as well as useful), such as the "calculator." Although many others are also quite useful, I think their primary advantage is in demonstrating what can be done with a macro, and it is easy to use that as a starting point to set up your own customization. Macros are a relatively sophisticated form of programming, and like programming, it is usually easier to see how to "fix" something once you have an idea what you can do.

Despite the problem I had in ordering this software, I have been assured that

"it will not happen again." I trust not. If you have a problem with a vendor of a product that I recommend, I hope you will first let them know about the problem. If that doesn't help, then you should let me know, and I will try to help. Above all, be sure that you contact a vendor FIRST to let them know you are unhappy with a product. You will find that most vendors will make a special effort to resolve the problem to your satisfaction, one way or the other.

### Powering Down

There seems to be no shortage of topics to look at, and maybe next time I will manage to make a bigger dent in my list.

For help in solving specific computer problems, be sure to include the exact model number of your system (from the back of the unit), the ROM version you are using (use CTRL-ALT-INS to find it), the DOS version you are using (including both version and BIOS numbers from the VER command), and a list of ALL hardware add-ons (including brand and model number) installed in your computer. The list of hardware add-ons should specifically include memory capacity (either added to an existing board or on any add-on board), all other internal add-on boards (e.g., modems, bus mouse or video cards), the brand and model of the CRT monitor you have, and the brand and model of the printer, with the type of interface (i.e., serial or parallel), you are using. Also be sure to include a listing of the contents of the AUTOEXEC.BAT and CONFIG.SYS files unless you have thoroughly checked them out for potential problems (e.g., TSR conflicts). If the problem involves any application software, be sure to include the name and version number of the program you are running when the problem appears.

If you have questions about anything in this column, or about Heath/Zenith systems in general, be sure to include a self-addressed, stamped envelope (business size preferred) if you would like a personal reply to your question, suggestion, comment or request.

### Products Discussed

HUG Laptop Utilities (885-6014) .. \$20.00  
Powering Up (885-4604) ..... 12.00  
Heath/Zenith Computer Centers  
Heath/Zenith Users' Group  
P.O. Box 217  
Benton Harbor, MI 49022-0217  
(616 982-3463)

### MS-DOS 3.3 Plus (OS-51-3)

List Price ..... \$149.00  
Mail Order w/Update Card ..... 49.00  
SupersPort 286 w/20 MB HD .... 4999.00  
SupersPort 286 w/40 MB HD .... 5499.00  
Heath/Zenith Computer Centers

Continued on Page 33

## A-One H/Z Enhancements!

### Clock Uses No Slot

**FBE SmartWatch:** Automatic date/time on bootup. Installs under BIOS/Monitor ROM. Ten year battery. Works with all Heath/Zenith MSDOS computers. For PC's \$35.00, Z-100 \$36.50 Module: \$27.50

### Configuration Control

**CONFIG MASTER:** Menu-select active CONFIG.SYS during bootup. Software for PC/Z-100 MSDOS. \$29.95

### H/Z-148 Expansions

**ZEX-148:** Adds one full-size and one half-size expansion card slot. \$79.95

**ZP-148:** Replacement PAL chip expands existing 640K memory to 704K. \$19.95

### H/Z-150 Stuff (Not '157, '158, '159)

**VCE-150:** Eliminate video card. Install EGA or VGA card. All plug in. \$39.95, SRAM Chip (Required) \$20.00

**RM-150:** PROM used in removing video card. With detailed instructions. \$9.95

**ZP640 PLUS:** Expand standard memory card to 640/704K with 2 banks of 256K RAM chips (not included). \$19.95

**LIM150:** Get 640K RAM plus 512K of simulated Lotus/Intel/Microsoft EMS v3.2 expanded memory. Installs on standard memory card. No soldering. Must have 45 256K RAM chips (not included). \$39.95

**MegaRAM-150:** Get 640/704K plus 512K RAM disk on standard memory card. No soldering. Without RAM chips. \$39.95

**COM3:** Change existing COM2 address. Put internal MODEM at COM2. Don't lose serial port. Includes software. \$29.95

### Maximize Your Z-100

**ZMF100A:** Expand "old" motherboard (p/n 181-4917 or less) using 256K RAM chips (not included). No soldering. \$65.00

**ZRAM-205:** Put 256K RAM chips on your Z-205 board. Get 256K plus 768K RAM disk. Contact us for data sheet before ordering. Without RAM chips. \$49.00

### Z-171 Memory Expansion

**MegaRAM-171:** Put 256K RAM chips (not included) on existing memory card. Get 640K plus 384K RAM disk. \$59.95

### H/Z-89 Corner

**H89PIP:** Parallel printer 2 port interface card. With software. \$50.00 Cable \$24.00

**SLOT4:** Add fourth expansion slot to right-side accessory bus. \$39.95

Order by mail, phone, or see your Heath/Zenith dealer. UPS/APO/FPO shipping included. VISA/MasterCard. WA residents add 8.1% tax. Hours: M-F 9-5 PST. We return all calls left on our answering machine!

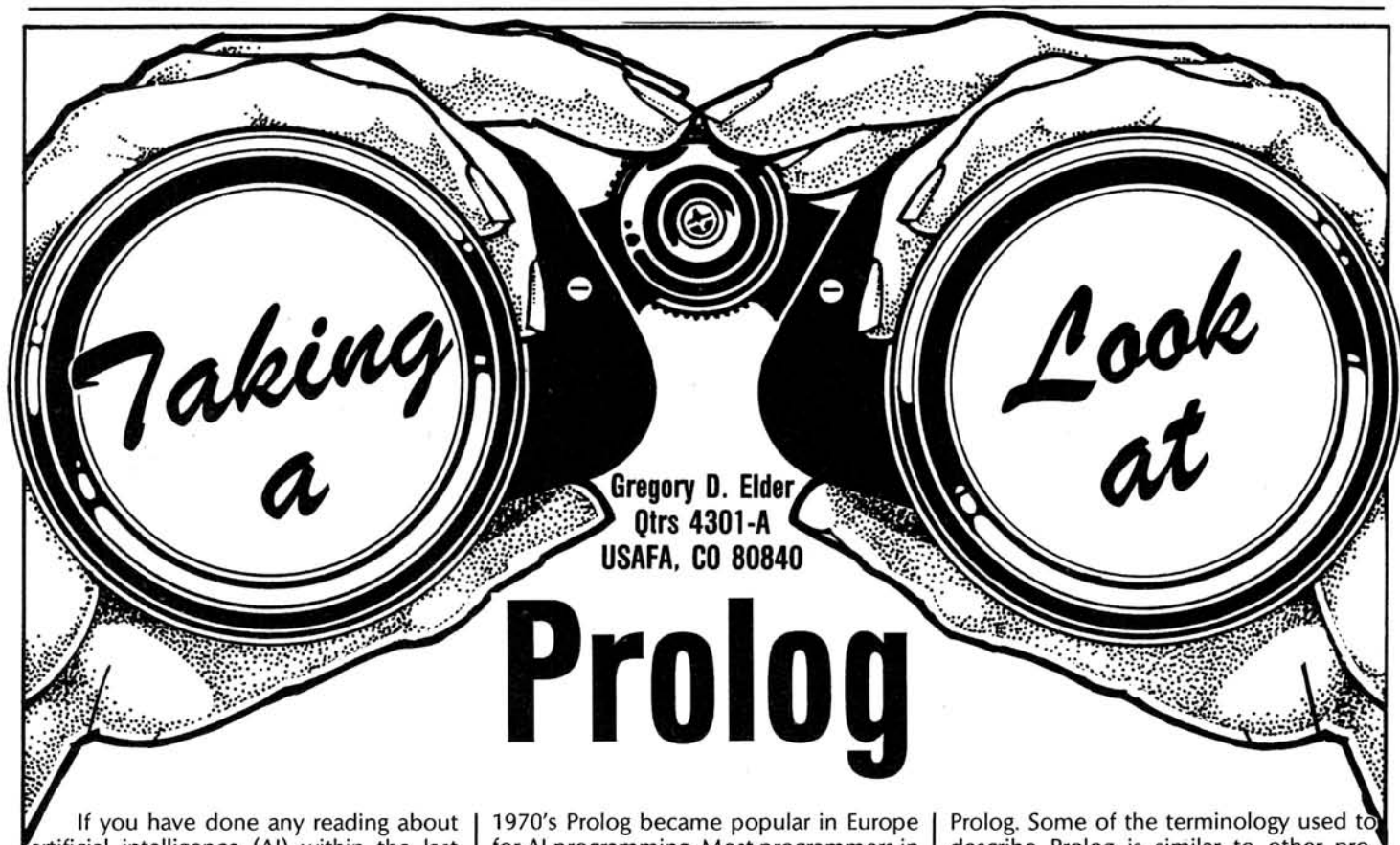
**FBE**

**FBE Research Company, Inc.**

P.O. Box 68234, Seattle, WA 98168

**206-246-9815** Voice/FAX TouchTone Selectable

Reader Service #104



If you have done any reading about artificial intelligence (AI) within the last three years or so, you have probably come across Prolog. Prolog is a programming language used primarily for AI applications, such as expert systems. This article will introduce you to the Prolog language — its major features, how to use it, and what types of applications are good for Prolog. The intent of this article is not to make you an expert Prolog programmer, but to provide you with enough information to start using the language.

If you want to try out Prolog, I recommend obtaining A.D.A.'s public domain Prolog interpreter. While not as fast as some of the commercial Prolog compilers on the market, it does have most of the features described in the book *Programming in Prolog* by Clocksin and Mellish. This book is considered by some to define the standard for the Prolog language. A.D.A. Prolog will allow you to experiment with the language at a reasonable price (free). Then, if you decide you would like to do some serious Prolog programming, you can throw down some bucks on a faster compiler with more features (Turbo Prolog and Arity Prolog to name just a couple). A.D.A. Prolog runs under MS-DOS version 2.0 or higher, so you can use it on a Z-100 or IBM-PC compatible machine. All of the examples shown in this article run under A.D.A. Prolog.

Prolog stands for **programming in logic**. It was developed in 1973 by A. Colmerauer and P. Russel at the University of Marseilles in France. During the

1970's Prolog became popular in Europe for AI programming. Most programmers in the United States, though, stuck with LISP as the language of choice for AI work. However, in 1981 Japan announced they would use Prolog for their Fifth Generation Computer project. As a result, more people began to take an interest in this logic programming language. Initial implementations of Prolog were slow and used large amounts of memory. In recent years, more efficient Prolog systems have been developed and a number of them do run on microcomputers.

Prolog is probably different from any programming language you have already used. Languages like BASIC, C, and Pascal are traditional procedural languages. This means a procedure, or algorithm, must first be developed in order to solve a problem. A program can be written in a procedural language which implements the desired algorithm. Prolog, however, does not use procedures. A Prolog program consists of facts and rules. The program makes inferences about these facts and rules. Another way of looking at a Prolog program is as a data base of facts and rules. The user can then define a goal or problem and let Prolog find the solution.

#### Syntax

The first thing you should understand about a programming language is its syntax or grammar. Syntax describes the different components of a language. Just as the English language has its own syntax (nouns, verbs, adjectives, etc.), so does

Prolog. Some of the terminology used to describe Prolog is similar to other programming languages, while some is unique to Prolog.

A Prolog program is composed of terms. A term can be either a constant, a variable, or a structure. Constants are used to represent specific or known objects. A constant may be either an integer, a combination of letters and digits, or special characters. For example, 10, -15, and 3478 are integer constants. Constants composed of letters and digits must begin with a lowercase letter. The underscore character may also be used for readability. Examples of these kinds of constants follow:

`owns, zenith_computer, year1989, fred`

You can have constants which begin with an uppercase letter, or have special characters like spaces, as long as the constant is enclosed between single quotation marks. Here are valid examples of these kind of constants.

`'John', 'Z100', 'Hello World!', 'Heath/Zenith Users Group'`

Finally, special character constants are used for certain operations in Prolog. Examples of these are '=' for equals, '\=' for not equals, and ':' to define a rule.

The second type of term, variable, is used to stand for an object which is not currently named. A variable must start with an uppercase letter or underscore character, followed by any combination of letters, digits, and underscore characters. Examples of valid variables are:

`Computer_type, Name, Address1,  
_4th_qtr_pay`



The last type of Prolog term is the structure. Structures are used to group objects or terms together. Structures are useful because they allow you to treat a group of objects as a single entity. A structure consists of a functor and its components. To write a structure, you first list the functor (a name consisting of a lowercase letter followed by 0 or more letters or digits) with the components listed between parentheses and separated by commas. The following example is a structure used to define the parts of a computer.

```
computer_parts(cpu, keyboard, memory,
               screen)
```

In the example, `computer_parts` is the functor and the components consist of everything inside the parentheses. By the way, the components may be constants, variables, or other structures. Here is another example of a structure.

```
computer_parts(cpu(Processor, Speed),
               keyboard, memory, screen)
```

In this example, the first component of `computer_parts` is another structure called the `cpu`. The two components of the `cpu` are two Prolog variables used to describe the CPU (Processor and Speed). The variables can be instantiated (given values) within a program to describe a specific processor (8088, 80286, etc.).

## Facts

As mentioned earlier, Prolog programs consist of facts and rules. Facts in Prolog describe relationships between objects. For example, by typing the two statements below into Prolog we add two new facts to the Prolog data base.

```
computer(z100).
computer(z248).
```

These two facts state that a Z-100 is a computer and a Z-248 is a computer. In this example, 'z100' and 'z248' are objects. The relationship is 'computer.' In Prolog, the names of relationships and objects must begin with a lowercase letter. (Notice that a fact is simply a structure.) As shown in the example, the relationship is written first, followed by the object in parentheses. (In Prolog, the name of the relationship is known as a predicate). Also, the fact must end with a period. If the fact contains more than one object, the objects must be separated by commas. Here are two additional facts we can define for our example:

```
owns(greg, z100).
owns(john, z248).
```

These two additional facts state that Greg owns a Z-100 and John owns a Z-248. (Remember, relationships and objects must start with lowercase letters).

With these facts defined, you can query Prolog for information about them. A Prolog question looks like a fact, except it begins with the special symbol '?-'. For example, the following Prolog question asks if a Z-100 is a computer.

```
?- computer(z100).
```

Since that fact is in our Prolog data base, Prolog would respond 'yes'. If, on the other hand, we had asked

```
?- computer(corvette).
```

Prolog would respond with 'no'.

When you ask Prolog a question, it simply searches its data base in an attempt to match the question's predicate with a predicate stored in the data base. If a match is found, then Prolog will attempt to match the objects. As shown in the examples, Prolog responds with 'yes' or 'no' depending on the outcome of the search. You may think of these 'yes' and 'no' responses as Prolog's way of telling you whether questioned facts are true or false.

We can also use variables to obtain information about the facts we have defined in Prolog. Remember, variable names must start with uppercase letters. Here is an example using a variable.

```
?- computer(X).
X=z100.
X=z248.
no.
```

In the above example, you would type '?- computer(X).' Prolog then searches its data base for the fact 'computer.' If found, Prolog instantiates the variable 'X' to the specific object and responds with 'X=z100.' Prolog continues searching its data base and instantiating the variable until there are no more 'computer' facts. At this point, Prolog responds with 'no' to indicate no more matches in its data base. Please note that in many Prolog systems, Prolog will ask after each instantiation if you want to continue searching the data base for another match.

## Rules

A rule in Prolog is a fact which depends on other facts. If you have ever taken a logic course in college, you probably developed rules which looked like the following:

```
Something is a fish if
    it lives in the water and
    has gills.
```

Prolog rules look very similar. As an example, here is a Prolog rule which states 'Ann likes people if they own a Z-100' (Ann is into quality computers).

```
likes(ann, X) :- owns(X, z100).
```

The special symbol ':-' means if. Thus, this rule says Ann likes X if X owns a Z-100. The part to the left of the ':-' is known as the head of the rule. Everything to the right of the ':-' is the body of the rule. As with facts, rules must end with a period.

Having added this rule to our Prolog data base, we can now ask more questions. For example,

```
?- likes(ann, greg).
```

Prolog would respond with 'yes', since we have defined the fact that Greg owns a Z-100. If, on the other hand, we ask

```
?- likes(ann, john).
```

Prolog would respond with 'no' (John owns a Z-248).

More complex rules may be written by adding additional facts to the body of the rule. The extra facts must be separated by commas or semicolons. In Prolog, commas used to separate facts may be thought of as the word 'and.' Semicolons may be considered to be 'or.' For example, the following states 'Ann likes people if they own a Z-100 and a Corvette.'

```
likes(ann, X) :- owns(X, z100), owns(X,
corvette).
```

This next rule states 'Ann likes people if they own a Z-100 or a Z-248.'

```
likes(ann, X) :- owns(X, z100); owns(X,
z248).
```

With this last rule defined, Prolog would respond 'yes' to both of the following queries.

```
?- likes(ann, greg).
?- likes(ann, john).
```

## Arithmetic

Prolog is not a number-crunching language, but it does provide for simple mathematical expressions. Prolog has operators for addition, subtraction, multiplication, and division (+, -, \*, /). Like many other programming languages, Prolog also provides operators which allow you to compare values. These are equals (=), not equals ( $\neq$ ), less than (<), greater than (>), less than or equal to ( $\leq$ ), and greater than or equal to ( $\geq$ ). (Note that less than or equal to is not written as ' $\leq$ ' like in other programming languages). Finally, the Prolog operator 'is' is used to assign values to variables (similar to '=' in Pascal).

Here are some examples using arithmetic in Prolog. First, let's define some facts and rules.

```
memory(z100, 192).
memory(z248, 512).
memory_required(wordstar, 128).
memory_required(multiplan, 80).
memory_required(newsmaster, 512).
```

The first two facts state the amount of memory in kilobytes in a Z-100 and a Z-248 computer. The next three facts state the amount of memory required for various programs. Now let's define a rule to determine the amount of free memory after loading a specific program.

```
free_memory(Computer, Program, Mem) :-
memory(Computer, X),
memory_required(Program, Y),
Mem is X - Y.
```

The following examples illustrate the use of this rule.

```
?- free_memory(z100, wordstar, X).
X=64.
?- free_memory(z248, multiplan, X).
X=432.
?- free_memory(z100, lotus123, X).
X=no.
```

Here is the way Prolog works when you ask the question '?- free\_memory(z100, wordstar, X).' Prolog first locates the rule 'free\_memory' in its data base.



Next, it searches its data base for the fact 'memory' with the variable 'Computer' instantiated to 'z100'. Once this fact is found, Prolog instantiates the variable 'X' to 192. After this, Prolog looks for the fact 'memory\_required' with the variable 'Program' instantiated to 'wordstar'. When this fact is located, the variable 'Y' is instantiated to 128. Finally, the expression 'X - Y' is evaluated and 'Mem' is instantiated to the result (64). If at any time, Prolog cannot match a rule or fact in its data base, it returns 'no.' In the last example, Prolog returns 'no' since no fact is defined using 'lotus123'.

## Input/Output

Programming languages usually provide a method of reading input from the keyboard and writing output to the screen. Prolog contains some built-in predicates for input/output operations.

The built-in predicate 'write' lets you display terms. If X is instantiated to 'z100' then 'write(X)' would display 'z100' on your computer screen. If X were not instantiated, 'write(X)' would display an underscore followed by a unique number, like '\_312.' (This is Prolog's way of informing you that the variable X has not been instantiated to a given value).

Two other built-in predicates are useful for formatting output — 'nl' and 'tab'. The predicate 'nl' means new line and causes succeeding output to be displayed on the next line. The predicate 'tab' allows you to position the cursor to the right of the display. For example, 'tab(5)' causes the cursor to move 5 spaces to the right. Here is a short example using 'write', 'nl', and 'tab'.

```
write('Hello World!'), nl, tab(13),
write('Goodbye World!').
```

Here is what the output would look like on your screen.

```
Hello World!
Goodbye World!
```

By the way, the write predicate in A.D.A. Prolog displays the single quote characters when the text is printed on the screen. For example,

```
write('Hello').
displays as
'Hello'
```

To avoid displaying the quotes, A.D.A. Prolog contains a built-in predicate called 'print'. It works like 'write' but does not display the quotes.

The built-in predicate 'read' allows you to read terms from the keyboard. The statement 'read(X)' would read the next term from the keyboard and instantiate the variable 'X' to that term. The term must end with a period; however, the period will not be a part of the term when instantiated with 'X'.

If you wish to read or write to files with a Prolog program, then you can use the built-in predicates 'see', 'seen', 'tell', and 'told'. The predicate 'see(X)' opens

the file X for reading. Succeeding 'read' predicates will get input from file X rather than the keyboard. The predicate 'seen' closes the current input file. For writing to a file, you must first use the predicate 'tell(X)'. This will open the file X for output. From that point on, any 'write' predicate will write data to file X. The predicate 'told' closes the current output file.

One other important built-in predicate needs to be mentioned here — 'consult'. The 'consult' predicate allows you to load an external file containing a Prolog program into the Prolog data base. The proper form of the predicate is 'consult(File)' where File is the name of the file to load. The file name should be enclosed between single quotes.

## Applications

Having seen just a few of the features of Prolog, you are probably wondering what good is this strange language. Well, as I already mentioned, it is not suited for complex mathematical applications. If you want a program to compute the trajectory of an ICBM, you would be better off using C, FORTRAN, or Pascal. Prolog, on the other hand, is an excellent language to use for artificial intelligence applications. If you wanted to develop an expert system or a program for natural language processing, Prolog would be a good choice.

I have included a small program in this article to illustrate using Prolog for an expert system. First, you need to understand what an expert system is. Basically, it is a program which attempts to mimic the thinking or reasoning of an expert, or group of experts. For example, a medical expert system would attempt to "reason" like a doctor when solving certain types of problems. When developing an expert system, knowledge engineers will spend many months observing and interviewing the human expert. The goal is to understand the way the expert thinks and solves problems. The knowledge engineer then writes a program to mimic that thinking process as much as possible.

Listing 1 shows the small expert system I have written in Prolog. This is a program to diagnose an illness. By asking a series of questions, the program determines what type of illness you may have. (Please note this is an example program only. I am no doctor and I am sure my diagnoses are not accurate. Sorry to any medical professionals out there reading this article.)

I created the program with my favorite text editor (WordStar) and named the file ILLNESS.PRO. Once in Prolog, use the 'consult' predicate to load the program.

Once the program has been added to the Prolog data base, it can be run simply by typing the predicate 'start.' (Refer to

Listing 1 as each part of the program is described.)

The 'start' predicate begins execution of the program. Print predicates are used to explain how to use the program. Next, the predicate 'diagnose' is called with the variable 'Illness.' If the illness is diagnosed, the variable 'Illness' will be instantiated to the name of the illness. In that case, the program ends by stating what illness you have. If the illness could not be determined, 'Illness' will not be instantiated. In this case, you can think of 'diagnose' as being false. (It is true if it does determine the illness). If 'diagnose' is false, then 'start' is false. At that point, Prolog will stop executing 'start' by backing out of that predicate. Prolog will then look in its data base to see if another 'start' predicate exists. Since there is another one, Prolog will begin executing it. Notice the second 'start' predicate simply displays a line stating your illness could not be diagnosed. (By the way, the 'write', 'print', and 'nl' predicates always succeed, i.e., they are always true.)

The next rule defined in the program is called 'true\_answer.' This rule is true if the input read from the keyboard equals 'y'. If the input is not equal to 'y', then 'true\_answer' is false.

The next set of rules are called 'have\_a'. These rules define symptoms a person may have. The specific rules are true if the user responds yes ('y') to the questions asking if he has the specific symptom. The final set of rules are 'diagnose.' These rules diagnose the specific illness — cold, flu, and strep throat. These rules are true if the person being diagnosed has all of the symptoms listed in the rule. For example, the rule 'diagnose(cold)' is true (you have a cold) if you have a cough, have a sneeze, and have a runny nose. If any of the symptoms are not true, then you do not have a cold (according to this expert system.)

The basic operation of the program is to first attempt to diagnose a cold. If the user answers no to questions about any cold symptoms, Prolog will backout of 'diagnose(cold)' and will attempt to use the next diagnose predicate ('diagnose(flu)'). If that proves to be false, the next diagnose predicate is tried. Prolog continues in that manner until 'diagnose' is true, or no more diagnose predicates are left in the data base.

This example program can be shortened by about 7 rules if you wish. All you would have to do is replace all the 'have\_a' rules with one which looked like the following.

```
have_a(Symptom) :-
    print('Do you have a '),
    write(Symptom),
    print('? '), true_answer.
```

Listing 2 shows a sample session using ILLNESS.PRO. By the way, a real expert system like this would probably suggest a

```

start :-
    print('I will attempt to diagnose your illness. '), nl,
    print('Please respond to each question by typing '), nl,
    print('y or n for yes or no. '), nl,
    diagnose(Illness),
    print('You probably have '), write(Illness), print('.').

start :-
    print('Sorry, I cannot determine your illness.').

true_answer :-
    read(Answer), nl,
    Answer = y.

have_a(cough) :-
    print('Do you have a cough? '), true_answer.

have_a(sneeze) :-
    print('Have you been sneezing? '), true_answer.

have_a(runny_nose) :-
    print('Do you have a runny nose? '), true_answer.

have_a( fever) :-
    print('Do you have a fever? '), true_answer.

have_a(muscle_ache) :-
    print('Do you have a muscle ache? '), true_answer.

have_a(head_ache) :-
    print('Do you have a head ache? '), true_answer.

have_a(sore_throat) :-
    print('Do you have a sore throat? '), true_answer.

have_a(vomiting) :-
    print('Have you been vomiting? '), true_answer.

diagnose(cold) :-
    have_a(cough), have_a(sneeze), have_a(runny_nose).

diagnose(flu) :-
    have_a(vomiting), have_a( fever), have_a(muscle_ache).

diagnose(strep_throat) :-
    have_a( fever), have_a(sore_throat), have_a(head_ache).

```

### Listing 1 Sample Prolog Expert System to Diagnose an Illness

```

?- start.
I will attempt to diagnose your illness.
Please respond to each question by typing
y or n for yes or no.

Do you have a cough? n.

Have you been vomiting? n.

Do you have a fever? y.

Do you have a sore throat? y.

Do you have a head ache? y.

You probably have strep_throat.

```

### Listing 2 Sample Session Using ILLNESS.PRO

#### Continued from Page 29

Heath Company Parts Department  
Hilltop Road  
St. Joseph, MI 49085  
(800) 253-7057  
(Heath Catalog orders only)

Word 5.0 ..... \$450.00  
Microsoft  
13221 SC 26th Street, Suite L  
Bellevue, WA 98005  
(800) 426-9400 (Orders only)

Word Companion Disk  
Version 5.0 ..... \$49.95  
Version 4.0 ..... 29.95  
Word Processing Power with  
Microsoft Word 5.0 (book) ..... 21.95  
Rinearson Support Associates  
219 First Avenue N, Suite 410  
Seattle, WA 90109-4893  
(800) 345-9111, Ext. 36

✱

treatment, once it had diagnosed the illness.

This article has taken a very simplistic approach towards describing Prolog. I have only been able to scratch the surface of this unusual language in these short pages. There have been a number of features of Prolog (lists, backtracking, the cut, grammar rules, recursion, trace debugging, and much, much more) that I haven't even mentioned. You should, however, have a general view concerning Prolog and know at least one type of application for which Prolog is well suited. If you are interested in learning more about Prolog, I recommend these two books:

- *Programming in Prolog* by W.F. Clocksin and C.S. Mellish, published by Springer-Verlag.
- *The Art of Prolog* by Leon Sterling and Ehud Shapiro, published by The MIT Press.

As I mentioned earlier in this article, the book by Clocksin and Mellish is con-

sidered by many to be the definitive word about the Prolog language. Definitely read this book if you are interested in Prolog. The second book I listed is more of an advanced book on Prolog. I would suggest reading that book, only after you feel comfortable using the Prolog language.

A.D.A. Prolog is available from:  
Automata Design Associates  
1570 Arran Way  
Dresher, PA 19025

A.D.A. provides about 5 different versions of Prolog. Their PD Prolog is in the public domain and is available on many computer bulletin board systems (it may even be on the HUG BBS). They also sell Education Prolog for \$29.95. This includes additional features not found in their PD version. Another version, FSM Prolog, allows full random file access and floating point arithmetic. This version costs \$49.95. For \$99.95, A.D.A. sells VMI Prolog. This version makes use of virtual memory to increase the maximum size of

your Prolog data bases. Finally, for \$200.00, A.D.A. has VML Prolog — a large model virtual memory system.

As I suggested at the beginning of this article, use PD Prolog to experiment with the Prolog language. PD Prolog even comes with numerous sample programs. If you find the language to your liking, then upgrade to a more sophisticated Prolog system (either one from A.D.A. or Borland's TurboProlog or something else). One other comment about PD Prolog — it requires approximately 210K bytes of free memory to run.

If the readership of REMark expresses a major interest in Prolog, maybe I can contribute more articles about Prolog programming in the future. One final note:  
quality\_computer(X) :- manufacturer(X,  
heath\_zenith).

✱

# Make a Dream\*

(\*The Dream Machine You Always Wanted, but Thought You Couldn't Afford)

## Step 1. Choose Your CPU

☐ **10MHz 8088**  
10MHz/No Waits 8088 CPU, 8087 coprocessor socket, 640k RAM, floppy disk controller, one serial, one parallel, clock/calendar.

☐ **20MHz 80286**  
20MHz/No Waits 80286 CPU, 80287 coprocessor socket, 1M RAM (to 8M on board), two serial, one parallel, clock/calendar, NEAT BIOS.

☐ **25MHz 80386**  
25MHz/No Waits 80386 CPU, 80287 & 80387 coprocessor sockets, 1M RAM (to 16M on board), two serial, one parallel, clock/calendar, powerful BIOS.

## Step 2. Choose a Keyboard

☐ **84-Key Standard Layout**  
10 Function keys on left-hand side, original key layout.

☐ **102-Key Layout**  
12 Function keys at top, new style layout, with MACRO key.

☐ **101-Key Zenith**  
New style Zenith 101-key enhanced keyboard.

## Step 3. Select a Cabinet

☐ **Baby-AT Cabinet**  
Case with 150 watt power supply and mounting for up to four 1/2 height drives (two exposed).

☐ **Desktop AT/386 Case**  
Full size cabinet with 200 watt power supply. Holds up to 4 1/2 height drives (2 exposed).

☐ **Tower AT/386 Case**  
Full size vertical cabinet with 200 watt supply. Holds up to 5 1/2 height drives (3 exposed).

☐ **HighTower AT/386 Case**  
Includes 200 watt power supply and mounting for up to 8 - 1/2 height drives (four exposed). This one you WON'T outgrow!

## Step 4. Add Your Video

☐ **Hi-Res Monochrome**  
Hi-res mono graphics card with CGA, extra parallel port, amber monitor with tilt/swivel base and "flat" screen.

☐ **CGA Color Video**  
640x200 pixel resolution card, CGA color monitor, tilt/swivel base and extra parallel port.

☐ **EGA Color Video**  
640x350 pixel resolution card, EGA color monitor and tilt/swivel base.

☐ **VGA Color Video**  
640x480/600x800 resolution card, VGA color monitor, tilt/swivel base and autosync.

## Step 5. Pick a Hard Drive

☐ **20Meg, 65ms, MFM**  
1/2 height drive, normal speed.

☐ **40Meg, 28ms, MFM**  
1/2 height high-speed drive.

☐ **80Meg, 18ms, MFM**  
Full height super-speed drive.

☐ **150Meg, 18ms, ESDI**  
Full height super-speed drive with super-speed controller.

## Step 6. Call First Capitol

 **First  
Capitol  
Computer**

16 Algona, St. Peters, MO 63376

Orders: 1-800-TO-BUY-IT  
BBS: 1-314-928-9228  
Others: 1-314-447-8697



# Getting Started with . . .

# WordStar Professional Release 5

Gregory D. Elder  
Qtrs 4301-A  
USAFA, CO 80840

WordStar is one of the granddaddies of the microcomputer software world. I first learned to use WordStar way back in 1981 on an Apple II computer running CP/M. (For those of you unfamiliar with CP/M, it was pretty much the standard operating system of choice for 8-bit computer users years ago.) When I bought my Z-100 in 1984, I decided upon WordStar to meet my word processing needs since I already understood how to use it. Over the years, numerous other word processing programs became available with advanced features not present in WordStar. I always stuck with WordStar, however, because it met my word processing needs and I didn't want to learn a totally new word processing system.

With release 5 of WordStar, MicroPro has added some significantly new features to this great word processor. For example, this new version supports newspaper-style columns in documents. It also provides windows for split-screen editing of two files or two parts of the same file. In addition, WordStar now has a page preview feature that lets you see what a document will look like before you print it. The standard release 5 package also comes with a telecommunications program, a file and directory organizer, and a program to create outlines. These are just a few of the many new features.

This article has a two-fold purpose. First, it will provide enough information about WordStar to enable a beginner to start using this word processor. Secondly, it will explain many of the new features of release 5 so that old-time WordStar users can decide if they wish to upgrade.

## Basic Information

Throughout this article I may refer to special characters on the computer keyboard. I will represent control characters by preceding the character with a caret. For example, control-X would be repre-

sented by ^X. This is also the way WordStar's menus depict control characters. (A control character is entered by holding the control key down while pressing the specific character.) I will represent Alt characters by preceding the character by "Alt-", as in Alt-X. (An Alt character is entered by holding down the Alt key while pressing the specific character.) I will show other special characters by enclosing them in angle brackets. For example, the return key will be represented by <RETURN>. By the way, this key may be labeled as <ENTER> on some keyboards.

This article assumes you already know how to boot-up your microcomputer and you understand some of the basics of MS-DOS (formatting disks, copying files, listing a directory, and so on). To run WordStar release 5, you need an IBM-PC compatible computer, DOS version 2.0 or higher, and at least 384K of memory. If you have a Z-100 computer, you can order HUG's Z-100 WordStar Connection program. This lets you run WordStar release 5 on a Z-100.

The entire WordStar release 5 package comes on twelve 5-1/4" disks. The actual program is on one disk, while the remaining disks contain support programs (spell checker, telecommunications, etc.), installation programs, and a tutorial. For ease of use, it would be best to load all the WordStar programs onto a hard disk. WordStar will work fine on a dual floppy-system, but you will have to swap disks

when you want to run a support program like the outline or telecommunications program. (Since I rarely run most of the support programs, I do not find using a dual floppy system to be an inconvenience.)

## Getting Started

With your computer turned on and booted, place the WordStar program disk into drive A (if a dual floppy system), or change directory to your WordStar directory (for hard disk systems). To start WordStar simply type

WS<RETURN>

After displaying a copyright notice, WordStar will present its opening screen as shown in Figure 1. Old-time WordStar users may be shocked at first because release 5 does not use the classic WordStar menus, unless you want it to. WordStar release 5 uses pull-down menus. The opening screen consists of a menu bar displaying the names of available pull-down menus (File, Other, and Additional). Below the menu bar is a dialog box with the message "Press F1 for Help." Finally, the rest of the screen is filled with the names of the files in the current default directory.

At this point, you can create a new file or edit an existing file. Before doing this, though, you may want to change your default directory. If you are using a dual-floppy system, you will have the WordStar program in drive A and your

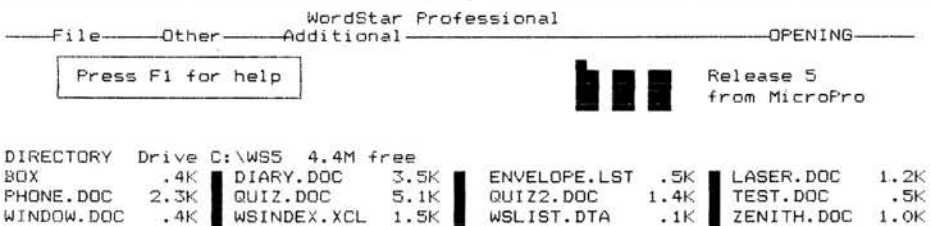


Figure 1  
WordStar Opening Screen

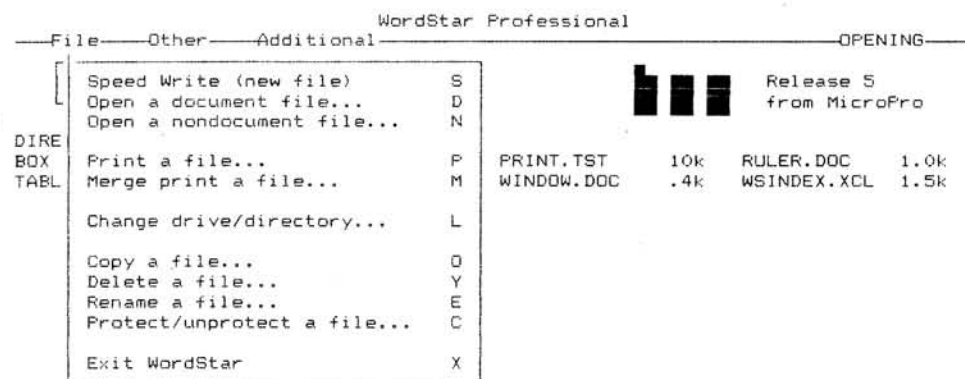


Figure 2  
File Pull-Down Menu

data disk (the disk which will hold your document files) in drive B. To change your default drive to B, first press F. This will display the File pull-down menu as shown in Figure 2. One of the commands in this menu is "Change drive/directory." You may either use the down arrow key to highlight that command and press return, or simply press the letter L. Then, when prompted for the new drive/directory type in B:<RETURN>. (If using a hard disk system, you could specify a different directory on the hard disk). After you change drives or directories, the opening menu will list all the files in the new drive/directory.

Now you are ready to start editing a file. Again, press F for the File pull-down menu. As before, you can use the down arrow key to highlight the command you want, or you can just type the appropriate letter. WordStar 5 has a new command called Speed Write. This allows you to create a new file and to begin entering text immediately, without specifying a file name. You give the file a name when you save it. The other command which creates a file is listed in the menu as "Open a document file...". Select this command by typing D, or use the arrow key. This command will prompt you for the name of the file. Type in a file name and press return. If this is a new file, WordStar will tell you it can't find the file on disk and ask if you want to create a new one. Simply type Y or N for yes or no. As you can see, this command may be used to create a new file or to edit an existing file.

#### Entering Text

After processing the desired command (Speed Write or Open Document), WordStar will present you with the Edit screen as shown in Figure 3. Again, old-time WordStar users will be surprised to not see the traditional WordStar menu. The traditional menus list every command available in WordStar. If you prefer the traditional menus to the pull-down menus, you can change the help level. Simply select the Other pull-down menu by pressing Alt-O and choose the "Change help level" command. Then enter 3 to turn on the classic menus (see Figure 4). I prefer

the pull-down menus because they take up less space on your screen. This allows you to see more of the file you are editing on the screen. (By the way, to select any pull-down menu shown at the top of the screen, just press Alt and the first letter of the menu name.) If you know the control key combinations to process a command, you can even avoid using the pull-down menus. Under WordStar release 5, the control key combinations are known as accelerator commands.

To enter text, simply type as if you were using a typewriter. You need not hit <RETURN> at the end of each line as WordStar automatically wraps around to the next line for you. You only need to press <RETURN> when you want to end a paragraph. If you make mistakes while typing, don't worry. You can correct them later. Also, you may notice after typing 2 or 3 lines that there may be more than one space between words. By default, WordStar does full justification to text. This means, both the left and right hand sides of text extend to the margins. To do this, WordStar inserts extra spaces between words. If you do not like full justification, you can turn it off. You can either type ^OJ (this toggles justification on and off), or select the Layout pull-down menu and choose the appropriate command.

Notice that while you type, WordStar keeps track of where you are in the document. The top line in the Edit screen is a status line. On the far left hand side is the name of the file you are creating/editing. (If you used the Speed Write command no name will be listed.) Next, is the page number, followed by the line number, and then the column number for the cursor listed in number of columns and inches. The remaining information in the status line tells you if insert mode, automatic alignment mode, and justification are on or off. More about insert mode and automatic alignment later.

#### Moving Around the Document

Once you have entered text in your document, you may want to make some changes or correct any mistakes in the file. First, you need to know how to move around in your document. The arrow keys will, of course, move the cursor one character position in the appropriate direction. As an alternative, you can also use ^E to move the cursor up one line, ^D to move the cursor right one character, ^X to move the cursor down one line, and ^S to move the cursor left one character. You can move the cursor right or left one word by using ^right-arrow and ^left-arrow, or ^F and ^A. To scroll up or down a full screen, use the PgUp and PgDn keys, or ^R and ^C. Furthermore, you can scroll up or down a line while leaving the cursor in the current line by using ^PgUp and ^PgDn, or ^W and ^Z. Finally, you can move to the beginning or end of your document by using ^Home and ^End, or ^QR and ^QC.

WordStar Release 5 has a new feature called "Go to" which allows you to move to a specific page in your document. You can either select this command from the Goto pull-down menu, or use ^QI. Then, enter the page number to go to when prompted. You can also use + or - to go to next or previous page. In addition, you can add numbers after the + and - to go forwards or backwards that many pages. For example, +3 would move you forward three pages in your document.

#### Editing

Having described how to position the cursor and move around a document file, I will now explain some basic editing features of WordStar. WordStar provides a number of ways to remove portions of files. Both the delete key and the backspace key are used to remove single characters. The delete key (sometimes abbreviated DEL on keyboards) erases the character located at the cursor position. The backspace key deletes the character to the left of the cursor. To delete an entire word, position the cursor under the first character of the word to delete. Then, press ^T. (You could also use the Delete Word command from the Edit pull-down menu.) You may also delete an entire line of text by using ^Y. Finally, to remove a block of text you must first mark the beginning and end of the text. Simply move the cursor to the first character in the block to erase. Now press ^KB to mark the beginning of the block. Next, move the cursor to the end of the text to delete and press ^KK. At this point, the block of text you have marked will be highlighted

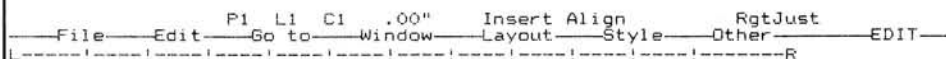
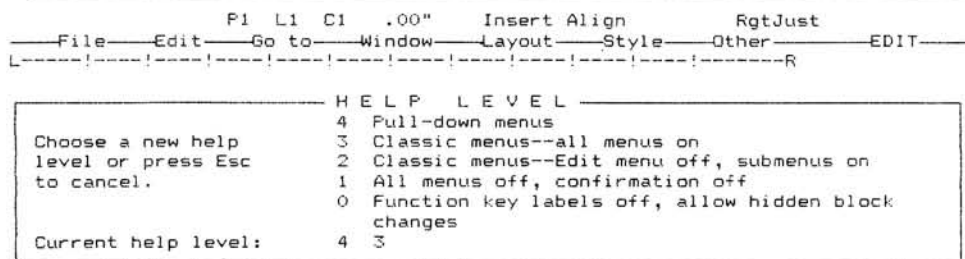


Figure 3  
WordStar Edit Screen



**Figure 4**  
**Help Level Menu**

in reverse video. To delete this block, all you have to do now is type ^KY. By the way, if you accidentally delete something by mistake, WordStar has an Undo command. You can either type ^U or select the Undo command from the Edit pull-down menu. Undo will bring back the most recently deleted text.

Inserting new text into an existing file is very simple. Just position the cursor and begin typing. As you type, the old text will be pushed over as the new text is inserted. WordStar Release 5 has automatic alignment. As new text is inserted into a paragraph, the paragraph is automatically reformed so that all of the text fits within the specified margins. Users of older versions of WordStar will remember that you had to type a ^B to reformat a paragraph after inserting additional text. By default, when you begin WordStar you are in insert mode. You may use the Insert (Ins) key or ^V to toggle insert mode off and on. (The word "Insert" will be displayed in the menu bar to let you know when insert mode is on.) With insert mode off, anything you type will print over existing text.

With WordStar you can easily copy or move portions of a document (phrases, sentences, paragraphs, etc.). You must first mark the desired block of text as already described by using ^KB and ^KK. Next, move the cursor to where you want to copy or move the block. Finally, use ^KC to copy the text and ^KV to move the text. It's just that simple. (You may also use the Edit pull-down menu for these commands rather than using the control key combinations.)

Another editing feature of WordStar which I find handy is the find and replace capability. WordStar can quickly locate text of up to 65 characters long. To find a string of text, type ^QF or use the "Find text" command from the Goto pull-down menu. When prompted, type in the text to find followed by a <RETURN>. WordStar will search your document for the first occurrence of that text. When found, the cursor will be placed at the beginning of the text in your file. You can also perform a find and replace command with WordStar. Use ^QA or the "Find and replace text" command from the Goto pull-down menu. You will be prompted for the text to find and for what you want the text replaced with. When the text to replace is

found, WordStar will ask if you really want to replace it. Just type a Y or N for yes or no. Finally, ^L can be used to perform the last find or find/replace command again (in case the text you are looking for occurs in more than one place in your document).

### Windows

WordStar Release 5 lets you divide the screen into two windows for editing of two separate files or two parts of the same file. You simply select the "Open Window" command from the Window pull-down menu, or use ^OK. This splits your screen into two windows. Once you have opened a new window, ^OK will switch the cursor between the two windows on the screen. You may specify the size of your window in number of lines by using ^OM or the appropriate command from the Window pull-down menu. You can edit files in either window as you usually would in WordStar. Furthermore, if desired, you can copy or move portions of text from one window to the other. Just mark the block of text you want, then use ^KA to copy the block or ^KG to move the block.

### Shorthand Macros

Shorthand macros is a timesaving feature which lets you assign a single letter or number to a longer line of text or commands. For example, suppose you wanted to use the phrase "WordStar Professional Release 5" throughout a document you were creating. Instead of typing the entire phrase each time, you could assign the phrase to the letter W, for instance. Then, whenever you wanted to enter that phrase into your file, you would type <ESCAPE>W. The escape key, as you can see, is the command key for processing a shorthand macro. By the way, WordStar comes with some macros already defined. Some of these macros are <ESCAPE>@ to insert the current date, <ESCAPE>! to insert the current time, and <ESCAPE>T to transpose words.

To define a macro, select the command "Shorthand macros" from the Other pull-down menu. The shorthand menu will be displayed. At this point, type a ?. WordStar will display a list of the current macro definitions and prompt you for the character to use for the new macro. After entering a letter or number, WordStar will

prompt for a description. This is simply a short title for the macro which WordStar will use whenever it displays a list of macro definitions. After entering the description, you will be prompted for the definition. Simply type the keys you want used in the definition. They can even be control keys used for WordStar commands. When the definition is complete, press <F10>. The new definition will now be listed on the screen with the other shorthand macros. WordStar will also ask if you want the new macro stored to disk. If this is a macro you will use frequently with many files, then save it to disk so you will not have to define it each time you use WordStar.

### Dot Commands

Dot commands are special codes placed in documents which affect the format of the document when it is printed. Dot commands start with a period (dot) which must be placed in the first column of any line. The particular command follows immediately after the period. Dot commands are used for such things as setting margins, defining footers and headers, and specifying page breaks. An example of a dot command is ".op". This command turns off the printing of page numbers. Another example is ".pa". This command inserts a page break (new page) into a file. WordStar has over 70 dot commands available for your use.

A new dot command in WordStar release 5 is ".co". This command allows you to add newspaper-style columns to your document. To use this command, you type ".co" at the beginning of a line followed by the number from 1 to 8 to specify how many columns you want. You may also add a comma and the amount of space you want between columns. For example, ".co3,4" gives you 3 columns of text with 4 character spaces between each column. From this point on, WordStar will enter text in columns of 3. To turn off multiple columns, just enter the dot command ".co1". (WordStar will display the columns one under the other on the screen, with a dashed line followed by a C between each column. When the file is printed, however, the columns will line up side by side.)

### Spell Checking and Thesaurus

Once you have created a document, you may want to have WordStar check your spelling (especially if you are as bad a speller as I am). WordStar can check the spelling of an entire document, or just a single word. To spell check the whole document, pull down the Other menu and select the command "Check document spelling", or type ^QL. At the top of your screen will be the spell checking menu. As WordStar finds misspelled words, it moves the cursor to the individual words. WordStar also lists suggested



correct spellings. For example, if you misspelled "forward" as "ferward", WordStar would suggest the words "forward", "forwarder", "forwarders", and "forwarded". When a misspelled word is located, you have the option of ignoring the misspelling (the word may be spelled correctly but is not in WordStar's dictionary), replacing the misspelled word with one of the suggested words, typing in a correction yourself, or adding the word to a personal dictionary. You may also turn on the global replacement feature so that all misspellings of the same word in your document receive the same correction.

WordStar also has a thesaurus feature which helps you find synonyms for words. To use the thesaurus, place the cursor on the word you want a synonym for. Then, chose "Thesaurus" from the Other pull-down menu or type ^OJ. WordStar will give you a list of synonyms for the word. To select a synonym, just move the cursor to the word you want and press <RETURN>.

WordStar release 5 has expanded its spelling dictionary to over 100,000 words. The thesaurus has 220,000 synonyms. However, both the spelling dictionary and the thesaurus use a new compressed format so that 65% less disk space is used as compared to earlier versions of WordStar.

### Saving Files

WordStar has a number of save commands which you can use to store files to disk. When you open an existing document file, WordStar keeps the original on disk while you work with a copy in your computer's memory. When you save the edited file, it replaces the original on disk. The original file is saved as a backup copy. Backup files have the extension '.BAK'.

Here are some of the WordStar save commands. These commands can be selected from the File pull-down menu, or you can use the appropriate accelerator commands as shown. ^KS saves the file and then resumes editing at the current location in the file. It's a good idea to use this command periodically while editing a file so as not to lose all of your work if there is a problem like a power failure. ^KD saves the file and then returns you to WordStar's Opening screen. ^KT can be used to save the file to a new name. ^KX saves the current file and then exits to DOS. ^KQ is used to abandon the current file. This command quits without saving any edits you have made since the last time you have saved the file. Finally, if you have an IBM AT machine, you can configure WordStar to automatically save files every few minutes.

### Printing

WordStar supports over 100 printers. You can install WordStar for just the specific printers you will be using, if you like. To print a file, use the "Print a file" com-

mand from the File pull-down menu. WordStar will then ask a series of questions, such as the name of the file to print, what pages to print, whether you want to print all, odd, or even numbered pages, the printer name, number of copies to print, and if you want the file printed to disk instead of a printer. Printing usually occurs in the background. This means you can work on one file while you are printing another one.

WordStar release 5 files are not fully compatible with earlier versions of WordStar (WordStar release 5 can, however, read files created with older versions of WordStar). To convert a WordStar release 5 file for use with an older version of WordStar, use the print command. WordStar has a special "printer" called WS4. When you select this printer, the file is printed to disk with the format used by older versions of WordStar.

### Page Preview

Some people used to criticize WordStar because it wasn't WYSIWYG (What You See Is What You Get). This means that you see the document on the screen exactly the way it will look when printed. Since many of WordStar's dot commands do not take effect until you print, documents on the screen do not look like the printed file. WordStar release 5 overcomes this limitation with its advanced page preview capability.

Advanced page preview lets you see what a printed file will look like on your screen. Just select the "Page preview" command from the Layout pull-down menu, or type ^OP. The onscreen display will change to show you a reduced version of what the page you are editing will look like when printed. The display shows everything about the printed file — headers, footers, page numbers, different fonts, side-by-side columns, underlines, superscript, subscript, and so forth. Page preview also has a zoom capability which lets you magnify a portion of a page or zoom out to display multiple pages on the screen. In fact, depending of the type of monitor your computer has, you can display up to 144 pages on your screen at one time. Of course, when you display multiple pages, you lose some of the detail, but you can see how the text will align when printed. If you are viewing only one page at a time on your screen, you can use PgUp and PgDn to scroll through all the pages in your document. In my opinion, page preview is the best new feature which has been added to WordStar.

### Other Features

WordStar release 5 has a number of other features which some people may find useful. I will briefly list some of these other features. The ^K? command performs a word count analysis. It will tell you

the number of words in a file or just the number of words within any block of text. The ^OZ command may be used to have WordStar consecutively number your paragraphs. Paragraph numbers are in the format found in legal documents, i.e., 1.1, 1.2, 1.2.1, and so on. WordStar now has an automatic hyphenation feature which you can turn on. If on, WordStar will hyphenate words without asking your approval. Concerning WordStar's help system, it has been expanded to over 200 help messages. You can get help anytime by pressing <F1>. WordStar also has line drawing capability and a simple, built-in calculator feature. Lastly, WordStar can now import Lotus 1-2-3, Symphony, Quattro, and dBase files. You may even specify portions of those files for importation.

WordStar release 5 comes with 5 support programs. The first is MailList. MailList allows you to create mailing lists. You can use it to easily print envelopes and mailing labels. In addition, MailList can be used to build form letters for mass mailings. The second support program is TelMerge. This is a complete telecommunications program which lets you communicate with other computer, such as computer bulletin board systems. TelMerge supports the Xmodem and CompuServe A protocols for file transfers. It even has the option of stripping out unnecessary characters from WordStar files before transmission, for clean ASCII transfers. The third support program is ProFinder. ProFinder is used to organize and manage files and directories. ProFinder lets you assign descriptive titles to files to make them more recognizable. You can also use ProFinder for synonym word searches of files (if you searched for "car" you could also have ProFinder look for the synonym "automobile"). (ProFinder is probably only useful if you have a hard disk system.) The fourth support program is PC-Outline. This is simply a outliner which helps you organize information. It comes in handy for organizing your thoughts before you begin writing a new document. Finally, the last support program is a tutorial which helps you learn — WordStar.

### Conclusion

In this article I have introduced you to WordStar release 5. I have attempted to provide enough information to allow a beginner to start using this word processor. In addition, I have attempted to let old-time users of WordStar know about some of the new features available under release 5. I believe WordStar is one great word processing program. It gets better with each new release. Try WordStar — you'll probably like it. ✱



# Z-100

**Paul F. Herman**  
**3620 Amazon Drive**  
**New Port Richey, FL 34655**

## SURVIVAL KIT

### Cursor Position, Terminal Identification, and Re-Directed Input

In the last issue of "Survival Kit", I began an in-depth discussion of those odd Z-100 escape sequences which cause characters to be transmitted from the console. This column, we're going to look at several more of these escape sequences. One that tells you the current cursor position, and two that are used to determine the Z-100 system configuration. I'll also talk a little about re-directed input, and how these odd escape sequences can affect that capability.

### Maneuvering the Cursor

There are lots of ways to manipulate the cursor on the screen. The most obvious of these is by sending text characters, tabs, line feeds, or carriage returns to the console. All of these commands cause the cursor to move to a new location, and are understood by even the dumbest of terminals.

Smarter terminals will allow you to tell the cursor to go directly to a specified line and column position. On the Z-100 this is done with the escape sequence:

```
ESC Y <row> <column>
... where 'ESC' is the ASCII escape command (decimal 27) and <row> <column> denote single ASCII characters which indicate the destination row and column.
```

Since ASCII codes below 32 are considered to be non-printing characters, the row and column characters in the escape sequence are offset by 32. In other words, if you want to move the cursor to the first column of the first row, you would issue the following command:

```
ESC Y <space> <space>
<space> indicates an ASCII space character (decimal 32). The ASCII byte equivalent of the escape sequence above would be like this:
```

```
27 89 32 32
```

The proper ASCII characters for the row and column can be found by adding 32 to the desired row or column number. This assumes that the first column of the first row is row zero, column zero. As another example, suppose you want to move the cursor to row 8, column 56. Adding 8 to 32 gives 40, which is the ASCII code for a left parenthesis. Adding 56 to 32 gives 88, which is the ASCII code for an uppercase X. Therefore, you would use the escape sequence:

```
ESC Y ( X or ... 27 89 42 88
```

### Finding the Cursor

There are several ways for a program to determine where the cursor is located at any particular time. Perhaps the most straightforward of these is for the program to simply keep track of the cursor position. After all, the program is in control of the cursor position, so it should be able to keep track of where it is at. The program would need to have two variables, one for the row position, and the other for the column position. And it would need to update these variables whenever the cursor was moved, or text was output to the screen.

This sounds simple enough, until you begin to think about what is involved in keeping track of the cursor position. For instance, the program would have to check each string of ASCII text sent to the console to see if there are any special

characters, like backspaces, tabs, carriage returns, or line feeds. The program would also have to check output to the console to determine if there were any escape sequences which move the cursor. This would be a major challenge, since there are many escape commands which affect the cursor position.

As long as a program is constrained in its use of special characters, it is feasible to determine the cursor's position by updating variables. But if your program will require a lot of flexibility in cursor positioning (such as would be the case with a word processing or spreadsheet program, or a game) there is just too much overhead involved with keeping track of the cursor. A better way is needed, and the Z-100 provides one. The following escape sequence can be sent to the console to inquire about the cursor position:

```
ESC n
```

This escape sequence is referred to as a "cursor position report". This is one of those odd escape sequences which transmits characters from the console. You may have wondered why I got sidetracked talking about the ESC Y escape sequence above, since it does not transmit any characters from the console. The reason is because ESC Y and ESC n perform opposite functions, and work quite similarly.

Whenever you send the escape sequence ESC n to the console, the console responds by transmitting four ASCII characters. The format of these characters is exactly the same as the ESC Y command. For example, if the cursor was sitting at row 8, column 56 on the screen when you requested a "cursor position report" with



ESC n, the console would respond by transmitting these four characters:

ESC Y ( X

Look familiar? These are exactly the same characters you would have transmitted if you had wanted to move the cursor to that position.

Here is a BASIC subroutine that will find the cursor coordinates:

```
100 PRINT CHR$(27);"n";
110 I$=INPUT$(4)
120 ROW=ASC(MID$(I$,3,1))-31
130 COLUMN=ASC(MID$(I$,4,1))-31
140 RETURN
```

This demonstrates how to use the "cursor position report" escape sequence to determine the cursor location. Note that we are simply discarding the first two characters returned by the console (ESC and Y). The only ones we are interested in are the third and fourth characters, which tell us the row and column. You'll also notice that we are subtracting 31 from the ASCII row and column codes, instead of 32. This is because BASIC numbers the rows and columns starting with row one, column one, instead of row zero, column zero.

Here is a 'C' language function that does the same thing:

```
getpos(r, c)
int *r, *c;
{
    printf("\33n");
    getch();
    getch();
    *r = getch() - 32;
    *c = getch() - 32;
}
```

In order to use this function, you should declare integer variables for the row and column, and then call the function like this:

```
int row, column;
getpos(&row, &column);
```

Since the 'C' language can only directly return one value from a function, we are passing the address of the row and column variables to the function so it may directly update the values. The alternative would be to have separate functions which return the row or the column position of the cursor.

### Going Directly to the Source

It's said there is more than one way to skin a cat. And there are at least three ways to find the cursor position. We have discussed two of them; having your program keep track, and using the "cursor position report" escape sequence. So you might ask, "how does the console know where its cursor is at?". Obviously, it has to be keeping track of the cursor position, if it is able to tell you the coordinates.

The MTR-100 monitor ROM program is responsible for processing the cursor escape sequences, and the current cursor coordinates are stored in its data segment at the following offsets;

	Offsets into MTR-100 data segment	
	v1.x	v2.x or greater
HORZ_CHAR (column)	028FH	0291H
VERT_LINE (row)	0290H	0292H

The address of the MTR-100 data segment can be found in the interrupt page, at address 0000:03FE. For assembly language programs, it may be easier to find the cursor position by directly reading the MTR-100 data, instead of the other methods we have discussed.

One last consideration... if your program directly modifies the CRT-Controller chip cursor registers (R14, R15), the MTR-100 monitor ROM program will become confused, and will not report the correct cursor coordinates. If you are going to use the "cursor position report" escape sequence, or read the MTR-100 data segment directly, you must use the standard console commands and escape sequences to move the cursor.

### Identifying the Terminal Type

Two of the odd escape sequences we have mentioned can be used to determine the terminal type with software. They are:

```
ESC Z      Identify as VT52
ESC i 0     Zenith identify terminal type
```

Now I hate to sound ignorant, but I don't know what the heck a VT52 terminal is, let alone what it would be used with. As far as I know, such a thing does not exist today, and if there is a standard terminal protocol built around the VT52, it must not be terribly popular.

The most important thing we need to know about the VT52 is that whenever it receives the escape sequence ESC Z, it transmits back the sequence:

```
ESC / K      or ... 27 47 75
```

That's it. Nothing more to it. Presumably, this escape sequence would be used by a communications program to determine if it is talking with a VT52 type of terminal. By returning the characters ESC / K, the Z-100 is merely saying "yes, I understand".

### Be a Little More Specific

The ESC i 0 "Zenith identify terminal" escape sequence is a lot more interesting than the VT52 one. This is because the console not only responds to the inquiry, but it does so with meaningful information about the video configuration. Whenever you send the sequence ESC i 0, the console responds by transmitting the following characters:

```
ESC i E <pov><vrs>
where ...
```

<pov> is a character that denotes the number of planes of video RAM. It will either be '1' or '3'.

<vrs> is a character that tells the video RAM chip size. It will be 'A' for 32K chips, or 'B' for 64K chips.

Okay, it isn't anything worth jumping up and down about, but this is useful information, right? A program can use this escape sequence to determine whether the host Z-100 has color capability, or if it may be used in interlace mode.

Some programmers also like to use this escape sequence to determine if the host computer is a Z-100. In other words, they send an ESC i E to the console, and if anything comes back, the computer must be a Z-100. I'm a little leary of using this approach. Oh, I guess it works fine, but what if your program performed this trick on some MS-DOS computer that used ESC i E for some other purpose? The results could get strange.

Here is a little BASIC program that can be used to determine the Z-100's video configuration using the "Zenith identify terminal type" escape sequence:

```
100 PRINT CHR$(27);"i0";
110 I$=INPUT$(5)
120 PVR=ASC(MID$(I$,4,1))-48
130 VRS=(ASC(MID$(I$,5,1))-64)*32
140 PRINT PVR;"planes of video RAM,
      using";VRS;"K chips."
```

### Redirected Input

One of the more interesting features of MS-DOS version 2 and above is redirected input/output. Redirected output gives you the ability to route the normal screen output of a program to another device, like a disk file or the printer. And redirected input allows a program to take its keyboard input from another source, like a disk file. Since this is a Z-100 specific column, and I/O redirection is a generic DOS feature, I'm not going to spend much time describing how to use these features here. However, there are some peculiarities of using redirected input with the Z-100 that I'd like to mention. But first, a short introduction.

Most of you are probably familiar with using redirected output. You can use it to output a disk directory to your printer:

```
DIR > PRN
```

or you can use it to map error messages and other screen output of a program to a disk file, for later reference. For instance, if your program was named TEST.EXE, you could use this command:

```
TEST > ERROR.TXT
```

But when it comes to redirected input, many of you are wondering "what good is it?" Why would you want to redirect the input of a program? Well, I could list quite a few situations where redirected input is useful, but far and away the most valuable is the ability to use a script file to automate the input to a



program. For example, create a file named SCRIPT.TXT consisting of the following lines (<RET> indicates the RETURN key):

```
B <RET>
E <RET>
<RET>
H <RET>
E <RET>
```

Now execute the command:

```
CONFIGUR < SCRIPT.TXT
```

This will cause serial port B to be automatically configured for the Diablo 630 printer, using the DOS CONFIGUR program. The way it works is by running the CONFIGUR program, and taking the required keyboard input from the file SCRIPT.TXT. In other words, every time the CONFIGUR program expects a key to be typed at the console, it takes one from the file, instead.

The advantage to doing this, is that the configuration process can now be done automatically (say from a batch file) without any attention from the operator. If you have a program that requires you to use a different printer, you could reconfigure "on-the-fly" with a batch file. Like this:

```
ECHO OFF
```

```
ECHO Switch to Daisywheel Printer
                                please . . .
```

```
CONFIGUR < DAISY.CNF
```

```
EDITOR
```

```
ECHO Switch to Dot Matrix Printer
                                please . . .
```

```
CONFIGUR < DOTMAT.CNF
```

In this batch file, the file DAISY.CNF would contain the configuration commands for the daisywheel printer, and DOTMAT.CNF would contain the command keystrokes for the dot matrix printer. EDITOR is the name of the program you are running that needs to use the daisywheel printer. Get the picture?

You can use this technique of using redirected input with many programs. In order to find out which characters need to be put in the SCRIPT file, simply run the program, and keep track of every keystroke you make. You can do this by simply making a note of each keystroke on a piece of paper as you are going through a trial run of the program. Remember to record EVERY keystroke, even the carriage returns and control codes. Remember also, that you must record the keystrokes required to exit the program, and return to the DOS prompt. Otherwise, the program will not return control to DOS when you run it using redirected input. After you have a record of all the keystrokes, use an editor program to create a disk file composed of the keystrokes you have recorded. If all the keystrokes are plain ASCII printable characters, you can just use a text editor, like EDLIN. However, if the program required any control characters (like Control C), you may need to use an editor like DEBUG, which allows non-printable characters to be included in the file.

## And Now for the Bad News

A while back, someone wrote to me saying that he would like to see a version of SETZPC (the program used to configure ZPC) that would allow all ZPC parameters to be specified on the command line. As it is now, SETZPC is an interactive program which requires the user to answer questions about his desired configuration. This person wanted to be able to invoke SETZPC from a batch file, and automatically change the ZPC parameters, without further user intervention. Quite a reasonable request, I would say.

My first response to this inquiry was that he should ask Pat Swayne, author of the program. But then the idea struck me that it should be possible to use redirected input with SETZPC, along with an appropriate script file, to automate the configuration process. So I began experimenting with that idea.

I stepped through the SETZPC process of selecting each ZPC parameter, noting each key that was used. Then I made a file named SCRIPT.TXT containing all of these keystrokes (ten in all). Next, I tried running SETZPC with redirected input:

```
SETZPC < SCRIPT.TXT
```

The program started executing automatically, drawing its keyboard input from the script file, until it reached the last question which required a keyboard response. Then it hung. Nothing could be done except to reboot. The only thing I could figure was wrong, was that my script file didn't contain enough characters. So I stepped through the program once again, and noted my keystrokes. Everything seemed to be okay. What's going on here? Well, it took me a long time to figure this out the first time it happened. To make a long story short, the problem has to do with those peculiar escape sequences we have been talking about.

One of the things SETZPC does when it first starts up is check to see if it is running on a Z-100 computer. (Actually, it is checking to see if the computer is in Z-100 mode or PC mode of ZPC). The way it does this (yep, you guessed it) is by using one of those "identify terminal type" escape sequences. The one it uses is ESC Z (identify as VT52). After SETZPC sends ESC Z to the console, it then expects to receive a character back from the console if Z-100 mode is in effect. Technically speaking, the Z-100 would send back ESC / K, but SETZPC doesn't care what it receives. Anything at all coming back from the console is considered to be fair notice that the host computer is in Z-100 mode.

So what effect does this have on our redirected input experiment? Our script file contains 10 characters, which are presumably the responses to the questions asked by the SETZPC program. But when SETZPC attempts to read a charac-

ter from the console after sending the ESC Z command, it reads one from our script file instead, since the input is being redirected. The result is that our script file comes up one character short at the end.

The solution? Our script file should contain the character(s) that SETZPC expects to see when it issues the ESC Z escape sequence. In other words, our script file should include a dummy character at the very beginning, to fool SETZPC into thinking the console is transmitting the character. Then everything will work okay. Try it and see.

## Ignore the Garbage

We have now solved our problem of using SETZPC with redirected input. But there is one other small thing you will notice. After the SETZPC program is run using this technique, the letter 'K' will be displayed at the DOS prompt. Where did that come from?

Well, when the SETZPC program issued the ESC Z command the console wanted to reply with ESC / K. But since the program was taking its input from the script file, the characters transmitted by the console weren't being received. As soon as the SETZPC program was done, input was directed back to the console, and the transmitted characters popped out at the DOS prompt. The ESC / didn't print because they were considered to be non-printable characters. But the 'K' was displayed.

## Which Programs Do, and Which Don't?

Good question. That is, "How do you know if a program uses one of these odd escape sequences that will goof up your redirected input attempts?". The answer is to use redirected output. (Those of you who are already lost in this discussion, will be raving maniacs by the time I'm through!).

As an example, try running SETZPC using redirected output . . . like this:

```
SETZPC > OUTPUT.TXT
```

As soon as the first screen comes up, you can just terminate the program using Control-C. Now look at the OUTPUT.TXT file with an editor like DEBUG. One of the things you'll find is the infamous ESC Z sequence, which tells you that the program expected some characters to be transmitted back from the console. Therefore, you know that you need to include those characters in your script file in order to use redirected input with that program.

For another example, you might want to try running a compiled ZBASIC program using redirected output. When you check the output file, you'll find that the first thing sent to the console by the compiled program is ESC i 0. The compiler apparently uses this to determine the color video status of the Z-100. This should tell you that your script file for compiled ZBASIC programs needs to be prefaced with the characters:

Continued on Page 48

### \*\*\* ZENITH PC COMPUTER UPGRADES \*\*\*

**SmartWatch from FBE Research** Installs in ROM Socket on CPU Board in Zenith computer series Z-100/138/148/150/160. This clock/calendar contains a ten year battery and keeps your computer informed of both time and date at each boot-up. Instructions and software included. . . . . \$38.00

**Z-150 Series Hard Disk Drive Kit** Includes new generation High Speed (40 MS) Seagate Drive with Auto Park heads. Each kit is complete with controller card, cables, hardware and instructions to mount the Hard Disk under your two floppy drives in the Z-150 series computers. 31 MEG ST-138/150 Kit . . . . . \$338.00

**Z-148 Hard Disk Drive Kit** Includes the Hard Disk Drive and controller in the kit above plus the Z-148 Expansion card described below. Each kit includes all cables, hardware and instructions required to replace one floppy drive with a high speed low power Hard Disk Drive. 31 MEG ST-138/Z-148 Kit . . . . . \$413.00

**ST-138/Z-148 Kit With SmartWatch** . . . . . \$443.00

**Z-148 Expansion Card** adds 2 IBM expansion slots . . . . . \$79.00  
with SMARTWATCH clock/calendar. . . . . \$109.00

**INTERNAL MODEM** Fully Hayes compatible (software included)  
1200/300 baud . . . . . \$67.00  
2400/1200/300 baud . . . . . \$109.00

**EXTERNAL MODEM** Fully Hayes compatible (software included)  
1200/300 baud . . . . . \$89.00  
2400/1200/300 baud . . . . . \$128.00

**VCE 150 Video Eliminator for Z-150**  
Allows use of EGA or any video card. Required memory chip included. . . . . \$54.00

**Z-150 MEMORY ZP640 PLUS** Replacement PAL chip for your Z-150/Z160 computer. Allows the use of 256K RAM chips to increased total memory to 704K. To complete this kit order 18 each 256x1 RAM chips. . . . . \$18.00

### \*\*\* IBM COMPATIBLE SOFTWARE \*\*\*

PART NUMBER	DESCRIPTION	LIST PRICE	SALE PRICE
MS-5063-30	Microsoft Windows	\$99.00	\$14.00
NU-413	Norton Utilities Adv.	\$150.00	\$99.00
WP-528	WORDPERFECT 5.0	\$495.00	\$269.00
BO-290	QUATTRO	\$239.00	\$179.00
CP-311	PC TOOLS DELUXE	79.00	\$68.00

### \*\*\* Z-100 SERIES COMPUTER UPGRADES \*\*\*

**High Density 1.2 Meg Drives.** External floppy drive set-up complete with drive, power supply, case and cable. Ready to connect to your 8" floppy controller Single Drive Unit 217.00 . . . . . \$309.00  
Bare Drive and Cable for internal mount . . . . . \$127.00

**SmartWatch by FBE Research.** If you don't have a clock for your Z-100, get this one. More details under PC UPGRADE listings . . . . . \$38.00

**ZMF100A by FBE Research.** A modification package which allows 256K chips to be used on the old-style motherboard (part number 85-2653-1) to reach 768K. Simple assembly with no soldering or trace cutting. Compatible with Easy PC and Gemini Emulator. Order 27 256x1 RAM chips to complete this kit. . . . . \$60.00

**Gemini Emulator Board.** Makes the Z-100 compatible with the IBM PC library of programs. . . . . \$432.00

**UCI EASY PC.** IBM PC Emulator. Makes your Z-100 IBM Software Compatible. Full 8 MEG operation, color graphics and audio compatible. . . . . \$477.00

**UCI Memory Upgrade Pal Chip Set** For the Z-100's with the newer motherboard part number 181-4918 or greater. This chip set allows the installation of 256K RAM chips on the motherboard. With the addition of 27 256K RAM chips (sold separately) a total memory of 768K is obtained. PAL Chip Set . . . . . \$64.00

**UCI Memory Upgrade Card** The board has sockets for up to 2 MEG of RAM (72 chips). Board with no RAM installed \$288.00. Add \$35.00 for EasyDrive RAM Drive Software if desired. Either 64K or 256K RAM chips may be used to complete this kit, see prices under MEMORY CHIPS listing.

**UCI EASY-I/O** S-100 board that provides IBM PC communications port compatibility with your EasyPC. Easy/O-1, One Serial Port \$91.00. Easy/O-2, Two Serial Ports, One Game Port, Clock/Calendar . . . . . \$127.00

**UCI EasyWin** Winchester Drive Systems at reasonable prices. Complete Hard Disk Systems for mounting inside your Z-100. Systems include S-100 bus board, matched XT hard disk controller, EasyWin software, manual and Misc installation hardware. Order a hard disk (ST-125 or ST-138) under the SEAGATE HARD DISK DRIVE listing to complete the kit. . . . . \$288.00

**CDR Z-100 Speed Module** Run your Z-100 Computer at 7.5MHz. Installs easily with no soldering. Externally switchable between Speed and Normal mode. Payload . . . . . \$44.00

### \*\*\* Z-100 SERIES SOFTWARE \*\*\*

PART NUMBER	DESCRIPTION	LIST PRICE	SALE PRICE
MS-463-1	Z-Basic (16 bit)	\$175.00	\$12.00
MS-253-1	Basic-80 (8-bit)	\$175.00	\$12.00
CD-463-2	Condor File Manager	\$299.00	\$12.00
LT-Z100-3	All 3 Listed Above	\$649.00	\$32.00

### NEW LOW PRICES

### \*\* SEAGATE HARD DISK DRIVES \*\*\*

All hard disks are shipped complete with installation instructions. The 100 series drives are 3.5" drives and come with a 5.25" frame and faceplate for standard mounting. All are standard half height except the ST-4096 which is full height. RLL (run length limited) drives require a RLL controller. All have auto-park heads except the ST-225 and ST-238R. Partitioning software is included with all drives over 32 Meg. XT kits include controller card, cables, mounting hardware and instructions. Most AT computers contain a dual controller therefore only a \$7.00 cable set is required. If a dual AT controller is required add \$124.00 for the controller (1:1interleave) and cable set.

MODEL	CAPACITY/SPEED	DRIVE ONLY	XT KIT
ST-125	21 MEG, 28 MS,	\$227.00	\$277.00
ST-225	21 MEG, 65 MS,	\$199.00	\$249.00
ST-138	31 MEG, 28 MS,	\$279.00	\$329.00
ST-238R	31 MEG, 65 MS, RLL	\$218.00	\$273.00
ST-157R	49 MEG, 28 MS, RLL	\$306.00	\$361.00
ST-251	42 MEG, 40 MS,	\$312.00	\$362.00
ST-251-1	42 MEG, 28 MS,	\$329.00	\$379.00
ST-151	42 MEG, 24 MS,	\$379.00	\$429.00
ST-4096	80 MEG, 28 MS,	\$554.00	\$604.00

### \*\*\* FLOPPY DISK DRIVES \*\*\*

MITSUBISHI MF501	5.25" 48 TPI DS/DD 320K/360K	\$79.00
MITSUBISHI MF504	5.25" High Density 360K/1.2 MEG	\$92.00
MITSUBISHI M-353	3.5" in 5.25" frame 720K	\$94.00
MITSUBISHI M-355	3.5" in 5.25" frame 1.44 MEG	\$105.00

M-355 runs on AT compatible or special controller only. Software not required for MS-DOS 3.0 or above. M355 software driver . . . . . \$15.00

### \* PAYLOAD CUSTOM ASSEMBLED COMPUTERS \*

We assemble 8088 XT, 80286 AT or 80386 IBM compatible computers to your specifications. Please write or call for a work-up sheet showing items available and prices.

### \*\*\* VIDEO MONITORS \*\*\*

ZCM-1490	ZENITH Color Flat Screen VGA	\$679.00
MA2565	SAMSUNG Amber TTL 720x350	\$89.00
CW4644	SAMSUNG Color RGB 640x200	\$274.00
CM4531	SAMSUNG Color EGA 640 x350	\$389.00
CN4551	SAMSUNG Multi-sync VGA 800x560	\$489.00

### \*\*\* VIDEO CARDS \*\*\*

EGA350	PARADISE AUTOSWITCH 640x350	\$164.00
EGA440	PARADISE AUTOSWITCH 640x480	\$184.00
VGAPLUS	PARADISE 800x600	\$254.00
VGAPLUS16	PARADISE 16 BIT 800x600	\$314.00
AUTOEGA	STB SYSTEMS 640x350	\$149.00
AUTO VGA	STB SYSTEMS 800x600	\$199.00

### \*\*\* MURATA FAX MACHINES \*\*\*

M-1200 FAX, PHONE & COPIER	\$597.00
M-1600 SAME AS M-1200 + 30 MEMORY DIALER	\$697.00
M-900 20 MEMORY DIALER + SHEET FEEDER	\$734.00

### \*\*\* MEMORY CHIPS, ECT. \*\*\*

Memory chips are once again at reasonable prices. The market prices have been changing daily, therefore for this reason we are only able to list estimated prices below. Please call for the current price before placing your order. We buy in large quantities and work on the smallest of margins in order to bring you great values.

41256 256x1 100 ns	\$4.30
41256 256x1 120 ns	\$4.05
41256 256x1 150 ns	\$3.95
1Mbit 1Mx1 80 ns	\$14.25
1Mbit 1Mx1 100 ns	\$13.50
44256 256x4 100 ns	\$14.50
LaserRamboard for HP Laser Jet series II & IID	\$147.00

Populate with 44256 chips. Each eight chips yield 1 MB. Board accepts 1 MB, 2 MB or 4 MB configurations. We can also supply the HP Laser Jet printer at a competitive price, please call.

## PAYLOAD COMPUTER SERVICES



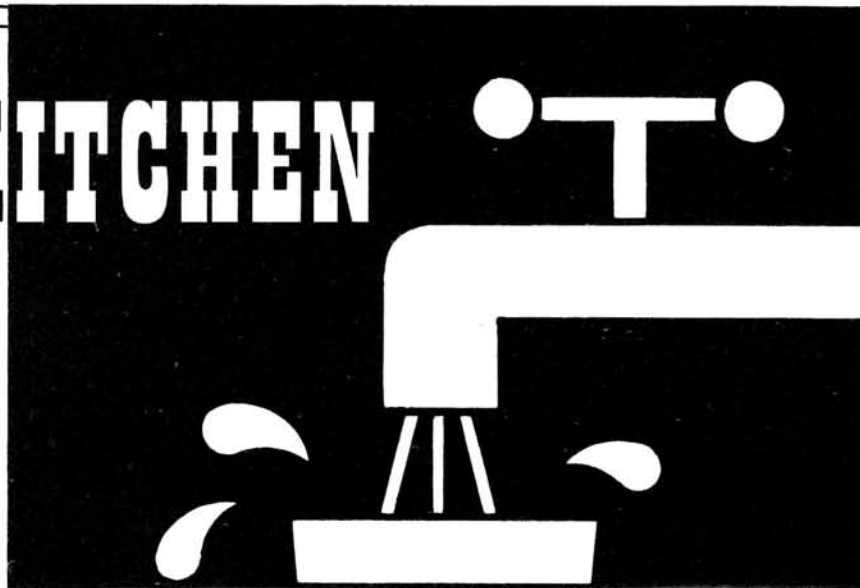
15718 SYLVAN LAKE  
HOUSTON, TEXAS 77062  
PHONE (713) 486-0687



Your satisfaction is guaranteed. All hardware carries a 90 day Payload warranty. VISA and MASTERCARD orders welcome with no surcharges. Add \$5.00 to all prepaid orders for handling and shipping in Continental USA, we pay the balance. Actual shipping costs for foreign, overseas and net billing orders to approved accounts. We accept purchase orders from schools, government and approved accounts. Mail or Phone your order for prompt service. Texas residents please add 8.0% state sales tax.



# EVEN THE KITCHEN SINK PART 2



## Using the Desktop Manager From PC Tools Deluxe, Version 5

Richard J. O'Connor  
848 Fenske Drive NE  
Olympia, WA 98506

### Introduction

Last time, we began our look at an example of "productivity enhancement" software with an overview of the PC Shell and PC Format programs provided with PC Tools Deluxe, Version 5. "Shells are nice," I hear some of you saying, "and a better format program could be useful. But there's got to be more to it than just that!"

You're right. There's a LOT more to this particular package than just a better format program and a pretty face (shell). In this article, we'll look at one of the major programs included in PC Tools: the Desktop Manager. There are nine applications in the Desktop Manager; we'll look at each one in turn and discuss ways to use them in your work. By the time we finish, you may have decided that there is room on your machine for this \$79 product (even less through mailorder software companies)! If not, stay tuned: the third article in this series will cover the five remaining programs that complete the PC Tools package.

Recall that there at least two and sometimes three ways to perform a command in PC Tools: a standard keyboard sequence, mouse commands, and (often) "short-cut" keys that are simply function keys mapped to commonly-used key sequences. In general, I'll describe the mouse commands, using phrases like "point to" and "click on". Sometimes the short-cut keys are the simplest way to get something done, so I'll mention some of them as well. A little experience in using

PC Tools will help you discover the method that comes most naturally to you.

The new three-volume manual set that now ships with Version 5.1 of PC Tools goes into much more useful detail in several areas than the original manual did. As we walk through the Desktop applications, I'll briefly describe what each one does, and give an example or two. The details of using each of these applications are best described by the Desktop manual, so I won't re-state what Central Point Software has already documented for you.

### Getting Started

The Desktop Manager is activated by a special hot-key sequence (one which can be activated at any time during your work). When you originally install PC Tools, the Desktop Manager hot-key is <CTRL>-<SpaceBar>. By pressing the <CTRL> key and the spacebar together, the small portion of Desktop resident in your computer's memory loads the remainder of the program, and then displays a window listing the nine applications you can select from. You can also run the PC Shell program from this menu,

or exit back to your previous work. Note that Desktop can be activated from the DOS prompt or from within any other program, as long as that program does not intercept the <CTRL>-<SpaceBar> key combination for its own use.

To select one of the applications, simply point to its name and click the mouse. You'll then be introduced to your first Command dialog box; the File Load dialog box. Command dialog boxes ask you for further information or list options for you to choose. The File Load dialog box is probably the one you'll encounter most often; see Figure 1 for an example of one listed when the Notepads application was selected. By learning how to work with these dialog boxes, you can fly right through the mundane details of file specification and get right to the application itself. File Load dialog boxes have a text box at the top, where you can type the file name or pattern of interest. Often, a default file extension will be provided to make your search easier; the Notepads editor saves files with a default extension of TXT, Outlines uses the OUT extension, and so on. Files that match the displayed pattern on the default drive are listed be-

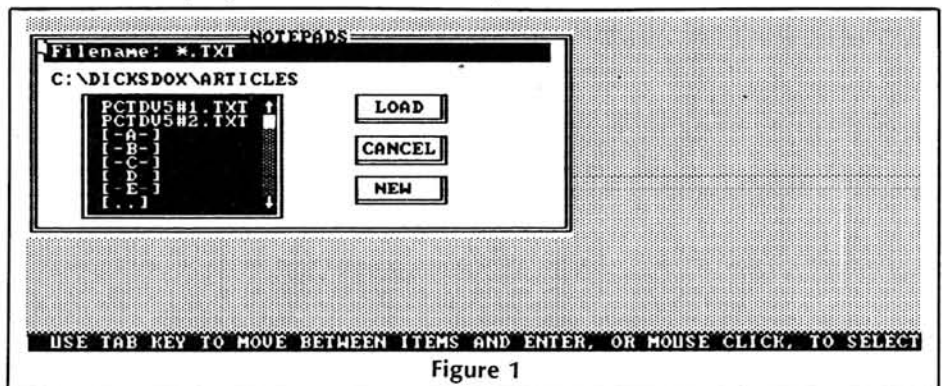


Figure 1



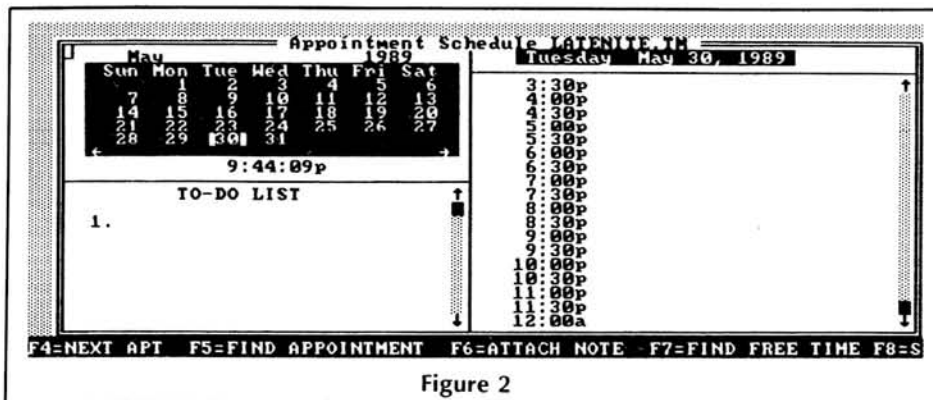


Figure 2

low in the list box. You select one of those files, or a different drive or directory for your search, by the point-and-click method. To the right of the list box are command buttons, that, when selected, allow you to Load your file, Cancel the File Load dialog box entirely, or create a New file.

Once you've loaded your file, the horizontal menu bar appears at the top of the screen, listing the names of the pull-down menus appropriate for the application you selected. The last choice on each menu bar is Window, which is worth taking a look at before we discuss specific applications.

The Window menu allows you to change the background, border, text and status line colors for this application. You can also resize the window assigned to this application and move it to a different location. Finally, you can elect to Zoom your window temporarily (it will expand to fill the screen), or switch the active window when you have multiple windows open on-screen at once. Why would you want to do this? The fact that you can have up to 15 windows open at once in the Desktop Manager may be a little overkill, but it is handy to be able to visually confirm what you're doing when copying information from one application into another. Other times, you may want to check an ASCII value or pull up the calculator for some quick arithmetic without the bother of exiting your current work. In either case, assigning specific color schemes to each of the applications is an easy way to keep track of what you're doing. There are default schemes that you can start off with; but surely your data base work can be more colorful than white on gray!

The Desktop Manager has nine applications: Notepads (edits text files), Outlines (creates document and presentation outlines), Clipboard (cuts and pastes information from one application into another), Database (reads and creates dBase III-compatible data base files), Appointment Scheduler (lists appointments, sets alarms), Telecommunications (performs simple communications, including file uploading and downloading), Macro

Editor (records keystroke sequences for later execution), Calculators (algebraic, programmer's, and HP-12C financial calculators), and Utilities (redefines hot-keys, displays ASCII table). None of these applications by itself can out-perform similar "dedicated" software packages. But I think you'll be surprised at just how useful and consistent the entire Desktop Manager is.

Is the whole greater than the sum of its parts? Let's take a closer look and see...

#### Using the Notepads Text Processor

Notepads is a simple screen-oriented text editor that should put an end to your EDLIN days. You can use Notepads to create, edit, spell-check, and print text files. You can read and write WordStar files, as well as standard ASCII text files. Text can be searched for and optionally replaced. Page headers and footers can be printed on your documents. You can import and export text between files or other applications using the Clipboard. Notepads also has one of my favorite features: AUTOSAVE. I tend to get involved in (excuse me... my two-year-old wants some Jello) side conversations with the kids while I write, and the fact that Notepads saves my file every five minutes (as I have it set up) helps protect me from losing my work when I'm not concentrating on the editing task at hand!

Is Notepads a word processor? No, not quite. I'm writing this article using Notepads, but I'll save the final draft in ASCII format and import it into PC Watch-Word for finishing touches. Why? For one thing, Notepads won't center lines, which I like to use for article headings and sub-headings. Another problem is that there is no easy way to delete a single word or line. In order to do this, you must choose Mark Block from the Edit menu, mark the offending text, and choose Cut to Clipboard to send that piece of text away. Of course, it's nice to have it stored on the Clipboard in case you change your mind, but I'd prefer to have a nice simple Word/Line delete facility. I've been leaning on the Delete key far too much up to this point, and this article has just gotten

started!

Movement about your file can be a little inconvenient for non-mousers. Top/bottom of file, start/end of line, pageup/down are all implemented, but sometimes you'd like to just shlep over about three words or so. With a mouse, you can point and click to reposition the cursor. Without one, you will spend lots of time with your arrow keys.

For all that, Notepads is a good editor for those simple chores that crop up often, like fixing your AUTOEXEC.BAT file. I used to use EDLIN for that, but I find that I prefer a good full-screen editor even for small tasks. Notepads is also handy for spell-checking files, both files you create within the editor and files you read in. If your word processor can't check your spelling, you may get a lot of mileage out of this feature. BE FOREWARNED: it's a fair spell-checker, but it doesn't set any speed records. It took 22 minutes to check the original (5800 word) version of this article and find nine spelling mistakes (along with 84 false alarms). There are surprising "holes" in the word list provided: "filename" and "spreadsheets" are in the list, but "filenames" and "spreadsheet" are not. Adding new words to the list took 8-10 seconds each on my 8 MHz Z-159 (with 30 MB, 65 ms RLL hard disk); you may find yourself selecting "Ignore" to avoid this slowdown. Speed isn't always crucial, though; spell-checking can be a very nice feature to have handy. Central Point should consider making improvements here, since spell-checking is an attention-grabbing feature for users like me.

#### Using the Outlines Text Outliner

Have you ever used outlining software? I never have, because I'm not that organized a writer. The Outlines application uses the same editing commands that Notepads uses, so working in it is simple once you've used Notepads. Text outliners, like Outlines, allow you to list the main points of a talk or a speech, using indentation to separate entries at major levels from minor entries. Outlines allows you to view only parts of your outline to get either the "Big Picture" or the "Details Level", depending on your needs.

I used Outlines to create a brief outline for this article. It was helpful, in that there was a lot of material I wanted to cover, and I have a natural tendency to skip around a bit. For overall planning purposes, I used Outlines on my finished outline to view just the main headlines.

Thus, the display collapsed to something like:

```
Introduction
Getting Started
Using the Notepads text processor
Using the Outlines text outliner
Using the CLIPBOARD...
```

Later, when reviewing the minor

points I wanted to cover under one of my main headlines, I used the Expand command to generate a display like:

Using the CLIPBOARD

List editing capabilities (like Notepads)

Opening the Clipboard

Cut/paste example (note the 2K limit: overflow)

Example of opening, cutting, pasting from other applications

Outlines uses the current position of your cursor to determine the levels of text to expand or collapse. Entire levels can be promoted or demoted if you decide a major re-organization is needed. Since I do much of my text composing at the keyboard, I loaded the outline for this article into Notepads when I began to write. When I decided to move a few topics around, it was easy to mark and move the appropriate sections of the outline to help me keep on track. Outlines is a useful tool, and if I could discipline myself to use it every time I write, I might ALWAYS say things in the right order! It's worth a try...

### Using the Clipboard

The Clipboard tool is used for cut-and-paste operations involving chunks of text that you want to transfer from one application to another. It is an integral part of some of the Desktop applications (like Notepads and Outlines); the Edit menu allows you to mark and then copy or cut sections of your Notepads or Outlines file to the Clipboard. It's a quick way to delete entire sections from your file, or to save certain sections as independent files for later use. Text stored on the Clipboard can be edited with the same commands Desktop users already know when the Clipboard is opened. You can click on the Desktop menu, then choose Clipboard to manipulate the Clipboard text, even while still in Notepads or other Desktop applications.

According to the manual, you can also store text captured from other DOS applications like a spreadsheet display by hotkeying into Desktop, selecting Clipboard, the Copy/Paste menu, and the Copy to Clipboard command. This should cause the Clipboard screen to disappear, leaving you in your previous screen so you can select the text you want using the mouse or arrow keys. In reality, the Clipboard screen disappeared, to be replaced by a blank gray screen. This happens in the version I currently am using because the default PC Tools Setup program installs Desktop with the command DESKTOP/R/CS. The CS (Clear Screen) parameter causes Desktop to display a gray screen as the background when running in resident mode, so you won't be confused by the sight of your last screen prior to running Desktop. Central Point knows about this, and said it was fixed in the 2-27-89 version of the Desktop executable

file. That's what I'm running, so I guess it isn't fixed quite yet. But the helpful folks in Technical Support suggested I run Desktop without CS, and sure enough, Copy to the Clipboard then works as advertised.

The reverse is said to work as well; that you can hotkey into Clipboard from within other applications and Paste the Clipboard contents into your application. This worked as described when I pasted a paragraph to the DOS prompt (though I got a series of Bad Command or File names from DOS, of course!), or into EDLIN when in Insert mode. Only the first 50 characters of a 350 character paragraph made it into PC WatchWord, and only 7 characters arrived inside a QuickBasic 4 program I was editing. Again, Central Point is aware of this (now, anyway!), and promised to determine whether it was the applications or the Paste utility that causes this problem.

One other thing you should watch out for is the 2K limitation of the Clipboard itself; carelessness can confuse you! As a test, I loaded a 3K text file into Notepads, cut almost the entire file into the Clipboard, ignored the warning message by choosing OK instead of Cancel (the message asks "CUT OFF TO FIT?"), and then surveyed the damage. The Clipboard contained the first 2K I had marked, as expected. However, the Notepads file was short the entire block (nearly 3K). I used the <ESC> X sequence to return to DOS to find that my original file had been doctored, even without an explicit Save command! Luckily, the original contents were preserved in a BAK file created by PC Tools, so all was not lost. Don't try this at home, as they say, unless you clearly realize that cutting out text affects your original file immediately. The warnings are there; don't ignore them!

### Using the Databases Manager

I looked forward to working with THIS part of PC Tools, and I wasn't disappointed. Now, I don't have a lot of use for data base manager software at home, where I do most of my "interesting" micro work. Cathy wrote a fine program to organize our name and address file and to generate mailing lists. But at work, we have some applications that are written in dBase III, and the dBase III compatibility of the PC Tools Databases application caught my eye right away. Occasionally (groan!), the need to finish working on a little data at home arises, and an application that could read and work on dBase III data files was well worth checking out.

When you choose Databases from the Desktop menu, a File Load Dialog box appears, showing you all fields with the extension DBF that exist in the current working directory. This extension is commonly used by data base managers to indicate the data base files themselves, as

distinguished from any associated record files or form files. I picked a 105,719 byte dBase III file containing 1521 records from the file list and clicked open the Load box. It took my Z-159 11 seconds to load the DBF file and created the associated REC (file display information) and FOR (the formatting information) files. Once the DBF file was loaded the first time, subsequent loads required only a fraction of a second. A status information line near the top of the Databases display window shows the complete path name of the current DBF file and information about your current position, in the form of "Record 231 of 1521". Several shortcut function keys are defined to instantly jump to the first, last, next and previous records in the data base. These are listed on the bottom line of your screen for reference.

I searched the entire data base for the first occurrence of the phrase TRIS (a chemical buffer solution used in starch gel electrophoresis, but that's another story), which I knew in advance occurred in record #937. It took 11 seconds to find and display this record, which is fairly good performance. "Go to" searches, as in "Go to record #387" occurred instantly. While searching, the popup search window remains open with the same criteria active, so that a simple <RETURN> finds the next record matching your criteria. To work on the current record, click on the Cancel box to interrupt the search. You move from one field to the next using the <TAB> key and changes are made by typing over existing information (be sure to end each new field entry with the <RETURN> key). The changes you make are instantly incorporated into your data base file without the need for an additional "Save" command at the end of your session. I left the Databases application for awhile, and when I reloaded my data base later on, I found that this program remembers all changes made, as well as the number of the record I last worked on, and the last search criteria I used.

Records can be easily added, deleted, or hidden (to protect them from display or deletion). In case you get a little too "delete-happy", you can also undelete all of the records you've marked for deletion with a single command. As your Databases grows, you may eventually want to "pack" it, permanently removing all deleted records and reducing the size of your Databases file. I deleted two records from my text Databases, and it took 17 seconds to pack the Databases and release the unused disk space.

One final capability worth mentioning is the Autodialer, even though I haven't really used it yet. If you have a Databases that contains phone numbers, you can use this feature to dial the phone number found in selected records. Imagine you created a Databases file containing the names, phone numbers, and birth-



days of all family members and other acquaintances. You could ask Databases to check to see if anyone is having a birthday each morning when you boot up, and automatically place a call through your Hayes-compatible modem to them. Now, to be kind, you should also have a phone handset connected to this line! Once the number starts ringing, do their ears a favor and pick up the handset and disconnect the modem by pressing the <ESC> key. Then you can wish them a happy birthday, and they'll all begin to wonder how you got so organized all of a sudden!

If you use your Zenith computer at home, you may be wondering "Why would anyone use a computerized appointment scheduler?" I know I always felt that way. Imagine booting up your computer while your corn flakes get soggy on Saturday morning so you can enter the chores you need to do that weekend. It might be a diversion, but you've got to get down to the garage and find the broom eventually. That nice little package that can BEEP you to remind you when it's time to pick up the girls from softball practice won't do you much good if your computer isn't ON at the time.

At work, it might be a different story. Many of you who use microcomputers at work keep them powered on all day, so appointments with alarms and places to electronically jot down things to do can be very useful. And yet, I find myself using the Appointment Scheduler of PC Tools at home a lot more often than I ever imagined. Why? (Dare I say it?) Because it's irresistible!

Just take a look at Figure 2. There's the current month, with today highlighted . . . a blank TO-DO list, just waiting to be filled . . . an appointment list with configurable time periods to show off the time commitments you've made so far. (Say, my schedule isn't that empty, is it?) With the Appointment Scheduler, it's easy to make daily, weekly, or monthly appointments, just the ticket for those regular events in your life like your Boss's weekly "chats" or your monthly night out with the <insert plural gender-specific noun here>. You can set alarms, cause notes to pop up at the assigned time, even run a program while you're occupied elsewhere. And if you run short of ideas, the manual has many other suggestions, along with good step-by-step examples to guide you along.

It's easy to display your time usage for a five-day block, or to let the Appointment Scheduler find your nearest block of free time. Having this capability only a few keystrokes away can be invaluable when you're trying to set up a meeting with that elusive person that you've finally managed to reach on the phone. Just as in most other PC Tools Desktop applications, full printer support is available when you need paper copies of your

schedule. But my favorite use is running a program at a preset time, particularly running a time-consuming program at a time when I'm sound asleep!

When you first choose the Appointment Scheduler from the Desktop menu, the File Load Dialog box informs you that schedules are saved as files with default extension TM. You can either load an existing schedule file or start all over with a new empty schedule. Once you have a schedule file opened, setting up an appointment to run a program at a preset time is simple. Click on the Appointment menu at the top of the screen, and choose Make Appointment. You set the starting date, the time, the frequency (if you run this program regularly), the duration (so your schedule will reflect that your computer will be busy during that time slot), and an alarm if desired. On the Note: line, place any on-screen reminder you might want, followed by the vertical bar | character, and the name of the BAT, EXE or COM file you want to execute. (If you precede the vertical bar with a reminder, you will need to be present to verify that you want to execute the desired program or batch file.)

That's all you need to do. Now you can run spelling checkers over lunch, back-up your hard disk after work, and calculate complex Mandelbrot displays overnight!

### Using the Telecommunications Program

If you own a modem, you already have a telecommunications package that causes your micro to act like a "dumb"

terminal on command. Examples many of you are familiar with include HUG's own HUGMCP, Datastorm Technologies' ProComm, and the several versions of Kermit distributed by Columbia University. The telecommunications applications in Desktop has many of the features these examples have, but not quite the power you may be used to. However, bundled with the rest of the Desktop package, it adds the convenience of a "built-in" package that will handle many of your needs.

Desktop telecommunications can connect to Bulletin Board Systems (BBS's) and other computers at speeds up to 19,200 baud, limited only by your modem. All four of the standard serial ports (COM1 through COM4) are supported, as are tone/pulse dialing, full/half duplex, every combination of parity, data bits and stop bits I've ever needed, and XON/XOFF flow control. Files can be sent or received using the ASCII (text-only) or XMODEM protocols, and XMODEM transfers create a file called TRANSFER.LOG which keeps track of the success or failure of your transmission.

The major weakness of this package is the lack of terminal emulation for full-screen editing or displays. Only TTY (teletype) "emulation" is provided at this point, which is line-by-line display. I find that I don't need terminal emulation for BBS work, so this is not always a limitation.

When you first choose Telecommunications from the Desktop menu, the default phone directory file (PHONE.TEL) is loaded (see Figure 3). If you create other

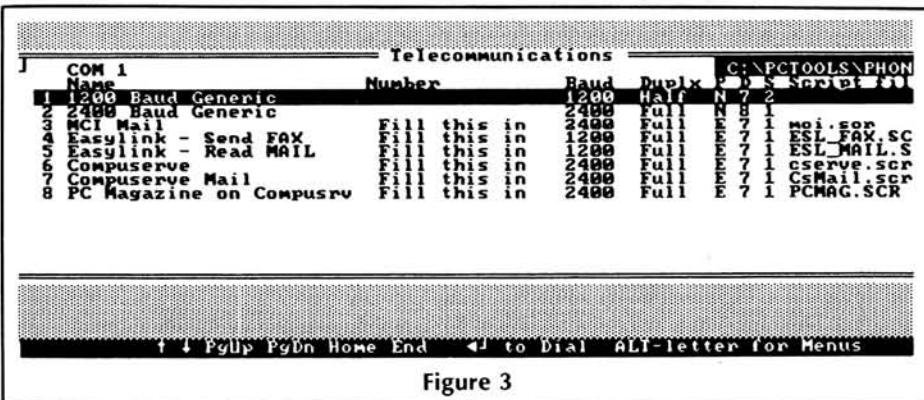


Figure 3

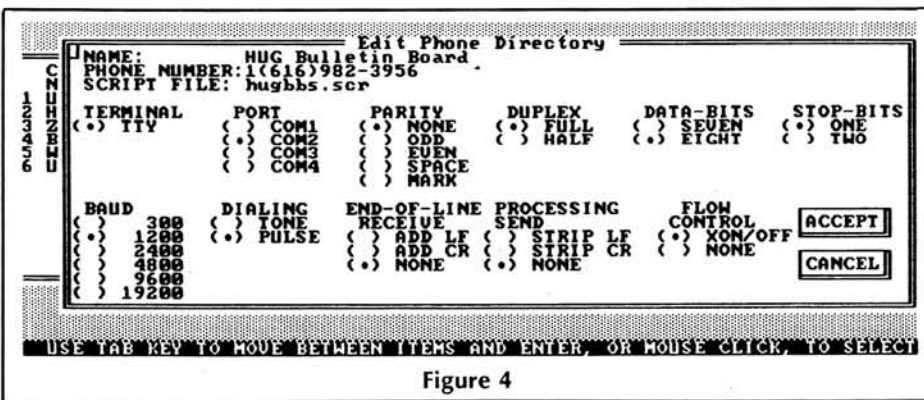


Figure 4



directory files with the .TEL extension, you will see the by-now familiar File Load Dialog box to help you choose the appropriate directory file. Note that the default directory contains sample entries and login scripts for services, such as MCI Mail, EasyLink and Compuserve. To add your own entries, choose Create New Entry from the Edit menu above your display, and follow the prompts. I have a 1200 baud modem configured as COM2 in my Z-159, so my directory edit screen for the HUG Bulletin Board looks like Figure 4.

The script language of Telecommunications is its most useful feature, as far as I'm concerned. This language gives most users the proper mix of power and simplicity. After experimenting for 15 minutes, I created a script (see Figure 5) to dial the HUG Bulletin Board, log in, download both the Online Bargain Center List (Who can resist?) and the current message base, and log out. To run this script, I choose Telecommunications, load my personal phone directory file, and click on the HUG BBS entry. The program automatically dials the number, waits for connection (pretty dependable after 11:00 PM PST!), and does the chores while I catch the late news!

PC Tools also includes a program called BACKTALK, which is a memory-resident program that allows background communications. If you load this program when you boot your machine (note that it will occupy 64K of your main memory), you can start up Telecommunications, begin a file transfer to or from a remote site, and then press <ALT>-B to return to your previous application. An audible alarm signals the end of the transfer, at which point you can return to Telecommunications to complete your remote session. I tested this by downloading the HUG message base while I worked on this article in Notepads, and it works as advertised. Since I don't need this luxury very often, I only load it when necessary. 64K is too much memory to devote to a program that I might not use. (Sometimes I really appreciate the 768K main memory of my Z-100 at work . . .)

### Using the Macro Editor

What collection of utilities is complete today without macro-building capabilities? For those of you who haven't used a macro program, these tools allow you to store sequences of keystrokes for later recall by pressing a single key or key combination. The Macro Editor supplied with PC Tools Desktop is compatible with macros built using ProKey (versions 4.0 and higher) in most cases, so you may be able to use some of your existing macros if you're a ProKey user.

Several macros are packaged with the Macro Editor, including printer macros for standard printers, such as the IBM Pro-

printer, the Hewlett-Packard LaserJet, and the Epson FX-80. Using these macros in combination with Notepads can deliver printer output that rivals the work of expensive word processing programs.

If the Desktop application is run as a memory-resident program, the macros you build are available at any time. You can specify where you want macros in a specific macro file to be active via the Macro Activation command. Four choices are given: everywhere, nowhere (deactivation), only during Desktop sessions, and only outside of Desktop. Macros are built using the Notepads editing commands, and are stored in files with extension PRO.

Macros have a very specific format that takes new users time to master. For example, the macro

```
<begdef><ctrlh>Long Live The Heath User's
will display Jim Buszkiewicz' rallying cry
every time you press <CTRL>-H. (This
would be an interesting macro to store on
the machine of someone who back-
spaces a lot, but I digress . . .) An easy way
to create your first macro is to use the
Learn Mode, which records your key-
strokes as you press them, once activated.
You choose Learn Mode from the Con-
trols menu, press <ALT> and the plus
key when you're ready to begin recording,
press the keystroke combination for this
macro (for example, <CTRL>-<W>), en-
ter the appropriate keystrokes (for exam-
ple, a sequence for starting up PC Watch-
Word), and end the recording by pressing
<ALT> and the minus key. The next time
you enter Desktop, this macro will be
copied into the file LEARN.PRO, which
you can then examine or edit (if necessa-
ry) in the macro editor. Hint: if you made
any typos during your macro recording,
you will have some editing to do!
```

With a little experimentation, you can combine the power of the Appointment Scheduler, Telecommunications, and Macro Editor programs to really automate your work. For example, I created a macro assigned to the <CTRL>-<F1> combination that logs into the HUG BBS at 11:30 PM and runs the download script I described earlier. The macro looks like this:

```
<begdef><ctrlf1><desk>T2<enter><cmd>d1:0:0<enter><esc>X<enddef>
```

This macro starts the Desktop program, chooses Telecommunications, selects and dials the phone number of entry 2 (the HUG BBS), delays one hour (which gives the script assigned to that phone number time to execute), escapes back to the Desktop main menu and exits back to DOS. To make a weekly appointment that would run this macro at 11:30 PM each Monday from April 24 through December 25, 1989, I made the appropriate selections on the menu in Appointment Scheduler (see Figure 6) and saved the appointment in a file. I used the vertical bar in the "Note:" line as the signal to execute the macro assigned to <CTRL>-

<F1>. With a little imagination (and your Desktop manual handy!), you can automate a number of tasks in ways you never thought possible.

### Using the Calculators

Pop-up calculators in a memory-resident "tools" program are not a new idea. However, the implementation of pop-up calculators in PC Tools Desktop has some unique features, making the package even more valuable.

There are three calculators provided in Desktop; an algebraic calculator, a programmer's calculator, and a financial calculator. The algebraic and programmer's calculators are similar to others you may have seen. The algebraic version performs the basic arithmetic functions using the

```
Group!<enddef>
```

visual icon of a "paper tape" calculator. The "tape" scrolls upwards, providing a record of the calculations done during your session. One nice feature is that numbers on the tape can be edited, and all results using those numbers are instantly updated. You can cut up to 100 lines from your "tape" and paste them into another application using the Clipboard (once Clipboard Paste is fixed, that is!). The programmer's calculator performs arithmetic in any of four bases (binary, octal, decimal or hexadecimal) and displays the result in all four. You can also enter a number in one of the bases and see the three equivalents, which is handy for converting from decimal to hexadecimal, and so on. If the value you enter happens to represent an ASCII character or IBM graphics character, that character is also displayed for reference.

The financial calculator is the pride and joy of this application: it emulates the Hewlett-Packard HP-12C calculator (see Figure 7). In addition to standard arithmetic calculator functions, the financial calculator makes short work of standard "five-key" problems involving interest, payments, mortgages, finance charges and so on. There is a good set of examples included in the manual, and a suggested reference for more information.

However, if you already know how to use an HP-12C, you know how to use this emulation.

Calculator "keys" are selected with keyboard sequences or mouse clicks as spelled out in the manual. Since our oldest daughter will begin college in ten years, I decided to figure out how much money I'll need at that point to pay for four years of tuition. This amounts to solving a five-key problem where the unknown is the "payment" I'll be making into her savings account each month. Optimistically, I assumed \$40,000 would do the trick, and that I could find an 8% APR

- \* This script will log you on to the Heath Users' Group Bulletin Board and capture the current online Bargain Centre listing, as well as the current message base. These are all stored in C:\COMM\HUGBBS.LOG.
- \* Insert debugging code here - the ECHO statement

```

ECHO
WAITFOR "Enter Your FIRST Name"
SEND "FIRSTNAME"
WAITFOR "Enter Your LAST Name"
SEND "LASTNAME"
WAITFOR "Enter Your HUG ID Number"
SEND "998877"

```

- \* Get the Online Bargain Centre List.

```

WAITFOR "Function or <H>elp"
SEND "0L";
DOWNLOAD ASCII "C:\COMM\HUGBBS.BC"

```

- \* Receive the message base using ASCII protocol in a continuous stream.
- \* Scanning from message 1 will always start at the lowest-numbered message in the message base.

```

WAITFOR "Function or <H>elp"
SEND "RC";
WAITFOR "Scan From Which Message"
SEND "1";
DOWNLOAD ASCII "C:\COMM\HUGBBS.LOG"
SEND "M";
WAITFOR "Function or <H>elp"
SEND "G";
WAITFOR "Leave Private Message"
SEND "N";
HANGUP

```

Figure 5

Figure 6

Figure 7

Continued from Page 41

ESC i E <pv> <vrs>

(See the description earlier in this column for the meanings of these characters.)

And don't forget... this same kind of problem using redirected input can be caused by any of the "odd" escape sequences we have talked about. Generally, the problems associated with programs

that use ESC Z or ESC i 0 to identify the terminal type are pretty easy to overcome. But if a program uses ESC n (cursor position report), or any of the transmit character escape sequences, it usually won't be practical to use redirected input with it, since it would be difficult to tell in advance exactly what input the program was expecting.

**Q&A, Where Art Thou?**

savings account (your mileage may vary considerably!). Using the manual's example on "Solving IRA and savings account problems" as a guide, I found that if I start saving \$218.64 per month, I'll reach \$40,000 after 120 months. The only trouble is, I have three other daughters; I suspect I'll have to write a lot more articles in the future!

### Using the Utilities

There's one last application provided with PC Tools Desktop that provides four useful utilities. Using this last menu selection, you can change the hot-keys assigned to pop up the Desktop itself or activate the Clipboard cut and paste facilities when you're running other programs. You can also display a complete ASCII table, change the menu and window colors used by the Desktop system (as I mentioned earlier), or unload PC Tools Desktop from memory. Addition of these functions rounds out an impressive collection of programs and gives you the control you need to keep PC Tools Desktop from interfering with other programs that share the use of your micro-computer.

### Wrapping Up

It's been a long journey through the programs that make up the PC Tools Deluxe Version 5 Desktop Manager. Hopefully, you will have found the information you need to make a decision on this package, especially in light of the PC Shell and PC Format coverage in the first article of this series.

But there's a few more programs to cover before we finish with PC Tools Deluxe Version 5, so you may want to stay tuned. Next time, I'll cover the PC Cache disk caching program, PC Secure (the file encryption/compression/decryption program), PC Backup (the quick floppy and hard disk backup system), and the hard disk utilities Mirror and Compress. If you have any questions about anything I've presented, please drop me a note and I'll try to respond promptly. The Technical Support folks at Central Point Software are very helpful, and are also quite willing to help you work out any problems or bugs you seem to be encountering. See you next time!

I know I've been promising another Question and Answer session for several columns now. And the questions are beginning to pile up. Don't worry — everybody gets a personal reply ASAP. But the purpose of the Q&A section is to spread the knowledge around a bit, so the same questions don't get asked over and over.

Well guess what? We're out of space again in this issue. I guess I just don't know when to shut up. But I'll tell you what. The next installment of "Z-100 Survival Kit" will be devoted entirely to answering some of your more interesting questions. I promise!

'till then, keep in touch!





# HADES II

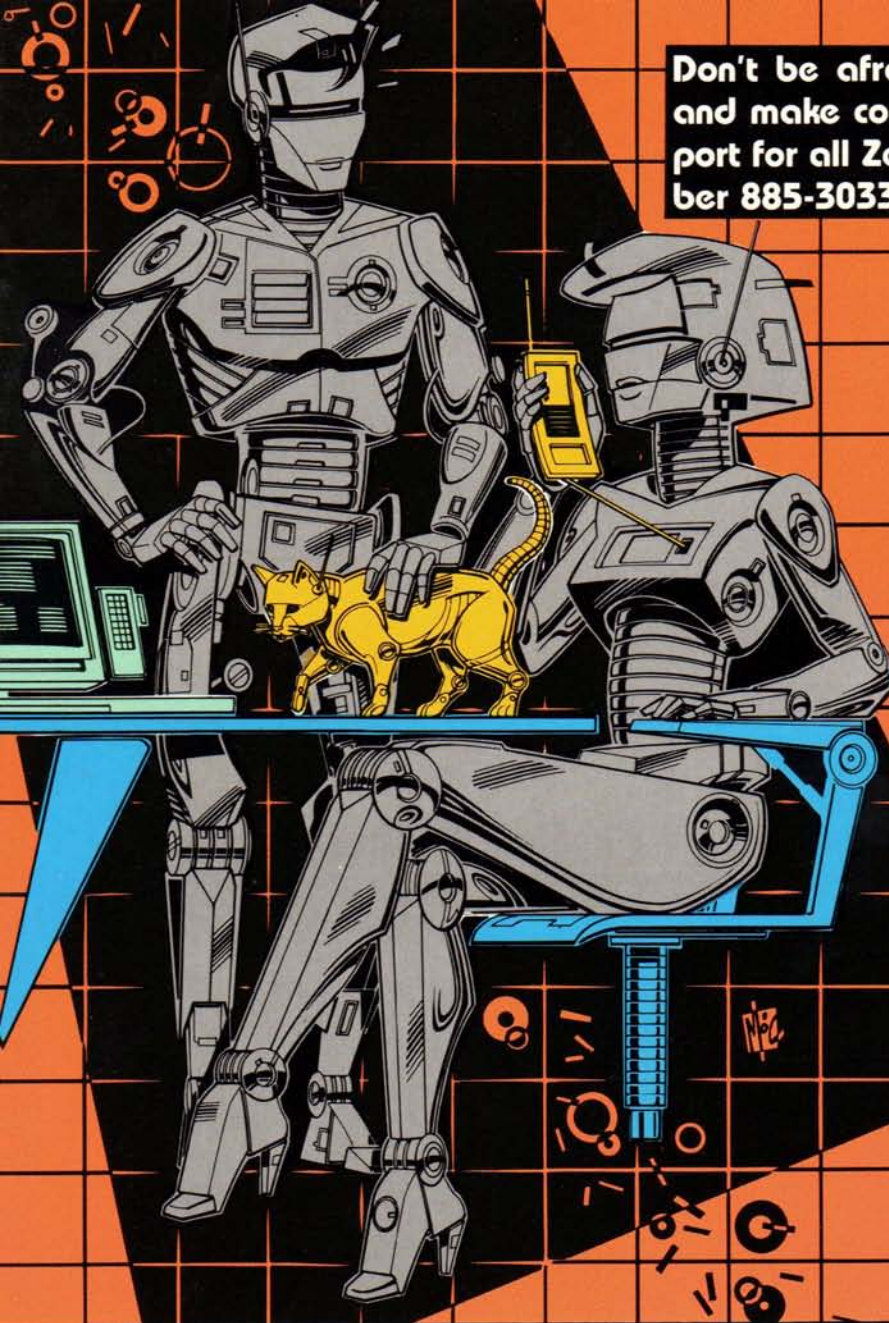
It's HOTTER than ever! Jam-packed with new features, HADES II still remains the easiest-to-use disk editor ever! Just look at some of the features:

- Sector Display/Editing
- Sector HEX/ASCII String Search
- File Display/Editing
- Physical and Logical Cluster Display
- File HEX/ASCII String Search
- Drive Parameter Display
- 512 MegaByte Drive Size Limit
- File Attribute Display/Edit
- Automatic Erased File Recovery
- Manual Rebuild File Recovery
- Works with Headerless MS-DOS Disks
- PC-Compatible or H/Z-100

HADES II is still only \$40, and original HADES owners can upgrade their distribution disk for only \$15. Call HUG today at: (616) 982-3463.



Don't be afraid to communicate! Get HUGMCP and make contact the easy way. Now with support for all Zenith Laptops, order HUG Part number 885-3033 today.



#### HUGMCP Commands

- F1 -- Prints This List, Your Storage Buffer Size, And How Many Bytes Are Presently In The Storage Buffer.
- F2 -- Allows Sending A Defined Message, Or Character Sequence. These Messages Are Entered Using The (F6) Setup Command.
- F3 -- Toggles The Storage Buffer On and Off. When The Buffer Is On, The (Buf) On The 25th Line Will Be High-Lighted.
- F4 -- Allows Saving Data To Disk From The Storage Buffer, Or Directly From The Mouse By Way Of XMODEM Protocol.
- F5 -- Allows Sending Data From Disk, Using Either XON-XOFF, Which Optionally Can Be Ignored, Or XMODEM Protocol.
- F6 -- Enters The Setup Mode So This Software Can Be Configured.
- F7 -- Clears Out Any Data That May Be In The Storage Buffer.
- F8 -- Send Data In Storage Buffer To Printer.
- F9 -- Exits Back To MS-DOS.

Storage Buffer = 524288 Bytes  
Storage Buffer Usage = 0 Bytes

Select Message (0-0), (F1) To List, Anything Else To Abort --> \_

F1:Help F2:Msg F3:Bufp F4:Save F5:Send F6:Clear F7:Clp F8:Print F9:Exit COM

#### HUGMCP Configuration Help #1

- 1 This Function Allows The Baud Rate To Be Changed. Depending Upon Which Mouse Mouse Driver, Normally, It Would Be Set To Either 9600, 19200, Or 38400 Baud. Select Connection To A Host, Will Allow Higher Baud Rates.
- 2 This Function Allows You To Change The Word Parity. Normally, you can select No Parity. This Is Acceptable For Most Serial Systems, BUT It Is Also Necessary For XMODEM Protocol To Work Properly.
- 3 This Function Allows The Changing Of The Word Length. Normally, The Word Length Is Set To 8 Bits. This Value Is Acceptable For Most Serial Systems, And Is Necessary For XMODEM Protocol To Work Properly.
- 4 This Selection Allows You To Enter Messages Which Can Be Automatically Sent With The (F3) Msg. Up To 25, 30-Character Messages Can Be Saved. Selection 00 Is Special. It Should Contain Your Computer's ID Number And Forward Selection 01 Is Also Special. This Selection Can Help Monitor The Host When This Program Is First Executed By Selecting The Proper Action During Setup.

Type (F6)S: S40: For More Help, Anything Else To Configure:

F1:Help F2:Msg F3:Bufp F4:Save F5:Send F6:Clear F7:Clp F8:Print F9:Exit COM

#### HUGMCP Configuration Menu:

- 0 --> Modify Baud Rate
- 1 --> Modify Parity Type
- 2 --> Modify Word Length
- 3 --> Modify Or Add Data-Messages
- 4 --> Miscellaneous Functions
- 5 --> Change Screen Color Assignments
- 6 --> Display Current Configuration
- 7 --> Make Changes Permanent

Select 0-6, (F1) For Help, Anything Else To Quit --> \_

Baud Rate: 19200  
Parity: NONE  
Word Length: 8  
Duplex: FULL  
Response To Keyboard Disable: NO  
Storage Buffer Data Parity Bit: SET TO ZERO  
Send Mouse Initialization Text: NO  
Delete Character: NORMAL  
Mouse Port Set To: COM1

F1:Help F2:Msg F3:Bufp F4:Save F5:Send F6:Clear F7:Clp F8:Print F9:Exit COM



P.O. Box 217

Benton Harbor, MI 49022-0217

BULK RATE  
U.S. Postage  
PAID  
Heath Users' Group