

THE MICRO-TECHNICAL JOURNAL
MICRO CORNUCOPIA

Maintaining PCs

Keeping Your Hard Drives Running page 8

Lose your hard drive and you're losing a lot more than \$500. You're losing data.

Troubleshooting PCs page 14

Finding the defective IC is usually easy if you know what you're looking for.

XT Theory Of Operation page 24

It's a lot easier to make sense out of a problem if you know how the circuit works.

Simulating A Bus page 32

Once we have the system running we may as well observe something with it.

Plus:

Ray Tracing (Part 2) page 18

West Coast Computer Faire page 40

Inside PCX Graphics Files page 42

Testing Your Product Idea page 67

And Much, Much, More



IT'S WHAT'S UNDER THE HOOD THAT COUNTS!

CASES & POWER SUPPLY

150 Watt Power Supply (XT)	50.00
200 Watt Power Supply (AT)	80.00
XT Slide Case	34.00
XT Flip Top or XT Slide with Lock & LED	38.00
AT with Lock & LED	65.00

MONITORS

EGA/CGA (Auto Switch)	452.00
VGA/EGA/CGA Color	600.00
CGA Color	295.00
Amber 12" TTL	89.00
Green 12" TTL	89.00

VIDEO CARDS

Color/Graphics/Parallel	52.00
256K EGA Graphics	125.00
Mono/Graphics/Parallel	49.00
ATI Graphics Solution— Mono, Herc. Color Emulation on Mono CGA	(List 299) 125.00
ATI Wonder Auto Switch Mono, Herc CGA, EGA, VGA Any monitor, Any software, Auto conversion	(List 499) 299.00
EGA, CGA, VGA (640x480)	185.00

EXPANSION CARDS

Clock Card	25.00
Dual Floppy Disk Controller	25.00
Joystick	25.00
Gravis Analog Joystick	49.95
Game Port	19.00
Multi-Function, 1 ser/par/clk/game/ 2 floppy	61.00
Parallel (printer)	19.00
Dual Serial Port Card — 1 installed switchable Com 1, 2, 3, or 4	29.00
Kit for 2nd Port	20.00
640K RAM (ØK installed)	35.00

Prices are subject to change without notice.
Shipping CHARGES will be added.

MOTHERBOARDS

XT/Turbo 4.77/10mhz	109.00
AT 6/10 mhz Choice of Award, Phoenix or DTK Bios	350.00
XT/Turbo 4.77/8 mhz	99.00
80386 8/16 mhz/Award Bios & 1MB RAM, made in U.S.A.	1595.00
For XT/AT memory	\$Call

FLOPPY DISK DRIVES

Fujitsu 360K	89.00
Toshiba 360K	99.00
Teac 1.2 MB	125.00
Toshiba 3½" Drive Kit 720K	125.00
Toshiba 3½" Drive 1.44mb	145.00



— Pictured keyboard is 5339 —

KIT OPTIONS

*MS DOS 3.21 or 3.3 w/ GW Basic	95.00
*5339 Keyboard Sub	24.00
*Color Options: (Includes video card & monitor)	
CGA Color	200.00
CGA/EGA Color	410.00
CGA/EGA/VGA Color	590.00

ASSEMBLY AND TESTING

XT Systems	60.00
AT/80386 Systems	80.00

XT KIT W/ 2 Floppy Drives.

Includes: ØK RAM, Serial, parallel and game ports, clock/calendar, AT-Style keyboard, cabinet, power supply, mono graphics card and amber or green monitor. Keyboard switchable turbo.
8 mhz with lock, LED, Reset
& Turboswitch

579.00

10mhz with lock, LED, Reset
& Turboswitch

599.00

XT KIT W/20MB Hard Drive.

Includes: ØK RAM, Serial, parallel and game ports, clock/calendar, AT-Style keyboard, cabinet, power supply, mono graphics card and amber or green monitor. Keyboard switchable turbo.
8 mhz with lock, LED, Reset
& Turboswitch

879.00*

10mhz with lock, LED, Reset
& Turboswitch

899.00*

*(For 30MB Miniscribe add \$50.00)

80386 KIT —

Includes: 8/16 mhz, 1MB RAM, 1 360K floppy drive, 1-1.2 MB FD, 1 40MB HD, Award bios, switchable keyboard, monochrome monitor, mono-graphics. Serial/parallel ports, case, power supply, game port, clock/calendar. Main board made in U.S.A.

2675.00

80286 - AT KIT

Includes: ØK RAM, 1.2 MB FD, 1 360K floppy drive and 40 MB Seagate St 251 hard drive, 6/10mhz, serial, parallel and game ports, clock/calendar, AT-style keyboard, cabinet, power supply, monographics card, amber or green monitor, keyboard switchable turbo.

1495.00

KEYBOARDS

5339 Professional XT-AT w/12 function key	69.00
5060 Keyboard AT Style	55.00
KB101 Keytronic	67.00

Free Instructions with Each System

HARD DRIVES & CONTROLLERS

AT 40 MB Seagate #251-1	445.00
AT ST 4053 HD	545.00
AT (MFM) Hard Drive & floppy controller (WD)	140.00
AT RLL HD & FD controller	189.00
20 MB Miniscribe HD with controller	349.00
30 MB Miniscribe HD with controller	369.00

SOFTWARE

The Twin Spreadsheet	49.00
Leading Edge Word Processor	49.00
Ventura Desktop Publisher by Xerox	525.00
Learning Dos-MicroSoft	45.00

ACCESSORIES

1200 Baud Modem — Internal (Leading Edge Model L) Hayes compatible	99.00
2400 Baud Modem — Internal (Leading Edge Model L) Hayes compatible	219.00
1200 Baud Modem — External Hayes compatible	119.00
V20-8mhz	14.00
Memory Chips	(call for prices)

DEPEND ON MICROSPHERE

The components and products we sell are chosen specifically because they have been proven in our own use and testing. We guarantee our cards will be compatible when purchased all together.

NEW REVISED!

BUILDING YOUR OWN CLONE

****FREE BOOKLET****
*90-day warranty/30-day money back
(subject to restrictions)

MicroSphere
COMPUTERS

MicroSphere, Inc.
P.O. Box 1221
Bend, Oregon 97709
(503) 388-1194
Hours: Monday-Friday
9:00-5:30



MICRO CORNUCOPIA

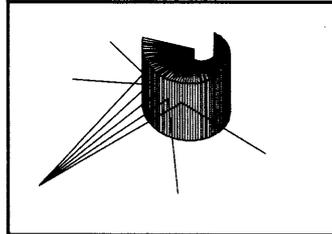
JULY/AUGUST 1988 - ISSUE NO. 42

FEATURES

8 David Thompson
Keeping Your Hard Drives Running
Think you're taking good care of your hard drives? Really? You're covered for every possible problem. Right? Ok, go ahead and ignore this article. Make my day.

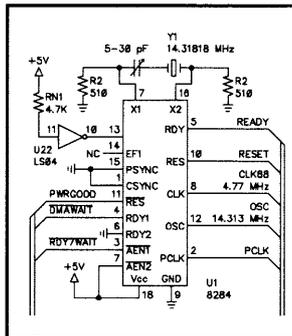
14 Dan Evans & Don Doerr
Troubleshooting PCs and XT's
Chip-level troubleshooting doesn't have to take days. Ten minutes is their max.

18 Earl Hinrichs
Ray Tracing on the TI34010 (Part 2)
If you're interested in graphics or just puzzled about solving quadratic equations in assembly language, then this is your dimension.



24 Larry Fogg
XT Theory Of Operation
Larry thought he was done after he finished the last of the XT's smart chips. However, you don't have a computer without pasting a few of these guys together.

32 Bruce Eckel
Simulating A Bus
If you're really interested in getting your computer off its digital jag and into the analog world then this bus based A/D project is a great start.



40 David Thompson
The West Coast Computer Swap Meet
After missing the Faire for six years I was very surprised.

42 Sam Azer
Working With PCX Files
This is a sneaky article. To understand graphics file formats you need to understand how graphics are displayed, especially color graphics. Sam explains both.

COLUMNS

50 C'ing Clearly
Scott Ladd looks at two exciting new C packages, one from Zortech, the other from WATCOM.

56 86 World
Laine leaves the U.S., his 286 system under his seat and a giant screen on his mind. (66 lines anyone?)

64 ShareWare

67 On Your Own

70 Pascal Column

75 Culture Corner

86 Technical Tips

CP/M CORNER

83 CP/M Notes

84 Kaypro Column

FUTURE TENSE

72 Tidbits

96 Last Page

Cover illustration by Paul Leatherwood

Editor and Publisher
David J. Thompson

Associate Editors
Gary Entsminger
Cary Gatton

Technical Department
Larry Fogg

**Director of Advertising
& Distribution**
Laura Logan

Accounting
Sandy Thompson

Order Department
Tammy Westfall

Graphic Design
Carol Steffy

MICRO CORNUCOPIA (ISSN 0747-587X) is published bi-monthly for \$18 per year by Micro Cornucopia, Inc. 155 NW Hawthorne, Bend, OR 97701. Second-class postage paid at Bend, OR and additional mailing offices. POSTMASTER: Send address changes to MICRO CORNUCOPIA, PO Box 223, Bend, OR 97709.

SUBSCRIPTION RATES:

1 yr. (6 issues)	\$18.00
2 yr. (12 issues)	\$34.00
3 yr. (18 issues)	\$48.00
1 yr. Canada & Mexico	\$26.00
1 yr. Other foreign (surface)	\$36.00
1 yr. Foreign (airmail)	\$50.00

Make all orders payable in U.S. funds on a U.S. bank, please.

CHANGE OF ADDRESS: Please send your old label and new address.

MICRO CORNUCOPIA
P.O. Box 223
Bend, Oregon 97709

CUSTOMER SERVICE: For orders and subscription problems call 503-382-8048, 9 am to 5 pm, Pacific time, M-F.

For technical help call 503-382-8048, 9 am to noon Pacific time, M-F.

RBBS - 24 hrs. 300-1200-2400 baud
8Bits, No Parity, 1 Stop Bit
503-382-7643

Copyright 1988 by Micro Cornucopia, Inc.
All rights reserved



By David Thompson

The West Coast Computer SOG

Jim Warren, founder of the West Coast Computer Faire (and wearer of the skates) will be joining us in Bend this summer for SOG VII. You too can join him for rafting, desktop publishing info, sessions on how-to service drives and systems, AI, C++, SCSI, the latest from Intel, compiler writers, a technical book author, and (of course) much, much, more. Sign up for the Saturday evening banquet and you'll hear Jim ask the burning question: "When are we going to do something useful with microcomputers?"

Don't wait to send in your registration. We're looking forward to seeing you in Bend July 14-16 for what's going to be the biggest, most informative, most laid-back Semi-Official Get-together (SOG) ever. (Held again at the Central Oregon Community College campus.)

Last year's attendees told us they'd gotten ten times as much out of SOG as they got out of those \$300+ conferences their employers were sending them to. At SOG they not only learned much more, they (and their families) enjoyed it immensely.

Added Attractions

During SOG, Bend will be hosting the Cascade Cycling Classic. The Classic has been on the bicycling tour for many years and attracts the top professional teams. On Sunday, July 17, they'll be winding up the competition with a criterium in downtown Bend.

In addition to the competition, there'll be brunches, beer gardens, concessions, the works. It's a great way to relax after all the mental strain of white water rafting and all-night (technical) discussions.

Tell your boss that SOG is not only a great source of information, but it's also an incredible opportunity to add really sharp technical people to your personal network. Networking made the original West Coast Computer Faire great.

Tell your family that Bend is one of the most beautiful communities in the Northwest. Over 250,000 skiers come here during Mt. Bachelor's seven to nine months of snow. (The U.S. Olympic alpine and nordic teams stay on the COCC campus while they train here.)

But summer is our biggest season. We get water skiers, sailboarders, backpackers, fishermen, horseback riders, bird watchers, hang gliders, astronomers, geologists, rock climbers (it's a world famous rock climbing area), and rock collectors.

(Continued on page 79)

Now, just \$129 and supports
Microsoft C 5.0 and QuickC

Blaise puts the Accent on C with C TOOLS PLUS/5.0™

BLAISE
N U

Enhance your Microsoft C programming environment with C TOOLS PLUS/5.0™ — a new, quintessential library of C functions. C TOOLS PLUS/5.0 from Blaise Computing Inc. puts a prime accent on quickly building professional applications using the full power of Microsoft C Version 5.0 and QuickC. Now you can concentrate on program creativity by having full control over DOS, menus, interrupt service routines, memory resident programs, printer and keyboard control, and more!

C TOOLS PLUS/5.0 prebuilt libraries are ready to use with either QuickC or the Microsoft C Version 5.0 command line environment. Complete documented source code is included so that you can study and adapt it to your specific needs. Blaise Computing's attention to detail, like the use of full function prototyping, cleanly organized header files, and a comprehensive, fully-indexed manual, makes C TOOLS PLUS/5.0 the choice for experienced developers as well as newcomers to C.

Continuous refinement of Blaise Computing's library products has produced a collection of tools that are unsurpassed for reliability, functionality and ease of use. Built upon the widely acclaimed C TOOLS PLUS, C TOOLS PLUS/5.0 includes such highly-developed features as:

- Accept user input, formatted output.
- "print" window-oriented output. **NEW!**
- ◆ **INTERRUPT SERVICE ROUTINES**
 - Capture DOS critical errors and keystrokes.
 - Install hardware interrupt handlers.
- ◆ **RESIDENT SOFTWARE SUPPORT**
 - Install, detect and remove memory resident programs.
- ◆ **MENUS**
 - Horizontal and pulldown. **NEW!**
 - Lotus-style support. **NEW!**

- ◆ **INTERVENTION CODE**
 - Schedule C functions at specified times, intervals or with a "hot key." **NEW!**
 - Take full advantage of DOS, even from memory resident programs. **NEW!**
- ◆ **FAST DIRECT VIDEO ACCESS**
 - All monitors, even EGA 43-line mode.
- ◆ **PRINTER CONTROL**
 - Access BIOS print functions. **NEW!**
 - Control the DOS PRINT utility. **NEW!**
- ◆ **UTILITIES AND MACROS**
 - Take advantage of DOS file structure.
 - Manipulate data types, far & near pointers. **NEW!**
 - Access any memory areas with fast "peek" and "poke" macros. **NEW!**

C TOOLS PLUS/5.0 supports the Microsoft C Version 5.0 and QuickC compilers, requires DOS 2.00 or later and is just **\$129.00**.

C ASYNCH MANAGER™ Version 2.0 IMPROVED!

C ASYNCH MANAGER is a library of functions designed to help you incorporate asynchronous communication capabilities into your application programs. Version 2.0 has been rewritten especially for Microsoft C Version 5.0 and Borland's Turbo C. Simultaneous buffered input and output to both COM ports at speeds up to 9600 baud, XON/XOFF protocol, modem control and XMODEM file transfer are among the many features supported and is priced at just **\$175.00**.

Blaise Computing Inc. has a full line of support products for both Pascal and C. Call today for your free information packet.

Turbo C TOOLS \$129.00
NEW! Windows; ISRs; intervention code; screen handling and EGA 43-line text mode support; direct screen access; DOS file handling and more. For Turbo C.

Turbo POWER SCREEN \$129.00
NEW! General screen management; paint screens; block mode data entry or field-by-field control with instant screen access. Now for Turbo Pascal 4.0, soon for C and BASIC.

Turbo POWER TOOLS PLUS \$129.00
NEW VERSION! Now supports Turbo Pascal 4.0. Screen, window, and menu management including EGA support; DOS memory control; ISRs; scheduled intervention code; and much more.

Turbo ASYNCH PLUS \$129.00
NEW VERSION! Now supports Turbo Pascal 4.0. Interrupt driven support for the COM ports. I/O buffers up to 64K; XON/XOFF; up to 19.2K baud; modem control and XMODEM file transfer.

PASCAL TOOLS/TOOLS 2 \$175.00
Expanded string and screen handling; graphics routines; memory management; general program control; DOS file support and more. For MS-Pascal.

ASYNCH MANAGER \$175.00
Full featured interrupt driven support for the COM ports. I/O buffers up to 64K; XON/XOFF; up to 9600 baud; modem control and XMODEM. For MS-Pascal.

KeyPilot \$49.95
"Super-batch" program. Create batch files which can invoke programs and provide input to them; run any program unattended; create demonstration programs; analyze keyboard usage.

EXEC \$95.00
NEW VERSION! Program chaining executive. Chain one program from another in different languages; specify common data areas; less than 2K of overhead.

RUNOFF \$49.95
Text formatter for all programmers; flexible printer control; user-defined variables; index generation; general macro facility. Crafted in Turbo Pascal.

**TO ORDER CALL TOLL FREE
800-333-8087**

TELEX NUMBER - 338139

Yes! Send me the prime accent!
Enclosed is \$_____ for _____ copies of

Please send me more information on your products.
CA residents add Sales Tax. Domestic orders add \$4.00 for UPS shipping, \$10.00 for Federal Express standard air.

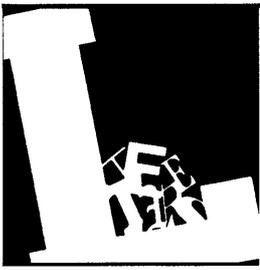
Name: _____ Phone: (____) _____
Address: _____ State: _____ Zip: _____
City: _____ Exp. Date: _____

VISA or MC#:

Microsoft is a registered trademark of Microsoft Corporation. QuickC is a trademark of Microsoft Corporation. Turbo C is a registered trademark of Borland International.

BLAISE COMPUTING INC.

2560 Ninth Street, Suite 316 Berkeley, CA 94710 (415) 540-5441
Reader Service Number 5



Letters

A Pat On The Back

Well done on the new look of the cover, although there was something nice about the "magazine-in-the-brown-paper-wrapping" look of old. A person needed to appreciate the merit of the product to see past the veneer.

Anyone subscribing to such a non-glossy magazine was demonstrating intelligence and uncommon good sense. To be a subscriber was to be one of the few. Now, alas, the front cover looks as good as its contents and thus will attract those shallow sorts who join the ranks of the intelligentsia just to look good. You know the people I mean — yuppies, computer store sales staff, and other pretenders.

But I suppose the bottom line had to be acknowledged and viability presumably depends on the survival of the prettiest. Or, as Joan Collins would say, "A stitch in time..." But I am so glad that the contents have remained untouched and unsullied.

What I find of more merit than the rest is the editorial. It is that honest, down to earth style of writing that appeals to my thinking. I would liken it to the style used by the host of the "Prairie Home Companion" radio show. A marvellous show which, to me, has as its strength a five minute story which espouses a philosophical tenet sometimes, but with subtle, quiet humour always. Keep up the good work.

You write of PC Network having major problems. I need to write in their defense for what appear to be genuine attempts to make good their debts. In September, 1987, I ordered two products, MASM and Turbo Pascal. One month later I phoned to be informed that neither was currently available. I cancelled MASM but requested that the TP3 order be continued. About that time they billed my credit card for

the TP3.

In mid-December I left the UK for Australia. The three month surface mail delivery time to forward mail from the UK to Australia meant that I had no idea if the product was in the mail or not. I had kissed that money goodbye.

So it was a very pleasant surprise when I noted in March this year a credit into my credit card account from PC Network. So I suppose PC Network is doing the correct thing and there might be hope for other creditors.

One advertiser in *Micro C* which I must comment upon is MicroSphere. I can only praise their service and friendly help. I purchased an XT clone in pieces and found the assistance in processing the order to be above average. A good company to deal with. And Cindy is nice to talk to.

I asked in my last letter to you (about two years ago) whether you really look like the "Around The Bend" column characature; I didn't get an answer. Whatever, please bring back the full-frontal. The dismembered head looks a little lost. What is his history?

Bevin J. Pettitt
Officers Mess
RAAF Glenbrook
NSW 2773
Australia

Editor's note: Everyone in the office said I had to write a great "Editor's note" for your letter. First, we're thinking of bringing back the brown paper wrapper. It will be a jacket that'll slip over the fancy new covers so you can fool those impressionable people.

Second, if you think the contents are "untouched and unsullied," you should hang around here at final edit time. The articles get touched, the columns get touched, and the editors get really touched. As for sullied...

Third, thanks for the comments on MicroSphere. I know the president of MicroSphere and he's really not too bad (as brothers go). The MicroSphere staff knows that when I show up I'm probably looking for Techtips, a weird part, or lunch.

As for the funny looking guy, he originally appeared in Punch many many years ago. Sandy found him in a clip-art book and we've both really enjoyed his company. We've had three illustrators try to modify him (sitting, lying down, etc). Only one was able to even approximate the original feel of the character.

C Defended

In his article "C vs. Assembly Language" (*Micro C* issue #40), Eric Isaacson raised some interesting issues. But his article reminded me of the sort of one-handed arguing one hears from a politician trying to gather votes. It was by no means a balanced discussion of the merits of C and Assembly languages.

Perhaps this is because Mr. Isaacson is somewhat misinformed about C and its capabilities. But he does have an excellent point when he states, "All those easily-accessed library functions can be very seductive; the programmer stops bothering to find the most efficient way to perform a task."

He then demonstrates this point with several lines of poorly written C. He went to great lengths to attempt to prove that C is inherently inefficient. What he did prove was that bad C compiles into bad assembler. Eric, you can't make chicken soup from chicken drop-pings.

In his example, a careful use of the `sizeof()` operator eliminates the need for the `strlen()` function that Eric thought was necessary. For the one line of code that Eric's example focused on, the 153

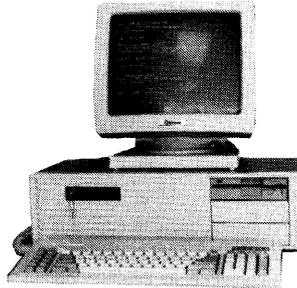
(Continued on page 77)

Stuff Dreams are made of...

The Dream-286

AT 6-10 MHz 0 wait state motherboard
 1 Megabyte of on board RAM
 1 5.25" TEAC 1.2M floppy drive
 40 Mb Seagate ST-251 hard drive
 Hard/floppy disk controller card
 12" amber Samsung monitor (tilt/swivel)
 Hercules compatible mono graphics card
 AT style keyboard
 AT case (UL and FCC approved)

Complete! \$ 1495



The Dream-386

Quality 80386 based motherboard
 1 Megabyte 80 ns on board RAM
 2 Parallel ports and 2 serial ports
 640 x 480 on board EGA/VGA card
 1.2 Megabyte 5.25" floppy drive
 Western Digital WA-2 hard/floppy controller card
 Professional enhanced (101 key) keyboard
 Case (UL and FCC approved; reset switch, power and turbo LEDs, keyboard lock)

Complete! \$ 2295

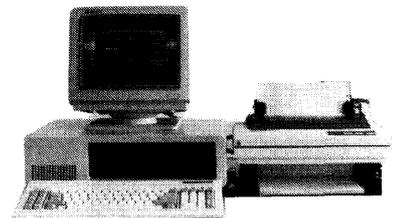
Novell Networks

Taylored to your exact needs. Call for more information.

The Dream-88

XT 10 MHz Turbo motherboard
 640 K of on board RAM
 1 5.25" 360K Fujitsu floppy drive
 30 Mb Seagate ST-238 Hard drive
 Floppy controller card (controls 2)
 Hard disk controller card
 12" amber Samsung monitor (tilt/swivel)
 Hercules compatible mono graphics card
 AT style keyboard
 XT slide case (UL and FCC approved)

Complete! \$ 895



Hard Drives

Seagate 20 MB ST-225 \$ 269
 Seagate 30 MB ST-238 \$ 299
 Seagate 40 MB ST-251 \$ 395
 Micropolis 71MB Hard Disk .. \$ 695
 Seagate 80 MB ST-4096 \$ 750

Floppy Drives

Fujitsu 360K 5.25" \$ 79
 Fujitsu 1.2 MB 5.25" \$ 99
 Fujitsu 720K 3.50" \$ 115
 Fujitsu 1.44 MB 3.50" \$ 145

Floppy drives come with mounts

Citizen Printers

180-D (180 cps 9 pin) \$ 189
 MSP-40 (260 cps 9 pin) \$ 339
 MSP-45 (wide carriage) \$ 489
 Tribute 124 (200 cps 24 pin) ... \$ 545
 Tribute 224 (wide carriage) ... \$ 689
 Premiere 35 (daisy wheel) \$ 499

Communications

1200 Baud internal \$ 95
 2400 Baud internal \$ 185
 1200 Baud external (w/ cable) \$ 99
 2400 Baud external (w/ cable) \$ 235
 PC FAX (card & software) \$ 415

Accessories

PC Mouse w/ paint \$ 115
 M-8 Logitech mouse \$ 99
 Kraft IBM/Apple Joystick \$ 25

Laptops

Sharp 4051 \$ 795
 NEC EL \$ 1685
 NEC EL HD \$ 2495

Software

Borland Turbo C \$ 79
 Borland Turbo Pascal \$ 69
 Borland Turbo Basic \$ 69
 Borland Quattro \$ 139
 Borland Sidekick \$ 49
 Microsoft Works \$ 149
 Microsoft Windows v2.03 \$ 69
 Microsoft DOS 3.21 \$ 69
 Microsoft DOS 3.30 \$ 99
 Norton Utilities (Advanced) . \$ 89
 Norton Commander \$ 45
 Stella Business Graphics II .. \$ 99
 Peachtree Accounting System \$ 179

DreamTech

5175 Moorpark Avenue
 San Jose, CA 95129
 Open Monday - Saturday
 10 AM to 7 PM
 Phone: (408) 996 - 2373



All systems carry a full 1 year warranty (original system configuration only). Prices are subject to change without notice. All orders are shipped UPS FOB San Jose unless otherwise specified at time of order. California residents add 7% sales tax. All registered trademarks are recognized. Dream-88, Dream-286, and Dream-386 are trademarks of DreamTech Computers.

Reader Service Number 16



Keeping Your Hard Drives Running

A couple of months ago I spoke to the Eugene PC group. My topic (officially) was desktop publishing but audience questions quickly turned the subject to hard drives.

Whenever the subject comes up in Micro C we get letters, lots of letters. Bring up the subject in a computer store and you'll be inviting fisticuffs. (Sound like a hot topic?)

You and I both know that RAM data is volatile. Turn off the machine and the data goes away. (Sometimes it goes away before you turn the machine off, but that's a different story.) Anyway, because of RAM's memory problem, we use floppy and hard disks for our permanent data storage.

Unfortunately, data stored on disk isn't necessarily permanent. After all,

floppy drives aren't exactly carving those bits onto rock. And hard drives? They write by long distance — their heads floating on air just above the disk surface. (At least, that's what they're supposed to be doing.)

I got out the old Tektronix 547 oscilloscope and looked at the data coming from the heads on a standard floppy drive. You talk about gibberish, this stuff was little better than static. (The fact that the drive was reading CP/M files had nothing to do with it. Really, it didn't.)

Anyway, if you saw the kind of junk a drive gets when reading a perfectly good disk, you'd realize just how easily a drive can get confused when things deteriorate.

A Magnetic History

Four score and 2,000 eons ago, when man was first discovering tools (a large club, Borland's Graphics Toolbox...), he kept data on magnetic tape, giant spools of magnetic tape. Unfortunately magnetic tape was susceptible to magnetic fields (surprise), and those magnetic fields included the information written onto the magnetic tape (even bigger surprise).

So, even though the tape was carefully stored in a dust free, humidity controlled, flux free, tiger free, everything but rent-free cave, data written on part of the tape would migrate to the tape surfaces pressed tightly above and below it. (Which explains why we have so little data from that period.)

Sometime after the Pleistocene era, scientists discovered they could stop the migration (of data) by writing the stuff onto disks. (Later, native worshippers of high-tech dumpboxes were credited with inventing the wheel.) Anyway, floppies were the perfect solution to data migration (at least until man invented the double-sided floppy).

Eight out of ten times, if we get a 225, it has the same problem. It's usually the stepper driver transistor

More Recently

So we arrive at the 20th century. Oh, we're still losing data, but that's okay, we've got plenty. And, backing up data just makes more of it to lose (which is no doubt why many people studiously avoid backing up). So I'll do my best to help you protect your data without overwhelming you with the "B" (Backup) word.

First, Backup Your Data

Backup your data but don't use BACKUP. Folks I've talked to recommend using Fastback from Fifth Generation Systems or Corefast from Core International. I've found that Backit is also reliable (though its menus are a bit overwhelming and obscure).

Second, The Drive/Controller

If you're going to be slovenly about backing up, you may as well get the most reliable hard drive and controller you can.

The more I investigate the drive marketplace, the more I'm convinced that hard drives are very, very good. With a few exceptions, it's hard to go wrong. Unfortunately, there's little agreement about which drives are the current exceptions.

The problem is change. Ask a manufacturer about life expectancy and he'll tell you 20,000 hours. And he's right. That's what he expects. He won't know whether a particular drive will make it until it makes it. If the batch averages 14 months before failure then it'll take him 14 months to find that out.

So, we're using history to predict the future, but because of rapid change, history isn't a very good predictor. Anyway, with that in mind, here goes.

Work In Progress

I called Seagate and talked to the guy who designed the 225. He told me they'd tried a number of fixes for the track-1 problem (munched guard band) before finding a solution. And, he said they were constantly working on increasing the life (and reducing infant mortality) of their units. (I'm not sure he could have said anything else, but he was convincing.)

He wouldn't say specifically what they were doing, but he promised to have a marketing person call me (one who could say, specifically). So far, no one has called.

I arranged to talk to the folks at Rotating Memory Service during the West Coast Computer Faire. (I was there. They were there. We should talk. Right?)

If you were to buy a hard drive right now, what would you get?

"Control Data, we use them in all our own systems. But they start at 40 meg. For a smaller drive look at the PTI, Peripheral Technology from Southern California."

Are they cheap?

"Yes, very. They're very competitive with Miniscribe's 3 1/2" and I think that's what they're trying to hit. We also found the Miniscribe to be very good. Miniscribe has the best technical support."

What about 3 1/2" drives in general?

"3 1/2" drives are like the others — they read, write, seek and spin and they have their share of problems. The 8425 Miniscribe is the leader in the 3 1/2s. Rodime also has a good 3 1/2."

If I Can Butt In A Minute:

I should mention here that I've also heard good things about the Miniscribe 5 1/4" half height 40 meg, and I've been hearing more and more positive reports about the latest Seagate 251s. The 251 is faster and cheaper than the Miniscribe, and its infant failures (at least) seem to have disappeared.

But, a Seagate technician I talked to in early April mentioned a call he'd just received from a customer. All 15 of their 251s had failed.

"The guy was crying on the phone and there wasn't a thing I could do for him. I really felt awful."

In late March I came across one of the latest full-height Seagates (4038, 4051...). It had a new non-kachunking park mechanism. (Listen to the drive as power is removed, if there's a loud kachunk it's an old one.) Hopefully the park fix will make those voice-coil monsters really solid.

I've had numerous repair people tell me that, single-handedly, the Seagate 238 (30 meg) has given RLL a bad name. A few are still insisting that because controller manufacturers are still tweaking the technology, RLL may not be ready for prime time. Of course they also couldn't agree whether the drives supporting RLL needed to have plated media or if oxide was sufficient. (These were heated discussions.)

Note: *The only thing that creates a readable signal pulse on magnetic media is a current reversal in the write-head. That reversal shows up as a spike during a read. The quality of the media and the head determine how close together those reversals can be and still be read, later, as distinct pulses. (Also, the knowing distance between pulses is very important to recreating the data.)*

MFM (Modified Frequency Modulation) is the standard encoding method for floppies and hard drives. It manages to encode data and clock information and put it on the disk with about half the reversals (for the same bit rate) as FM or the other early schemes.

RLL (Run Length Limited) is a newer encoding scheme which requires about 25% fewer reversals. Where MFM requires at least one transition (for clocking purposes) every X bits, RLL requires a transition every X+Y bits. As I

Native worshippers of high-tech dumpboxes were credited with inventing the wheel



understand it, to get the 50% greater bit density, RLL needs higher reversal density (thus the suggestions about plated media) and more precise positioning of those reversals. (Perhaps someone would like to do a quick piece on RLL. It's becoming the new standard, problems or not.)

Meanwhile Back At The Interview

What can I do to keep my drive from failing?

"Use an autopark program. Leave your system on. Use a program that parks the heads after 30 seconds. You can have a power-line spike that comes through causing the head to write over

a FAT file. Does a spiral write?"

Spiral write?

"Drives would sometimes start writing (wherever the head was) during a brown out. Now there's circuitry to prevent that, but you should still park the heads."

The spindle motors aren't a problem? Heat's not a problem when you let them run forever?

"You have most of the problems when you turn the system on and off. Of course, if you leave it running, you have to watch the vents so the system can cool and you have to dust it out, so it can breath.

"Keep your computer in a clean environment, otherwise the filter on the hard drive will clog, especially if you have a lot of smokers. Smoking is really bad around computers."

What else contributes to drive problems?

"It's not what people are doing, it's poorly manufactured drives. We see very few drives that are actually blown up. Most problems we see are with the 225 — the stepper circuitry.

"You start worrying about your Seagates when you hear the clicking noise when the head steps out. On boot, the head steps in increments of eight tracks watching for the data burst on track -1. If it doesn't find it, it keeps going and burns up the stepper transistor.

"Eight out of ten times, if we get a 225, it has the same problem. It's usually the stepper driver transistor that goes. You'll see a little spot of brown on the circuit board (near the front of the drive). On the other side of the board is the driver transistor. We just saved your readers a bunch of money by telling you this."

Thanks. I've been hearing grumbles about stepper shaft wobble.

"Wobble gives you soft errors. Actually, the bearings are pretty good on the stepper. I very seldom see soft errors on the 225s, they are really pretty good. Mostly it was the way the stepper circuitry was designed in the beginning without a track 0 sensor."

What about spindle noise?

"If any drive is getting noisy you should check the ground strap (the thin metal tongue that presses against the end of the spindle bearing). It can get bent or lose its tension. Just take it off, bend it so it'll be tighter against the carbon button and reinstall it. It's just a static ground but people think their bearings are gone when it gets loose.

"Another problem people call us

about is sector errors. When I worked at Seagate, I didn't see a single drive that had a head crash or other defect. But I see them now.

"That's because the problems show up after they're out of warranty. We get them at between 13 and 18 months. That's the life expectancy. They do have a good warranty, it's a replacement warranty."

"When people call in about increasing sector errors, I usually tell them to back up what they can and reformat. Backup once a week and keep two copies, one off premises. If you forgot to do a backup, then call us, we do data recovery."

On what percentage of the drives you receive can you recover the data?

"We're about 85% successful. Of the 85% of the drives, we'll retrieve an average of 90% of the data. On the others, the FAT files have been totally obliterated."

"We just spent 36 hours on a hard drive from a cardiovascular unit. We got almost all their data. Not all of it, but almost."

Controllers, what do you like or not like?

"Western Digital is far better, has an

excellent controller for the AT. The 1002 27X is the one I'm liking more and more for the XT."

The 27X is an RLL controller. If you were going to run RLL, which drive would you use?

"Miniscribe 3425; it's a 5 1/4" half height. Rodime has a 202E which is good, too."

But I'm hearing about problems with RLL.

"What you're looking for on RLL is a zero-defect error map. You don't want to see any consistent errors like head 3 cylinder 193... something like that."

Let's say you get a standard MFM drive with no defects. Would that work?

"Chances are it would. For example, the Rodime RO204 is a 20 meg MFM, but I'm willing to bet that it would format fine RLL (as 30 meg)."

Can people predict if they're going to have trouble with RLL?

"The RLL will work for a week or two and then all of a sudden they'll start getting soft errors."

In other words they'll get random "track not found" or "sector not found" messages.

"Then it'll just disintegrate. They'll have to reformat."

Protection From The Data Munchers

Let's say you have a competent hard drive and an aversion to backing things up. What can you do to protect yourself from Catastrophic Hard-drive Agony and Remorse (CHAR)?

Nothing.

However, you can do something to prevent or recover from the very common lesser problems. These problems often show up as "Sector Not Found" right at the beginning or end of the day. (There's a reason for the timing.)

They also show up as missing files (something garbaged the directory) or missing subdirectories.

Stopping The Slow Deterioration

As a drive ages (hey, we all age), the casting destresses itself, the bearings wear a bit, and the head assembly shifts. Add to this a bit of heat (expansion) and you'll see that when the heads are over track 427, they're not necessarily precisely over the same track 427 they were over when the drive was last formatted.

Some drives have a surface that's used only for verifying head position. That works very well, but it's not used

TRILOGY

A Powerful Procedural, Database, and Declarative Language.

SPEED — Where Prolog must backtrack, Trilogy can often solve the problem logically. Trilogy takes advantage of logic constraints (they constrain the search to possible solutions) which either eliminate backtracking or reduce millions of backtracks to a very few.

SYNTAX — Trilogy uses an intuitive, Pascal-like, program structure.

INTEGRATION — Trilogy is complete. It's the only language you need for writing Pascal-style routines, database handlers, and Prolog-style programs.

MODULARITY — Trilogy is modular language, very similar to Modula-2.

ENVIRONMENT — A complete programming environment, you get editor, library, linker, loader, error handling, automatic make, and contextual help. Plus, you get modules for: math, string handling, file manipulation, windows . . .

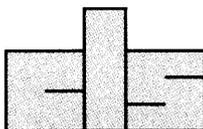
A TRUE COMPILER — Trilogy is an interactive compiler which produces native code for the 8086/8087.

LOGICAL PURITY — Trilogy was designed from scratch as a simple language with a completely logical foundation. Trilogy's speed results from its design, not from added commands. (Prolog's assert, cut, var, and retract, are not logical parts of that language. They were added to improve performance.)

DATABASE SUPPORT — Trilogy supports: variable size records, records with arbitrary values (lists, recursive trees); plus record insertion, deletion, and modification. (Anywhere in the file.) Files are relations and can be queried from within the language.

PRICE — Only \$99.95 postpaid, U.S. funds. Plus \$5.00 shipping & handling. Or \$12.00 shipping & handling outside North America. Check, money order or VISA accepted.

Order From:



COMPLETE LOGIC SYSTEMS

741 Blueridge Ave.
North Vancouver BC Canada V7R 2J5
(604) 986-3234

ONLY \$99⁹⁵

a lot because it wastes space. (These drives have an odd number of read/write heads.)

For the rest, it's guess and hope and careful allowances for the effects of heat and wear. Unfortunately, the allowances are never enough. (I've got kids, too.)

So, sector headers and trailers eventually get out of line from the data. You write headers only when you do a low level format. COM and EXE files are also susceptible since they're not updated very often.

Anyway, after six months or a year you'll probably start losing sectors. If you're lucky, the problem will warn you by showing up only when the drive is hot (or cold). The cure is to backup everything you hold dear (at least whatever's still readable) and reformat the drive.

That's a pain but there's a way to prevent the problem.

Disk Technician

Disk Technician goes through your drive, sector by sector, looking for bit errors. When it finds even a single erroneous bit, it reformats the whole track, then rewrites correct data onto the track. (It also keeps a record of the locations of the errors and watches them for future problems. A location which repeatedly generates soft errors will eventually get locked out.)

Note: Most hard drive controllers write an 11-bit error correction code onto the disk. That code makes it possible for the controller to correct an erroneous string of bits (I believe it's up to 64 bits long). It's only when the controller can't correct the data that it reports an error. By then, of course, things have really deteriorated.

The Disk Technician package includes a park routine which it installs in AUTOEXEC.BAT after the program locks other software out of the top cylinder. It does this so the heads won't park over important data.

Disk Technician is easy to run, generates a report of each check, and is quite useful. Its primary problem is it's copy protected. Once you've installed the package on a hard drive, that's it. It will only work with that winnie and only with the original distribution disk in a floppy drive. Do something that changes the drive (like their own low level format) and you have to call the factory to get the package reset. (They say you'll be able to reset your own disk soon...)

Meanwhile, this program still insists

you backup all your files before running it. So it's ironic that the only thing you can't backup is this program (at least not without a bit copier).

Recovering Garbaged Directories

While Disk Technician tries to prevent drive hardware from losing data, Mace Utilities is one of several packages designed to help you recover it. The recovery we're talking about is primarily from FAT and directory problems. But then FATs and directories seem to get the most abuse. (Have you run any fun trojans lately?)

If you get the Mace package before you have a problem, the routines will copy your FAT and directories onto the drive's innermost tracks. (Right where Disk Technician likes to lock out sectors. Fortunately they know how to get along.)

Assuming your directory structure gets eaten, Mace copies everything back. At least that's the theory.

After backing up my hard disk and removing the only copy-protected program, I ran RXBAK (which copies the FAT, boot, directory, and subdirectory data into the inner sectors). Then I tried deleting a large group of files, figuring that unformat would restore the directories to the point where I ran RXBAK.

Unfortunately, the unformatter (which does the restoration) got weird when it found an unformatted, locked-out sector. (The whole track was unformatted by the controller to keep MS-DOS from using it.)

There are some 40,000 sectors on a 20 meg drive. After finding the first unformatted sector, it began a cycle of rehomeing and retrying which slowed the sector reads down from blindingly fast to about 1 per minute. (And it continued this slow pace even after it had finished the locked-out track.) At about 1 minute per good sector, the final 38,000 sectors would take...

I reset the system and used their undelete utility to recover the files. However, I was bummed that the recovery portion didn't work better. (I tested the package on a floppy and it restored the directory just fine.)

H-TEST/H-FORMAT

I'm not finished before I tell you about the H-TEST/H-FORMAT package. This set includes the program HOPTIMUM and it reformats a single track with interleaves of 1 to 16 and tells you the data transfer rate for each

interleave. (Afterwards, the original data is written back onto the track.) Wow, one of my drives had an interleave of 3 (25Kbytes/sec). At 4, the program reported 80K bytes/sec and then the rate declined slowly after 4. So, I had the program reformat the drive at an interleave of 4.

HOPTIMUM rewrites the data onto each track after doing the low level format. As far as I can tell, the only way you're going to lose data with this package is to have power problems during the reformat.

These routines were written by Kolod Research and are marketed by Paul Mace. Every computer dealer, every clone builder, and every user group whose members care about system performance should have a copy of this package. (It's not copy protected.)

Finally

I don't have to tell you that hard drives are important. I don't have to tell you that data is important. But I may have to tell you that no matter what kind of hard drive you have, it's going to fail. The differences among drives are not whether they'll fail but when.

Rotating Memory Service

473 Sapena Court #26
Santa Clara, CA 95054
(408) 988-2335 (technical)

Fastback \$179

Fifth Generation Systems

1322 Bell Ave. Ste. 1A
Tustin, CA 92680
(800) 225-2775
in California (714) 259-0541

Corefast \$149

Core International
7171 N. Federal Hwy.
Boca Raton, FL 33431
(305) 997-6055

H-TEST/H-FORMAT \$89.95

Mace Utilities \$99.00
Paul Mace Software
400 Williamson Way
Ashland, OR 97520
800-523-0258



SOFTWARE ENGINEERING COMES OF AGE.

ANNOUNCING LOGITECH MODULA-2 VERSION 3.0

Modula-2 is the language of choice for modern software engineering, and LOGITECH Modula-2 is the most powerful implementation available for the PC. The right language and the right tools have come together in one superior product. Whether you're working on a small program or a complex project, with LOGITECH Modula-2 Version 3.0 you can write more reliable, maintainable, better documented code in a fraction of the time at a fraction of the cost.

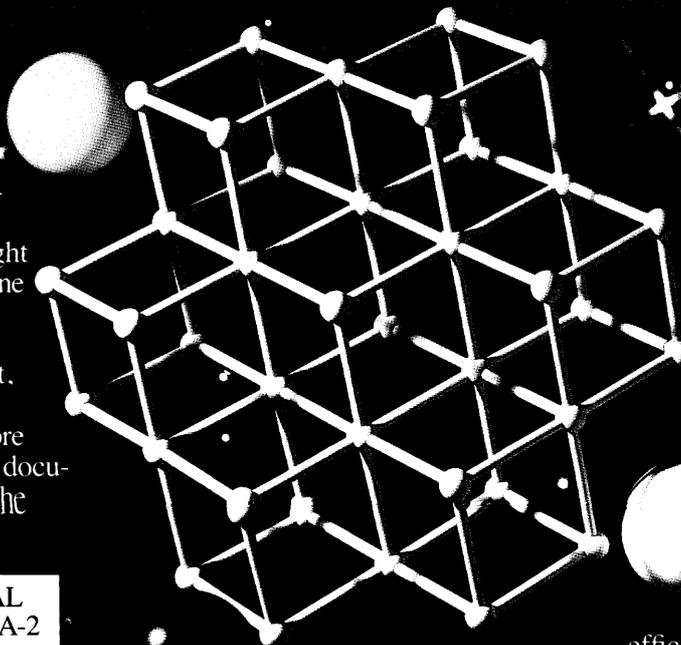
**FREE TURBO PASCAL
TO LOGITECH MODULA-2
TRANSLATOR**

NEW, IMPROVED DEBUGGERS

Time gained with a fast compiler can be lost at debug time without the right debugging tools. With the powerful Logitech Modula-2 Debuggers you can debug your code *fast*, and dramatically improve your overall project throughput. The Post Mortem Debugger analyzes the status of a program after it has terminated while the dynamic, Run Time Debugger monitors the execution of a program with user-defined breakpoints. With their new, mouse based, multiple-window user interface these powerful debugging tools are a pleasure to use.

NEW, INTELLIGENT LINKER

Links only those routines from a particular module that you need, so you eliminate unreferenced routines and produce smaller, more compact executable files.



NEW, IMPROVED COMPILER

Faster and more flexible. Now its DOS linker compatible object files (.OBJ) can be linked with existing libraries in C, PASCAL, FORTRAN and ASSEMBLER — so you can build on previous development and put the power of LOGITECH Modula-2 to work for you right now. Fully supports Wirth's latest language definition, including LONGINT and LONGSET, which provides large set support including SET, of CHAR. Provides optimization for tighter, more efficient code generation.

NEW EDITOR

Our new, mouse based editor is fully integrated, easy to learn, fast and easy to use, and very customizable. Its multiple, overlapping windows and color support make it easy to manage parts of one file or several files on the screen at one time. You'll love using it — with or without a mouse.

Call for information about our VAX/VMS version, Site License, University Discounts, Dealer & Distributor pricing.

To place an order call toll-free:

800-231-7717

In California:

800-552-8885

- LOGITECH Modula-2 V. 3.0 Compiler Pack **\$99**
Compiler in overlay and fully linked form, Linkable Library, Post Mortem Debugger, Point Editor
- LOGITECH Modula-2 V. 3.0 Toolkit **\$169**
Library sources, Linker, Run Time Debugger, MAKE, Decoder, Version, XRef, Formatter
- LOGITECH Modula-2 V. 3.0 Development System **\$249**
Compiler Pack plus Toolkit
- Turbo Pascal to Modula-2 Translator **FREE**
With Compiler Pack or Development System
- Window Package **\$49**
Build true windowing into your Modula-2 code.
- Upgrade Package
Call LOGITECH for information or to receive an order form.

Add \$6.50 for shipping and handling. California residents add applicable sales tax. Prices valid in U.S. only. Total Enclosed \$ _____

VISA MasterCard Check Enclosed

Card Number _____ Expiration Date _____

Signature _____

Name _____

Address _____

City _____ State _____

Zip _____ Phone _____

LOGITECH

LOGITECH, Inc.

6505 Kaiser Drive, Fremont, CA 94555

Tel: 415-795-8500

In Europe: LOGITECH, Switzerland

Tel: 41-21-87-9656 Telex 458 217 Tech Ch

In the United Kingdom: LOGITECH, U.K.

Tel: 44908-368071 Fax: 44908-71751

Troubleshooting PCs and XTs

Diagnosing The Frequent Failures

Dan Evans works for a technician training company and one of their services is supporting the technicians they've trained. If their emphasis on quick diagnosis surprises you, then you probably haven't worked in a shop. This is an introduction to how they work.

Back in early 1979, I escaped from the Iranian Revolution on a military C-130, returned to the States, and found myself in the PC Revolution. Two years in Iran had not prepared me for computer stores or the latest in beach fashions. But I pulled myself together and got a sales job with OB1, one of the original Apple Computer distributors. I was a somewhat technical salesperson, and over time I acquired a reputation for technical support.

This reputation was largely buoyed by an understanding of three of four failures very common to the Apple II, and a little common sense.

The most common failure, the tendency of the Apple to lose data because of dust or corrosion on the language card's edge connector, made me a hero with friends, neighbors, teachers and the Apple dealer. The fix: turn off the machine, take out the RAM card, clean the contacts with a pencil eraser, and put the card back.

The company Dan and I now work for, National Advancement Corporation (NAC), trains technicians in the fine art of maintaining IBM personal computers and compatibles.

One way to improve troubleshooting logic is with flow charts. We've spent thousands of man hours developing ours. But we've also found two other very effective methods. One we call "frequent failures" — watching for common symptoms to common problems. The other method requires a diagnostic ROM.

No matter which method you use: don't set out to do a component-level fix unless you're pretty sure you can find the problem quickly. Our rule is the "ten minute limit." If we don't think we can find a problem within ten minutes, we usually send it out.

There's no percentage in trying to work on an unfamiliar unit, or to identify an unusual problem. It's too easy to waste many hours.

Frequent Drive Failures

I wouldn't like to admit how many Seagate 225s, 238s, and 213s we sent out for repair before we discovered how to fix a frequent failure.

Symptom: Drive loses data and format (more noticeable with temperature changes).

Solution: The problem is caused by expansion of the chassis as the drive warms up. The circuit board mounted on the underside does not allow the case to expand normally as it heats up and the drive loses its alignment. Three screws hold the circuit board to the drive housing. Loosen the screw nearest the front of the drive.

Editor's note: You'll need to loosen all three screws if the holes on the board and the housing don't line up. Line up the front hole, then tighten down the back screws and leave the front screw just finger tight. (It's the way Seagate is currently shipping 225s.)

Then reformat the drive (low level, fdisk, and DOS format). Be sure you have an up-to-date backup of the drive before loosening anything.

IBM Monochrome Monitor (5151)

Symptom/Solution 1: A blank screen indicates a bad fuse, probably caused by a shorted transistor (location TR23; this can be a BU406, BU407 or BU408). Do not use a cross-referenced component.

To disassemble the monitor, disconnect the power and signal cables from the system. To remove the two screw

covers on top of the monitor, use a small flat blade screw driver and pry up the back of each cover. Next remove the two Phillips screws from the top of the monitor and turn it face down. Remove the six screws on the bottom of the case. Don't remove the feet or the two screws next to where the cables enter the monitor. Pull both cables from behind the plastic flange. You can now lift off the case.

If you still do not have a picture after replacing the fuse and the transistor, check for a glow in the back of the CRT (this will tell you if you have power). If there is no glow and the fuse is still good, the main power transformer probably has an open primary. Don't fool around, the transformer's primary is connected to 110 VAC.

If you have no picture but you do have a glow in the back of the CRT (you've replaced the fuse and transistor), quickly turn off the monitor. Check the heat sink at the transistor you just replaced (location TR23). Careful, it may be very hot. If it is getting hot, the flyback transformer is shorted and needs to be replaced.

Editor's note: Flyback transformers were invented in the Australian bush, so you have to be careful with them. Technicians have been injured trying to throw them away.

This part is available from Computer Parts Exchange (CPX), Chatsworth, California, phone (818) 709-4003. CPX also has a facility in New Jersey, phone (201) 389-8333. Another source for this part is Acetron Dataproducts, Van Nuys, California, phone (818) 786-9789.

Symptom/Solution 2: Full brightness unaffected by the brightness control. This indicates a bad resistor (location R515; its value is 180K ohms at 1/2 watt).

Symptom/Solution 3: The symptom of the third "frequent failure" on the 5151 is a horizontal line across the screen. This indicates a bad chip (location IC401; it is a TDA1170 IC).

Floppy Disk Drives

Symptom/Solution: Read/write failures are most often caused by the drive being out of alignment. It is not our intent to teach drive alignment in this article. That kind of instruction is better left for a hands-on situation.

Editor's note: Allyn Franklin has been teaching floppy alignment at SOG, he'll be back this year to talk about floppies and hard drives.

Technicians often tell us they throw away bad drives, so we've asked them to toss them our way.

We've found that about 35% work perfectly. About 50% need only three to five minutes of attention to clean up alignment. About 15% need a read/write head, motor or chip (usually a preamp or stepper driver IC).

Dead PC Or XT

Symptom/Solution: System appears dead. There is no display, no beep, and no error message when power is turned on. Replacing the power supply doesn't help. Replacing the system board fixes the problem, but what's wrong with the system board?

To understand this one, let's look at the power-on self test (POST). The IBM POST routine, located in the BIOS ROM, runs numerous tests of the system board and add-on boards. If any of the first 11 tests fails, you'll see no cursor and no error message. The system appears dead, even if the failure is simply a bad chip in the first 16K (critical bank) of RAM.

A bad RAM chip in the first 16K accounts for about 50% of all dead system boards.

In the case of the PC, some compatibles, and sometimes the XT, the critical bank of RAM is soldered in, making it impractical to troubleshoot by swapping chips.

So we use the WindsorPOST diagnostic ROM module, made by Windsor

I wouldn't like to admit how many Seagate 225s, 238s, and 213s we sent out for repair before we discovered how to fix a frequent failure.

Technologies of San Rafael, California, (415) 456-2200.

The WindsorPOST replaces the socketed BIOS ROM. It does 36 diagnostic tests on power up, and about 70% of the time it will pinpoint the problem in an apparently dead board.

When the system is turned on, WindsorPOST beeps to announce that it is working. A red LED indicates whether or not the ROM socket is getting 5 Volts. The module then initializes the video display and starts its tests. Because the tests are all run from the WindsorPOST ROM, not much of the system needs to be running.

The WindsorPOST tests include the 8088 Microprocessor, first 16K of RAM, 8237 DMA Controller, 8259 Interrupt Controller, 8253 Timer/Counter, 8255 PPI, Read and Display Switch settings, Cassette Relay (PC Only), speaker and related circuitry, the IBM-PC ROM, full system RAM (to 704K), plus video at-

tributes and RAM on the Monochrome or Color Monitor Adapter.

We received one of the first WindsorPOST units about a year ago and have used it to locate failures on system boards, video display adapters, and memory expansion units.

Our graduates tell us they can isolate about 70% of all system board failures to the chip with the WindsorPOST module. (All without the use of an oscilloscope or logic probe.)

Another fifteen percent (15%) of failures can be attributed to the clock circuit, Non-Maskable Interrupt (NMI) circuit, or a Data/Address Line failure. These problems can often be isolated with an oscilloscope via the following:

- 1) Jumper pin 10 to pin 20 on the DMA Controller (location U35 on PC, U28 on XT) and turn on the system. If this solves the problem, replace the DMA Controller.

- 2) Check for a 4.77 MHz clock on pin 8 of the clock chip (location U11 on PC, U1 on XT). If there's no clock, check the clock chip and the crystal.

- 3) Check for 4.77 MHz clock at pin 8 of the 74LS04 (location U51 on PC, U89 on XT). Replace the chip if there is no clock.

- 4) Check for 4.77 MHz clock on pin 4 of the time delay (location TD2 on the PC or XT). If no clock is present, remove the time delay and replace with a 470 Ohm resistor across pins 1 and 4 on the system board.

- 5) Check the signal at pin 9 of the 74LS74 (location U96 on PC, U86 on XT) for a HIGH or FLOATING signal. If a high or float is found on this pin, replace the chip. This signal is the ALLOW Non-Maskable Interrupt (NMI).

- 6) Replace the 8088 processor. It's socketed.

These steps should help you locate 85% of all system board failures within five minutes.

Most of the other 15% are caused by a stuck address or data line. In many cases, you can find these problems with an oscilloscope.

If you're not very experienced, we don't recommend that you try to remove or install soldered-in ICs.

Finally

I hope that this information will save you time and money. If you have a solution to a "frequent failure," or are looking for a solution not mentioned here, please contact:

Don Doerr
National Advancement Corp.
2730-J South Harbor Blvd.
Santa Ana, CA 92704
(714) 754-7110

Editor's Notes On WindsorPOST

I received the ROM from Windsor Technologies just a couple of days before the final, final, final deadline for this issue so I'm just beginning to check it out. However, I did hop over to MicroSphere to rummage through their boneyard. This is what happened:

The first system was a Kaypro XT

clone. It wouldn't display anything no matter what video board. I wasn't optimistic but I tried the ROM. Nothing on the screen, though I did get the introductory beep and two beeps later on.

The second system worked, but I stuck some suspect RAM into it to see how well Windsor isolated RAM problems. Windsor reported no errors so I bent a pin on one of the RAM chips. (It seems that every second or third system I work on, I accidentally bend a pin or a socket doesn't work right, so open pins are a pretty common problem.)

First I bent under pin 10 (address line 5), reset the system and guess what? No error. Now, there's a reason Windsor's missing the problem. The processor doesn't care where data winds up in RAM. It just issues an address and data. Later, when it reissues the address, it expects to read the same data. Well, a disconnected address pin won't change that. However, that missing pin will cause later data to be written over earlier data since only half the RAM chip is being used.

The way to detect this kind of error is to write all ones to the first half of memory and all zeros to the second half. Then go back and read the entire first

half, looking for zeros. The starting address of the zero and which bit contains the zero tells you which chip and which address pin are defective.

I fixed the address pin and disabled the chip by bending out pin 15 (CAS). Windsor didn't miss this one. However, according to markings on the circuit board, I'd zapped bit 7 in bank 0. Windsor told me I had a defective bit 0 and a defective parity bit. Both in bank 0. It got the bank right.

However, it's possible the board was marked wrong, and of course the system would generate a parity error if there were a single bit failure. Windsor should have been smart enough to expect a parity error when it saw the single erroneous bit, so it shouldn't have reported a problem with the parity chip. (In fact, it should have reported a parity problem if it didn't receive a parity error.)

I also tried a dead board, one which had succumbed to a power-supply failure. I had checked a few obvious things earlier. I'd replaced the 8088 and I knew the processor was getting a clock signal. However, Windsor didn't so much as beep. Ah well, probably not their fault.

I'm going to let MicroSphere use this diagnostic ROM for a while to see how helpful it is in real life. (They were very anxious to try it.) I'll keep you posted.

Editor's Suggestions:

As long as I've gotten off on this tangent, I have some suggestions for those of you who are writing RAM diagnostic routines. Be sure to check for bent pins, especially address pins. Turn off refresh, write to RAM and then read at ever-longer intervals. (Remember, a read is a refresh.)

The length of time a RAM remembers its data is one test of RAM quality. (Some RAM chips can remember their data as long as 1 second without refresh.) If you have a flaky RAM chip, hopefully it'll be the first one to forget.

Of course you'll also want to include the standard tests, such as alternately writing and reading 5555 hex and AAAA hex to check for stuck bits and cross talk.

Finally, if you have servicing tips (or recommendations on diagnostics), please write Micro C or call our RBBS. Techtips are the most popular part of the magazine (though it hurts to admit it).

◆ ◆ ◆

HANDS-ON PC REPAIR TRAINING

NAC offers the highest quality advanced training available for PC technicians. The course and its documentation focus on the IBM personal computer lines, from PC to PS/2, peripheral devices, and compatibles such as Compaq and AT&T. Network maintenance training is also available.

- Diagnose more quickly and accurately, complete repairs faster, and reduce costs.
- Discover better diagnostics, parts sources, and new trouble-shooting tools and test units.
- Toll-free technical help and parts-locator line, free technical and vendor update service.
- Used by GTE, ITT, Rockwell, Xerox, SoCal Edison, regional Bell companies, the U.S. Dept. of Commerce, and many others.

Our five-day PC Maintenance Course is offered in California, New York, Washington, Atlanta, Chicago, St. Louis, Dallas, Seattle, and other major cities. Call without obligation to get a course outline and dates for the class locations most convenient to you.

National Advancement Corporation

Computer Hardware Training Specialists

2730-J South Harbor Blvd.
Santa Ana, CA 92704
(714) 754-7110

Reader Service Number 59

DIAGNOSTICS

for Personal Computers from Windsor Technologies

Windsor Technologies offers a family of diagnostic products for IBM PC, PC XT, PC AT and compatible computers. Originally designed for and used by service industry professionals, **Windsor's** products are well suited for anyone involved in the management and maintenance of personal computers.

PC-Technician™

Professional Level Personal Computer Diagnostic System

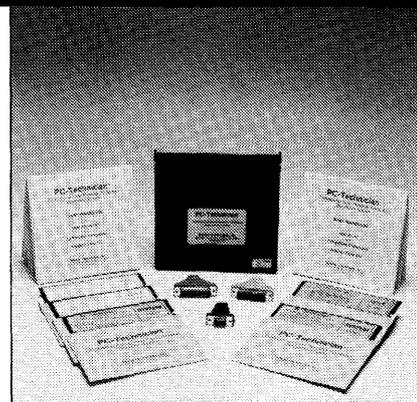
PC-Technician is an 'advanced' type diskette-based diagnostic system that provides comprehensive testing for each individual system component. Information about the testing and the results are presented via text and graphics on the system's display. **PC-Technician** provides on-line help displays and information about the system's configuration.

PC-Technician is ideal for service personnel, manufacturers, OEM's, and users who need detailed information about the status of their computer's hardware.

PC-Technician is available in the following configurations:

- Product Code T01 (for PC, PC XT) **\$195.00**
- Product Code T51 (for PC AT) **\$245.00**
- Product Code T90 (for PC, PC XT, PC AT—
T01 and T51 in the same carrying case) **\$395.00**

A FREE Evaluation Diskette is available—see the coupon below.



T90—PC, PC XT, PC AT

WindsorPOST™

Personal Computer Diagnostic ROM Module

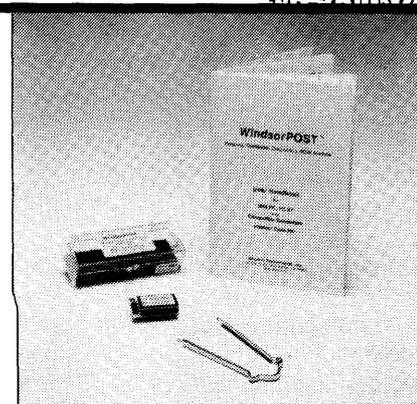
WindsorPOST is designed to be used when the system is malfunctioning such that its 'power-on-self-test' (commonly called POST) does not complete satisfactorily and a diskette-based diagnostic product, such as **PC-Technician**, cannot be loaded and used for troubleshooting.

WindsorPOST temporarily replaces the system's BIOS ROM module. When installed, its LED indicates if the proper amount of power is at the BIOS ROM socket. It then performs 36 tests on the circuits and electrical components on the system board, the system memory up to 640K, and the color or monochrome video display memory. Errors are identified down to the module (chip) level in three full displays of information on the system's video display unit.

WindsorPOST is intended for those service operations where specific identification of failing components is required and where soldered components are removed and replaced. **WindsorPOST** has been in use in the field for over one year and performance data indicates that it properly identifies the failing component at least 70% of the time.

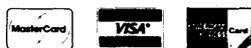
WindsorPOST is currently available for use on IBM PC's and PC XT's (a PC AT type product is under development) and those compatible systems where the system board circuitry and components are extremely similar to the IBM standard. **WindsorPOST** comes with a 30-day money back guarantee.

- Product Code P01 (PC, PC XT) **\$295.00**



P01—PC, PC XT

For immediate ordering, contact **Windsor Technologies** at (415) 456-2200 8:30 AM-5:00 PM Pacific Coast time. VISA, Mastercard, American Express and COD orders accepted.



Volume discounts and reseller discounts are available.

Windsor Technologies, Inc.

130 Alto Street
San Rafael, CA 94901
(415) 456-2200

PC-Technician and WindsorPOST are trademarks of Windsor Technologies, Inc., and IBM PC, PC XT, PC AT are trademarks of International Business Machines Corporation.

Windsor Technologies, Inc.

130 Alto Street, San Rafael, CA 94901 • (415) 456-2200

Please Send A Free **PC-Technician** Evaluation Diskette

Please Send Info About **PC-Technician** **WindsorPOST**

NAME _____ TITLE _____

COMPANY _____ (TELEPHONE) _____

ADDRESS _____

CITY _____ STATE _____ ZIP _____

SERVICE CO. OEM MANUFACTURER DEALER VAR/VAD

IN-HOUSE SERVICE USER OTHER _____

MC 0788

Ray Tracing On The TI34010 (Part 2)

Dealing with Rays, Reflections, & Registers

Last issue Earl introduced you to 3-D graphics (see "Graphics In 3-D," Micro C #41, Pg. 8-15) and the fastest, smartest, most gung ho video controller ever devised (at least by TI). That introduction was quite general. This time out he'll get down to the nitty gritty details of programming the 34010.

If we're going to get down to grits, then let's get on with it. After all, if we're going to be gadding about the screen with rays we'd best get cracking. So first off, let's take a quick review of the 34010.

Numbers are stored as 32-bit fixed point values. Vectors and points are stored as three consecutive 32-bit numbers, representing x, y, and z.

The 34010 assembler lacks the splendid structure constructs of MASM and C, so we use equates to build structures (Note: On the 34010 address offsets are in bits) —

```
;Point
px    equ 0000H
py    equ 0020H
pz    equ 0040H
```

A ray has an origin and a direction, and is stored in memory as two consecutive vectors —

```
;Ray
rox   equ 0000H ;origin
roy   equ 0020H
roz   equ 0040H
rdx   equ 0060H ;direction
rdy   equ 0080H
rdz   equ 00A0H
```

We define a plane by a point on the plane and a normal vector. The test for whether a point lies on a plane is —

$$P \cdot N = c$$

where $c = O \cdot N$ is a constant.

The plane structure consists of the normal vector, a point on the plane, and the computed constant, $c = O \cdot N$ —

```
pnx   equ 0000H ;normal vector
pny   equ 0020H
pnz   equ 0040H
ppx   equ 0060H ;point on the plane
ppy   equ 0080H
ppz   equ 00A0H
pc    equ 00C0H ;constant for quicker
      ;computations
      ;= point.normal
```

A rectangle will use the structure for a plane, but with two additional vectors, the X extent and the Y extent —

```
;Rectangle
rx    equ 00E0H ;X extent
ry    equ 0140H ;Y extent
```

We represent a sphere by a center and a radius. The test for whether a point lies on a sphere uses the square of the radius. So to simplify calculations we store the square of the radius (not the radius) in the sphere structure —

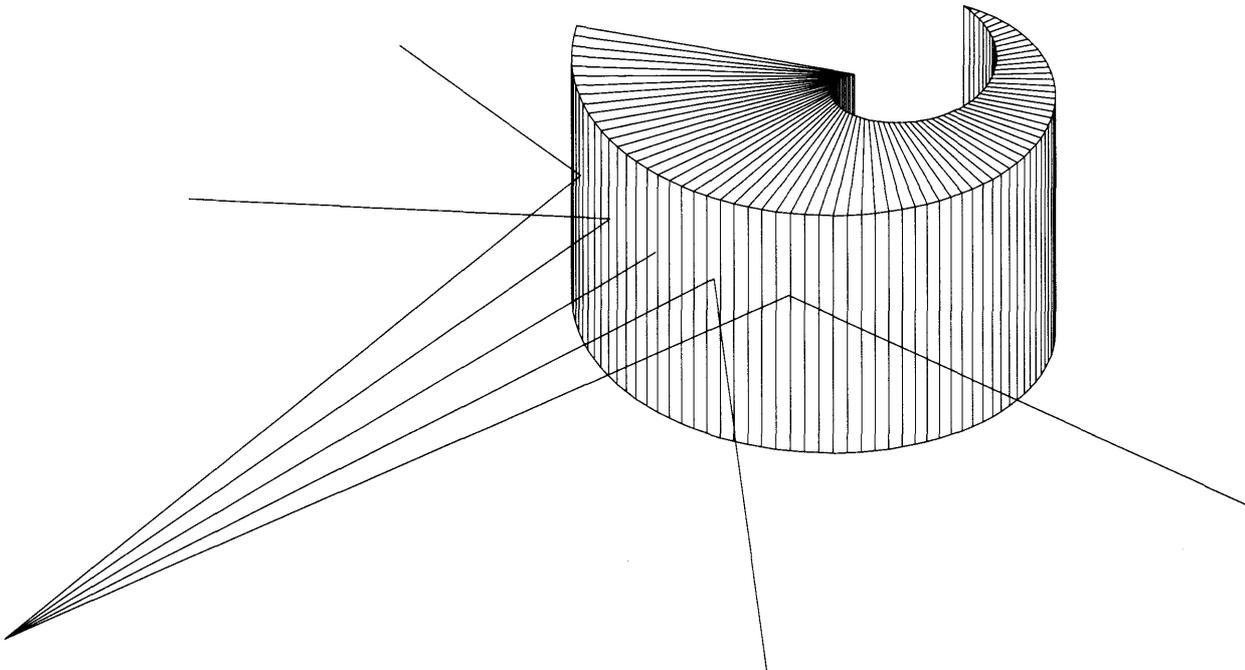


Figure 1 — Find a Point of Intersection

```
.text
PointOnLine:
;In:  a8 = Ray
;     a9 = Time parameter
;     a10 = Point
    move *a8(rdx), a0, 1      ;fetch x direction
    move a9, a1               ;scale by time parameter
    calla fxMultiply
    move *a8(rox), a1, 1      ;add x starting value
    add a1, a0
    move a0, *a10(px), 1      ;store result
    move *a8(rdy), a0, 1      ;repeat for y
    move a9, a1
    calla fxMultiply
    move *a8(roy), a1, 1
    add a1, a0
    move a0, *a10(py), 1
    move *a8(rdz), a0, 1      ;and for z
    move a9, a1
    calla fxMultiply
    move *a8(roz), a1, 1
    add a1, a0
    move a0, *a10(pz), 1
    rets
```

Figure 2 — Intersection of Ray with Plane

```
intLinePlane:
;In  a8 = Ray
;    a9 = Plane
;Out: a0 = intersection parameter
;     a0 <= 0 if no intersection
    mmtm sp, a4, a5, a6, a7, a8, a9
    calla vDot      ;ro*pn (ray_origin dot plane normal)
    move a0, a4
    addi rdx, a8
    calla vDot      ;rd*pn (ray direction dot plane normal)
    move a0, a1
    move *a9(pc), a0, 1
    sub a4, a0
    calla fxDivide
    mmfm sp, a4, a5, a6, a7, a8, a9
    rets           ; (pc-ro*pn)/(rd*pn)
```

```
;Sphere
scx equ 0000H ;center
scy equ 0020H
scz equ 0040H
sr2 equ 0060H ;radius squared
```

Intersection Routines

The intersection routines will answer the question “when does the ray intersect the object” rather than “where does the ray intersect the object.”

Often “when” is all we need to know. The subroutine in Figure 1 computes “where” from “when” for those instances when we need the actual position.

On input, a8 points to a ray structure, a9 is the time parameter, and a10 points to an empty point structure. The routine fills the point structure at a10 with the position of our moving point at time a9.

The subroutine in Figure 2 computes the intersection of a ray with a plane. The subroutine will return the time parameter for a point traveling along the ray.

Input to the subroutine has a8 pointing to a ray structure and a9 pointing to a plane structure. The time of intersection is returned in a0.

Situations may arise where the ray never intersects the plane. For example, the ray may be pointing away from the plane. This situation is flagged by a negative value being returned in a0.

Figure 3 computes the intersection of a ray with a rectangle. Our first step here is to make sure the ray intersects the plane. We do it by calling intLinePlane.

If there’s no intersection with the plane then we quit. Otherwise, we must determine whether the point of intersection lies within the rectangle. A call to PointOnLine puts the point of intersection in a temporary vector.

We then subtract the origin from this point. Then we compute the projection of this vector onto both the X and Y extents to check that the point lies in the bound-

ary of the rectangle.

The subroutine in Figure 4 computes the intersection of a ray and a sphere. Input to the subroutine has a8 pointing to a ray structure, and a9 pointing to a sphere structure. The intersection time is returned in a0.

For an arbitrary ray and sphere, the ray will intersect the sphere in zero, one or two points. Each of these situations has a geometric interpretation.

Earlier we derived a quadratic equation for computing the intersection. If both roots are imaginary then the ray completely misses the sphere. If both roots are negative then the ray is pointing away from the sphere. If one root is positive and one is negative, then the ray starts inside of the sphere. If both roots are positive then the ray begins outside the sphere and goes through the sphere.

Our routine will return a negative value if the ray doesn't intersect the sphere. If the ray intersects the sphere twice, the routine will return the time of the earliest intersection.

Some extra precaution is necessary in these calculations. The multiplication, division, and square root routines introduce some round-off errors. These errors are far less than the theoretical distance between pixels. So these errors never cause something to be displayed that shouldn't be.

However when we look at a reflected ray, the ray starts right on the circle. In a perfect world, the computer routine will return a zero. But because of round-off error, we may see a small positive number. Without some adjustment, the computer will decide that the reflected ray immediately intersects the circle again (and again, and again, and again...).

To avoid this situation, the program will test against a small positive number rather than 0. This number was determined by experimentation.

Reflection

Ray tracing images really become interesting when some surfaces are made reflective or partially reflective. So we *must* ask what happens when the ray strikes a reflective surface.

Reflection from the surface of a sphere is the same as reflection from the surface of the plane tangent to the sphere at the point of intersection. Thus, once the tangent plane is found, the case of reflection off a sphere is reduced to reflection off a plane.

The normal to the plane tangent to the circle is the vector from the center of the sphere to the point of intersection. The

Figure 3 — Intersection of Ray with a Rectangle

```
intLineRectangle:
;In:   a8 = ray
;      a9 = rectangle data
mmtm sp, a4, a5, a8, a9, a10
move a9, a5 ;save rectangle pointer
calla intLinePlane ;Test for intesection with embedding plane
btst 31, a0
jrnz ir_01 ;if not then quit
move a0, a4 ;Save intersection time
move a0, a9 ;a8 = ray, a9 = time
movi vTemp0, a10 ;a10 will hold point of intersection.
calla PointOnLine
move a10, a8 ;Intersection point
move a5, a9
addi 60H, a9 ;minus origin
calla vSub ;to temp.
addi 80H, a9 ;point to X extent
calla vCalibrate ;project temp onto X extent
btst 31, a0 ;miss if negative
jrnz ir_01
movi One, a1 ;or greater than one
cmp a1, a0
jrnz ir_01
addi >60, a9 ;point to Y extent
calla vCalibrate ;test projection onto Y side
btst 31, a0 ;miss if negative
jrnz ir_01
movi One, a1 ;or if greater than 1
cmp a1, a0
jrnz ir_01
move a4, a0 ;return time of intersection
jrnc ir_00
ir_01:
movi -1, a0 ;return negative number if miss.
ir_00:
mfm sp, a4, a8, a9, a10
rets
```

Figure 4 — Intersection of Ray with a Sphere

```
intLineSphere:
;In   a8 = Ray
;     a9 = Sphere
;Out: a0 = intersection parameter
;     a0 <= 0 if no intersection
mmtm sp, a4, a5, a6, a7, a8, a9, a10
move a8, a4
move a9, a5
movi vTemp0, a10
calla vSub ;temp0 = ro-sc
addi rdx, a8
move a8, a9
calla vDot ;rd*rd
move a0, a6 ;'a' in quadratic equation
move a10, a9
calla vDot
add a0, a0
move a6, a1
calla fxDivide
move a0, a7 ;'b' in quadratic equation
```

```

move a10, a8
calla vDot
move *a5(sr2), a1, 1 ;radius squared
sub a1, a0 ;'c' in quadratic equation
move a6, a1
calla fxDivide
sll 2, a0 ;4ac
move a0, a6
move a7, a0
move a7, a1
calla fxMultiply ;b^2
sub a6, a0 ;b^2-4ac
jrslt is_00 ;no intersection if negative.
calla fxSqrt ;sqrt(b^2-4ac)
move a7, a1 ;b
neg a1 ;-b
sub a0, a1 ;-b-sqrt(b^2-4ac)
sra 1, a1 ;(-b-sqrt(b^2-4ac))/2
cmpi SphereMinTime, a1 ;test with room to get started.
jrgt is_01 ;if positive then is first intersection.
add a0, a1 ;(-b+sqrt(b^2-4ac))/2
;if positive then it is the intersection,
;if negative then there is no intersection.
;In either case value is returned as is.

is_01:
move a1, a0
is_00:
mmfm sp, a4, a5, a6, a7, a8, a9, a10
rets

```

point of intersection is a point on the tangent plane. The normal and the intersection point define the plane. So only the case of reflection of a ray off of a plane will be considered in detail.

I started with the rule "Angle of incidence = angle of reflection." The angle, α , between two vectors, U and V , can be computed by —

$$\cos \alpha = \frac{U \cdot V}{|U| * |V|}$$

Originally, I developed reflections using this idea. However some algebraic simplification leads to a much simpler subroutine. It turns out that the algebraic simplification has a nice geometric interpretation. I've presented the simplified approach here.

Let I denote the incoming or incident vector, and let R denote the reflected vector. Let N be the normal vector to the plane. Decompose I into a component perpendicular to N , and a component parallel to N . Call the perpendicular component A and the parallel component B . So $I = A + B$.

Consider how each of these components is reflected off the plane. A is



DIAGNOSTICS

The Complete Diagnostics Solution for Your PC/XT, PC/AT, or Compatible

INCLUDES...

DRIVE TESTS—Complete diagnostics for Hard and Floppy drives, including controller cards. Tests read, write, and format capability as well as seek timings, historesis and rotation timings.

I/O PORTS—For both parallel and serial ports, confirms internal and external loopback capabilities at all baud rates and configurations.

MEMORY—Performs over eight different tests to check standard, extended, and expanded memory.

KEYBOARD—Verifies that all keys send correct key codes, including shift, CNTRL, and ALT modes.

CPU & NUMERIC COPROCESSOR—Verifies that all single and multiple instructions perform correctly and accurately, as well as testing all internal registers.

VIDEO DISPLAY—Checks video controller cards. Confirms attributes, graphics, colors (if applicable), and CRT alignment patterns.

REAL TIME CLOCK—Verifies correct timing, all internal registers, and battery backed up RAM.

...and many more features to insure the integrity of your computer.

PC/XT Diagnostic Software	\$ 49
PC/XT Disk Diagnostics (w/ test diskettes)	\$ 49
PC/XT I/O Loopback Test Plugs	\$ 19
COMPLETE PC/XT DIAGNOSTICS SET (save \$17)	\$ 99

PC/AT Diagnostic Software	\$ 49
PC/AT Disk Diagnostics (w/ test diskettes)	\$ 49
PC/AT I/O Loopback Test Plugs	\$ 19
COMPLETE PC/AT DIAGNOSTICS SET (save \$17)	\$ 99

BOTH PC/XT and PC/AT SETS (save \$55)	\$179
---------------------------------------	-------

After eight years of designing superior diagnostic software for manufacturers of computer equipment, **CAPITAL SOFTWARE** has created the definitive diagnostics package for both personal and professional use. For your custom needs, let **CAPITAL's** experienced team provide the best solution available.

SEND CHECK OR MONEY ORDER TO
 CAPITAL SOFTWARE
 300 DAVEY GLEN ROAD, SUITE 3311
 BELMONT, CALIFORNIA 94002
 (415) 592-9076

USE YOUR VISA OR MASTER CARD



parallel to the plane, so it isn't changed by reflection. B is normal to the plane, so it's fully reflected by the plane. B becomes -B after reflection. Thus the reflected vector R is A - B.

The decomposition of I into A and B has been given early —

$$B = \frac{I \cdot N}{N \cdot N} N$$

$$A = I - \frac{I \cdot N}{N \cdot N} N$$

So —

$$R = I - 2 * \frac{I \cdot N}{N \cdot N} N$$

Programming Reflections

The subroutine refLinePlane in Figure 5 will determine the reflection of a ray off a plane. On input, a8 points to the ray structure, and a9 points to the plane structure.

a10 is the intersection (point on ray) parameter returned from the intersection subroutine. a11 points to a memory area where the reflected ray will be built. The structure at a11 holds the origin and direction of the reflected ray.

The origin of the reflected ray is simply the intersection of the incident ray and the plane. The PointOnLine subroutine with the incident ray, a8, and the time parameter t, gives the origin. The reflected direction is computed from the formula —

$$R = I - 2 * \frac{I \cdot N}{N \cdot N} N$$

The incident vector is part of the ray structure at a8, and the normal vector is part of the plane structure at a9.

The reflection of a ray off a sphere is very similar to the reflections off a plane. The refLineSphere routine (Figure 6) will build the plane tangent to the sphere at the point of intersection between the sphere and the ray.

I've taken some nasty shortcuts here. First observe that in the refLinePlane routine, the only information about the plane that is used is the normal vector. So the normal vector is the only information the refLineSphere routine supplies in the tangent plane structure. Calculating the other information would be a waste

Figure 5 — Reflection of Ray Off Plane

```

refLinePlane:
;In:   a8 = Ray
;      a9 = Plane
;      a10 = intersection (point-on-line parameter)
;      a11 = reflection
mmtm sp, a4, a5, a6, a7, a8, a9, a10, a11
move a8, a4      ;Line
move a9, a5      ;Plane
move a10, a9     ;point distance
move a11, a10    ;build reflective line here
calla PointOnLine ;origin.
lp_01:
addi rdx, a4     ;line direction
move a4, a8
move a5, a9     ;plane normal
calla vDot      ;ld*pn
sll 1, a0       ;2*ld*pn
move a0, a7
move a5, a8     ;pn
calla vDot      ;pn*pn
move a0, a1
move a7, a0
calla fxDivide  ;2*ld*pn/(pn*pn)
move a0, a9
movi vTemp0, a10
calla vScale   ;(2*ld*pn/(pn*pn))pn
move a4, a8    ;ld
move a10, a9
move a11, a10
addi rdx, a10  ;reflection direction
calla vSub     ;ld-(2*ld*pn/(pn*pn))pn
mmfm sp, a4, a5, a6, a7, a8, a9, a10, a11
rets

```

END OF LISTING

Figure 6 — Reflection of Ray Off Sphere

```

refLineSphere:
;In:   a8 = Line
;      a9 = Sphere
;      a10 = intersection (point-on-line parameter)
;      a11 = reflection
mmtm sp, a4, a5, a6, a7, a8, a9, a10, a11
move a8, a4      ;Line
move a9, a5      ;Sphere
move a10, a9     ;point distance
move a11, a10    ;build reflective line here
calla PointOnLine ;origin.
move a10, a8     ;create tangent plane,
move a5, a9      ;normal ,= intersect pt - sphere center
movi vTemp1, a10
calla vSub
move a10, a5     ;only need normal
jruc lp_01      ;continue in line / plane intersection.

```

END OF LISTING

of time.

And while constructing the tangent plane, `refLineSphere` first must calculate the point where the ray intersects the sphere. This point is also calculated by `refLinePlane`. So `refLineSphere` jumps to the middle of `refLinePlane`, to a place after the point of intersection calculation. This avoids calculating the point of intersection twice.

Generally, the execution time saved by tricks like these won't offset the program maintenance headaches created by the tricks.

Wrap Up

I'll close with a description of the remainder of the ray tracing program. I'll only show you the algorithm, skipping the details of the 34010 implementation.

We begin by building a list of objects in memory. These objects can be anything we've discussed so far — planes, rectangles, and spheres. In addition to the location and size information, we also specify a color for each object. Call this table the Universe.

We imagine an eye somewhere in this computer universe. The location of our eye is stored in a point structure called Eye. The computer display screen is mapped to a rectangle in space. We use a rectangle data structure, called Screen, to describe the display screen location.

Every pixel on the screen will be colored. The main part of the program is a loop. The program loop picks a pixel on the display screen. This pixel is translated to a point in the display rectangle.

A ray is built which starts at the eye and goes through the point on the display screen. We determine the first object this ray intersects; the color of that object is used to color the pixel. The program loop is repeated for every pixel on the screen.

The program loop will be controlled by a routine called `NextDot`.

`NextDot` is called at the start of every loop. `NextDot` will return the coordinates of the next pixel to be colored, or an indication that the picture is finished. The order in which `NextDot` returns pixels isn't important to the final picture. However, we can watch the picture be built on the screen.

The natural tendency is to do each row from left to right, one after the other from top to bottom. A better idea is to have `NextDot` pick points scattered all over the screen. This way one has a good idea of how the final picture will look long before the last pixel is plotted.

The pixel coordinates are changed

into a percentage of screen fraction by dividing them by the full screen extent. The dot position, as a percentage of screen, is passed to a routine named `ViewRay`. `ViewRay` will build up the viewing ray.

A point on the screen rectangle is computed from the screen position. If O, X, Y are the origin, X extent, Y extent of the screen rectangle, and a and b are the X and Y percent of screen parameters, then we'll be looking at the point $P = O + a * X + b * Y$.

If E denotes the eye, then the start of the view ray is E and the direction is P - E. `ViewRay` will normalize the direction vector before returning.

A subroutine called `RayComputations` will take as input a ray and return a color. This routine computes the intersection time of the input ray with each object in the Universe table. The object with the earliest intersection time is determined. If the ray doesn't intersect any object, the "a" background color, usually black, is returned.

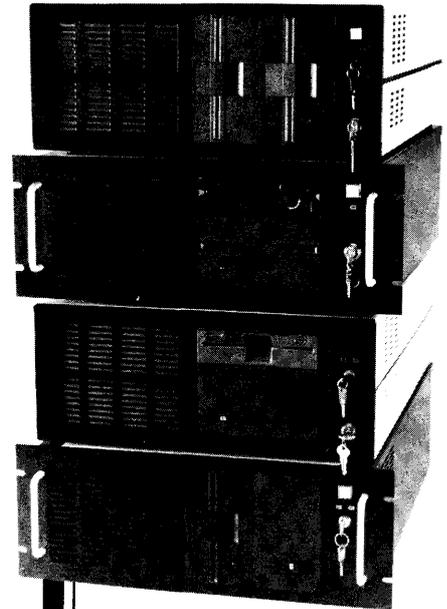
When the earliest intersection is determined, `RayComputations` will send the ray, the time of intersection, and the object of intersection to a routine called `ColorOfObject`. `ColorOfObject` will return the color of the intersected object.

An object may have a simple coloring rule such as all points on the object have the same color. Or a more complex coloring rule, such as covering the object with multicolored squares, stripes of circles.

Some objects may be designated as reflective. In this case, the reflected ray is built and `RayComputations` is called recursively to determine the color. Whatever the coloring rule of the object, `ColorOfObject` will determine the color and return it to `RayComputations`. `RayComputations` then returns the color to the main program loop.

The color returned by `RayComputations` is sent to `ColorDot` to place the color on the display screen.

◆ ◆ ◆



Integrand's new Chassis/System is not another IBM mechanical and electrical clone. An entirely fresh packaging design approach has been taken using modular construction. At present, over 40 optional *stock* modules allow you to customize our standard chassis to nearly any requirement. Integrand offers high quality, advanced design hardware along with applications and technical support *all at prices competitive with imports*. Why settle for less?

Rack & Desk PC/AT Chassis

Rack & Desk Models

Accepts PC, XT, AT Motherboards and Passive Backplanes

Doesn't Look Like IBM

Rugged, Modular Construction

Excellent Air Flow & Cooling

Optional Card Cage Fan

Designed to meet FCC

204 Watt Supply, UL Recognized

145W & 85W also available

Reasonably Priced

Reader Service Number 22

INTEGRAND
RESEARCH CORP.

Call or write for descriptive brochure and prices:
8620 Roosevelt Ave. • Visalia, CA 93291

209/651-1203

TELEX 5106012830 (INTEGRAND UD)

EZLINK 62926572

We accept BankAmericard/VISA and MasterCard

IBM, PC, XT, AT trademarks of International Business Machines.
Drives and computer boards not included.

XT Theory Of Operation

Putting All those Smart Chips Together

Larry's been researching and writing the pieces on the XT's LSI chips (the processor, timer, DMA controller...) so it was natural that he take the lead in this final piece. This is the theory of operation, the "how all those fancy ICs work together as a computer."

We're also finishing up one of our famous wall-sized schematics for the XT. If you have a hankering to really see what we're talking about then check the end of this article for details.

Okay, folks, here it is, pretty nearly the whole circuit in one place. We're using device names (S04 for 74S04) in place of U numbers and we're using signal names (e.g., RESET) in place of pin numbers. We're doing this because most of you have clones, and though the U numbers and pin numbers may differ, the device names and the signals remain (in nearly all cases) the same.

Note: A signal named RESET would be active (would reset the system) when the line is high (4V or above). A signal named -RESET (sometimes called RESET NOT or RESET BAR) would be active when the line is low (1V or lower). Schematic drafters usually put a line above names of signals that are active low.

Clock

The PC's heartbeat comes from an 8284 clock generator. (All of the 8000 series chips here are spawn of Intel.) Actually, the 8284 spits out three clock signals.

OSC, a 14.31818 MHz symmetric square wave, goes to the I/O bus for use by peripheral cards. This signal is also used by the 8284 to create the other two clock signals.

The frequency of the OSC output depends on the crystal connected to the 8284. A variable capacitor, labeled "color

adjust" by IBM, lets you fine tune the crystal's frequency. The name "color adjust" comes from the fact that IBM's CGA card requires a very precise 14.31818 MHz and 3.579 MHz (14.31818/4).

The 8284 can also accept an external clock input and switch between that and OSC. But its clock source select input is held low through an LS04 inverter connected to +5V. This assures that OSC will always be selected — a good thing since the external clock input connects to thin air.

But why not just tie the select line to ground? Dave guesses the unnecessary inverter is a vestige from the XT's development days when IBM may have toyed with a two-speed system. Larry thinks it's sheer perversity. See *Micro C* issue #31 for a 7.37 MHz switchable speedup which utilizes this feature of the 8284.

CLK88 provides the 4.77 MHz, 33% duty cycle (2/3 low, 1/3 high) signal which serves as the main system clock. The third output, a peripheral clock called PCLK, gives the 8253 timer chip and the keyboard section a symmetric reference signal — also 4.77 MHz.

In True Blue XTs, the 8284's -RES input connects to a PWGOOD signal from the power supply. So a loss of power causes a reset. None of the clones we've looked at implement this feature. But you can create your own reset switch by shorting the reset input to ground. This initiates a high on the 8284's RESET output which connects directly to the processor and any coprocessor. The falling edge of this RESET triggers the reset sequence.

During Direct Memory Access (DMA) transfers, some rather convoluted logic circuitry sets up the first of two possible READY inputs to the 8284. With this input held low by the -DMAWAIT signal, the 8284's READY output goes low. When the 8088 sees this low, it inserts

wait states until the DMA controller finishes its transfer. READY can also be forced low by slow devices on the I/O bus (telling the processor to wait a bit).

Processor

So much for the heart — on to the brains. We'll talk about most of the 8088's functions when we deal with the peripheral chips. But one thing should be mentioned here.

The 8088 can be configured in two modes. In the XT, the mode select input MN/MX (min/max) connects to ground. This selects the maximum mode where the 8088 receives help generating bus control signals and can deal with coprocessors.

The Busses And The Bus Controller

The 8088 can handle eight bits of data and 20 bits of address. It needs only 20 pins for both data and address because it uses 8 pins for double duty (these are labeled AD0 through AD7). So, the 8 bits of data share the 8 least-significant address lines. Separating the data from the address takes place as follows:

During the first part of a bus cycle, the 8088 places a complete address on its AD0-AD7 and A8-A19 pins. It also sends information to the 8288 bus controller via status lines S0-S2. The 8288 decodes this status information and generates an Address Latch Enable (ALE). (By the pint, of course.)

ALE latches most of the address into two LS373s. An LS244 buffers the remaining four bits, A8-A11. Address Enable Board (AENBRD - see DMA discussion below) enables outputs for all three chips.

Once the address has been latched, we deal with data. New status outputs from the 8088 tell the 8288 to generate a data transmit/receive (DT/-R) signal. DT/-R controls the direction of data flow through an LS245 octal bus transceiver connected to AD0-AD7. So we have data

and we know whether the processor will be receiving or sending. Another tweak of the status lines makes the 8288 pump out one of four sources and destinations: I/O read (-IOR), I/O write (-IOW), memory read (-MEMR), or memory write (-MEMW).

Finally, through its status lines and the 8288, the 8088 sends a Data Enable (DEN) to the LS245 and we've placed address and data on the address and data busses.

That takes care of one set of address and data signals: the ones belonging to the *internal* busses. These busses talk to the processors, RAM addressing logic, RAM data, and most of the I/O bus (J1-J7). J8 (the right-most expansion slot) is the black sheep of the family. More on J8 later.

External Address And Data

The demultiplexing circuitry described earlier doesn't have enough oomph to talk to everything on the address, data, or control busses. (Every input on an address line, for instance, adds a little capacitance and a little current drain to the line.) So, in order to give these little demuxers a break, the busses are divided into two parts, internal and external. The demultiplexers drive the internal address and data busses. Buffers then transfer the internal signals to the external busses.

The external address bus, for instance, services the system RAM, system ROM, I/O ports, and expansion slot J8.

Three LS244s buffer the addresses onto the external bus. Two of the LS244s (carrying A8-A19) have their two active-low output control pins tied to ground. So part of the external address bus (XA8-XA19) always reflects exactly the contents of the internal bus.

The LS244 responsible for A0-A7 has its output control pins tied to AENBRD. That's fine. But why treat A0-A7 differently than A12-A19? Both groups of

addresses originate in LS373s controlled by identical inputs. If anyone out there was involved in the design of the XT (or knows someone who knows someone who was involved in the design), we'd sure like to talk to you. Some of the workings of this machine aren't immediately obvious.

The external data bus is a bit stickier.

If the NMI makes it through the maze, the 8088 jumps to the NMI handler and things grind to a halt.

Addresses go only one way — from the processor (or DMA) to memory or I/O. Data has to travel in both directions. An LS245 octal bus transceiver handles the bidirectional buffering.

The LS245's direction control input (DIR) connects to an LS27 positive NOR gate which has three inputs. When any of the three inputs goes high, the output of the NOR gate goes low. That forces the LS245 to pass data from the external data bus to the internal data bus.

An active ROMADDRSEL (high) on the first LS245 input means the processor wants to read the ROM. Just right. The ROM lives on the external bus and the processor (of course) hangs out on the internal bus.

The Interrupt Acknowledge (-INTA)

and Card Selected (-CARDSLCTD) signals connect to an LS00 positive NAND gate which provides the second input to the LS245. When either is active, the LS245 sees a high from the LS00 and sets the direction of data from external to internal bus.

During an -INTA the interrupt controller writes data (a pointer to an interrupt handler vector) on the external bus. This has to be read by the processor on the internal bus, so the LS245 sets the direction accordingly.

Slot J8 has tighter timing characteristics and also is the only socket using the -CARDSLCTD signal. The system expects any card in J8 to respond via this signal line whenever the card is selected (shades of the new PS-2 system where every card is expected to respond). An active -CARDSLCTD sets the data direction from external to internal. But that means that whenever we select a card in J8, *all* we can do is read.

No problem. Or, at least not much of a problem. J8 was designed for extra ROM cards and you're not going to be writing very much into the ROMs. Try to put a floppy controller in J8, it wouldn't be smart enough to respond with a -CARDSLCTD and even if it did, we wouldn't be able to write data to it. The ultimate in write-protected disks.

The last of the three inputs to the LS245 comes from an LS27 positive NOR gate. This gate's output goes high whenever all three of its inputs, External I/O Read (-XIOR), XA8, and XA9 go low. That is, when a read of one of the lower FFh I/O ports occurs.

These ports belong to the system board and include the DMA Controller, Interrupt Controller, Counter/Timer, and Programmable Peripheral Interface (parallel chip) — all of which connect to the external data bus. So a read of any of the system ports on the external bus sets the LS245 direction from external to internal.

NMI

Parity errors on the system board or the I/O channel, or 8087 interrupts (caused by an "exception" such as divide by zero) can generate a non-maskable interrupt. But NMIs travel a tortuous path of logic gates. They have to pass through gates controlled by one of the system board parallel ports. Then the NMI gets ANDed with the latched contents of an LS74 NMI Mask Register.

If the NMI makes it through the maze, the 8088 jumps to the NMI handler and things grind to a halt.

8087

The 8088 has two Request/Grant lines that can be used to coordinate the activities of coprocessors. The XT only uses one, RQ/GT1. This connects directly to RQ/GT0 of the 8087 and allows the 8087 to request use of the bus and the 8088 to

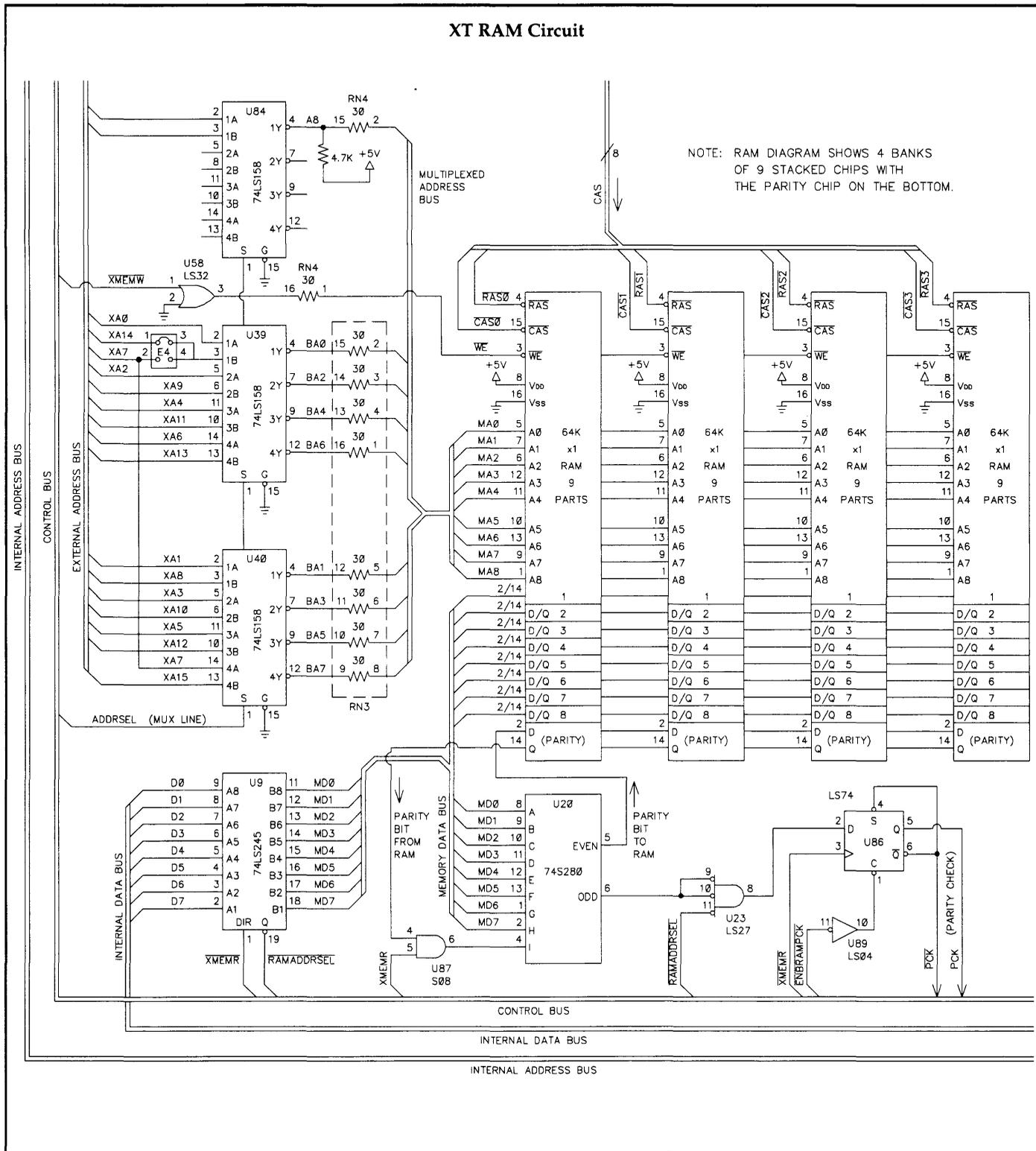
respond when it's ready.

Also, the BUSY output of the 8087 connects to the -TEST input of the 8088. This lets the 8088 know when the 8087 has finished a task.

I/O Space

I/O addresses can be broken into two groups. As mentioned above, the lower FFh I/O ports belong to the system board. 100h and up live on the expansion

XT RAM Circuit



bus. Let's look at chip selection on the system board — for example, how the DMA Controller is selected.

Active-low chip selects for the four smart system board chips come from an LS138 decoder. The LS138 decodes XA5-XA7 and must see lows on XA8, XA9, and a high on -AEN to turn on its outputs. It doesn't connect to any higher order address bits, so the only significant addresses are those we can make with XA7 and lower. That's FFh and less, just like we wanted.

Any address in this range with all of the select inputs (XA5-XA7) low will cause the LS138 to generate a DMA chip select. That's any address from 0-1Fh. This addressing in combination with an active -X10W allows for programming of the DMA controller.

Expansion boards must generate their own chip selects in response to addresses on the expansion bus.

Direct Memory Access

The DMA Controller (8237) handles data transfers (for example, disk I/O) without the intervention of the processor. Of the four sets of control signals for the DMA channels (DREQ1, -DACK1, etc.), three connect directly to the expansion bus for use by peripheral cards. The fourth, channel 0, is dedicated to memory refresh.

The 8237 generates its own addresses through its A0-A7 and DB0-DB7 pins. The low order byte of address goes from A0-A7 to an LS244 octal buffer. Bits 8-15 come from DB0-DB7 and feed an LS373 octal latch. This portion of the address gets strobed into the LS373 by the 8237's Address Strobe (ADSTB) output. The last four bits of address don't come from the DMA Controller.

The DMA Page Register holds address bits 16-19. This LS670 4X4 register file must be programmed before a DMA transfer with the high nybble of address for whatever DMA channel will be used.

All 3 chips have their outputs enabled by DMA Address Enable (-DMAAEN) and the full address goes out onto the internal address bus. -DMAAEN occurs in response to a Hold Request (HRQ) by the 8237.

HRQ requests that the 8088 release the busses for the DMA activity. The 8088 doesn't respond directly to the request. Instead, an LS20 positive NAND gate watches -S0, -S1, -LOCK, and HRQ. The only time -S0 and -S1 are both high is when the processor's either halted or has finished an instruction.

So, when HRQ goes high and the

Parity is simple. Sorta. We're dealing with 8 data bits and one parity bit. The parity bit is used to make sure the total number of "1" bits is odd.

processor finishes its current task, the output of the LS20 goes low (assuming we're not executing a locked instruction). This low output wends its way through an LS04 inverter, an LS175 flip-flop and an LS74 flip-flop to eventually generate the following signals.

First is a hold acknowledge for the DMA Controller. This signal also feeds back into the LS175 to create AENBRD and -AEN. AENBRD tells the expansion boards that DMA has control of the busses and -AEN does the same for the system board.

AENBRD goes through the LS175 yet again and produces the -DMAWAIT signal discussed in the clock section above and (finally!) -DMAAEN.

When the DMA Controller finishes a transfer it drops its End Of Process (-EOP) output. -EOP gets inverted by an LS04 and becomes Terminal Count (T/C) on the expansion bus. A high T/C tells expansion cards that DMA activity has ended.

During DMA activity, the DMA chip takes over the external control bus since the 8088 and 8288 are out of the picture.

Counter/Timer

The first of the three Counter/Timer channels (channel 0) in the PC's 8253 Programmable Interval Timer connects directly to the Interrupt Controller's IRQ0 input. This channel triggers the time of day interrupt.

Channel 1 sets an LS74 flip-flop which generates a DMA request for DMA channel 0. DRQ0 in turn causes a memory refresh cycle by the DMA controller. And the channel 0 acknowledge signal (-DACK0BRD) from the controller resets

the flip-flop in preparation for the next DRQ0.

Channel 2's output is ANDed with SPKRDATA from the 8255 parallel chip's port B bit 1 (in an S08). The result is buffered through the open collector 75477 to drive the XT's speaker. Also, Port B, bit 0 provides a gate for channel 2. So this bit must be high before the channel will count.

Since square waves have an irritating rasp, there's a low-pass filter (a resistor-capacitor network which blocks the high frequency associated with the rise and fall of a square wave) which, combined with the inductance of the speaker, turns the square wave into a (sorta) sine wave. Much gentler on the ears.

Interrupts

The 8259 Interrupt Controller accepts eight hardware Interrupt Requests. IRQ2-IRQ7 come from the expansion bus and deal with disk drives, printers, and serial ports.

IRQ0 sees the time of day clock ticks from channel 0 of the 8253 Timer. The 8259's INT output goes directly to the processor's Interrupt Request (INTR) input. The 8088 responds through its status outputs and the bus controller with an Interrupt Acknowledge (-INTA).

In response to a second -INTA, the 8259 releases a pointer to the appropriate interrupt handler vector through its D0-D7 pins. These data pins connect to the external data bus.

The last request, IRQ1, comes from the keyboard.

Keyboard Interface

The scan code from a keystroke comes into the system board as a stream of eight serial data bits with two start bits. An LS322 shift register takes care of converting that serial stream into a parallel byte. As the first start bit enters the LS322, it gets latched into the first of eight single-bit latches. As each following bit enters the LS322, the first bit shifts to latch 2, then latch 3, and so on. KBDCLK, a clock signal generated in the keyboard, times these shifts.

When the first of the start bits shifts into the least significant (eighth) latch, the following happens. Although the register's outputs have not officially been enabled, output -QH does reflect the contents of latch 8 (bit 0).

This signal, -QH, clocks through an S74 flip-flop and generates both IRQ1 and the register enable input for the LS322. By the time the register enable hits the LS322, the last two bits of data have

Performance and versatility For your CP/M or MS-DOS computer

QP/M

QP/M by MICROCode Consulting

Fed up with the message "BDOS error: R/O"? With QP/M, you'll never lose another file because you changed a diskette. QP/M offers full CP/M 2.2 compatibility with outstanding performance and more commands WITHOUT eating up precious program space. Get such features as automatic disk relogging, simple drive/user selection using either a colon or semi-colon, 31 user areas, drive search path, multiple program command line, archive bit maintenance, and transparent time/date stamping; all in the same space as CP/M 2.2. Installs from a convenient customization menu, no software assembly required. Bootable disks available with CBIOS for Kaypro, Xerox (8" or 5 1/4", -1 or -2), & BBI.

QP/M Operating System, bootable - specify system . . . \$ 64.95
QP/M without CBIOS (installs on any Z80 system) . . . \$ 49.95

Networks

QP/M Network File System by MICROCode Consulting

QP/M Network File System is an efficient local area network allowing up to seven CP/M computers to share peripherals and data resources.

- Transparent operation at speeds up to 11,000 bytes/second in synchronous mode
- Speeds of up to 1,920 bytes/second in asynchronous mode
- Local/remote disk drive and printer support
- Remote peripheral support for modems and real-time clocks
- All stations need not be on the network even though connected
- Local drive access protection and control
- Simple menu oriented configuration utility
- Extended DOS calls are provided for addition of custom network utilities.

Works with interrupt driven Z80 systems such as Xerox 820, Kaypro (KayPLUS & Advent ROMs), Eagle, and other computers running QP/M, or CP/M 2.2

QP/M Network File System \$ 39.95

Hard Disks

Need more speed and storage on your system?

Improve the productivity of your Z80 computer with a hard disk.

HDS Host Board

This daughter board provides a convenient interface for connecting a Western Digital WD1002-05 hard disk controller to your computer.

- Plugs into the Z80 socket, no other wiring required
- 40 pin interface for a WD1002-05 (or HDO) controller board
- Switch selectable I/O port addressing
- Comes as bare board or assembled & tested
- Kaypro '84 host board also available

Winchester Connection by MICROCode Consulting

The most simple and comprehensive hard disk software package available for CP/M.

- Designed for use with the WD1002-05 controller board
- Works with one or two hard disks - 5 to 64 meg
- Menu installed, no software to assemble
- Complete hardware tests and error handling
- Automatic swap, for warm boots from hard drive
- Software drivers install above or below CP/M
- Allows custom partition sizes and mixed drive types
- Independent block and directory sizes on each partition
- Includes manual, format, test, park, and swap utilities

Winchester Connection Software only \$ 39.95

HDS Board with Winchester Connection Software . . . \$ 79.95

HDS Bare Board with software \$ 59.95

HDS Board, WD1002-05, and software \$245.00

Call or write for other pricing options

WD1002-05 HARD DISK CONTROLLER BOARD by Western Digital

- Standard ST506 drive interface
- Same size as standard 5 1/4" drive
- 40 pin interface to host computer
- WD2797 floppy disk controller interface on board
- Can control up to three hard drives
- Direct replacement for Kaypro 10 controller

WD1002-05 Controller Board \$185.00

Other Western Digital boards available

Prices subject to change without notice. VISA and Mastercard accepted. Include \$5.00 shipping and handling, \$7.50 for COD, UPS-Blue or RED Label additional according to weight. Please include your phone number with all correspondence.

Kaypro

KayPLUS ROM Set by MICROCode Consulting

Want more performance and flexibility from your Kaypro? With the KayPLUS ROM set you can have the advantages of a Kaypro 4 or 10, even on your Kaypro 2.

- Install up to four floppies and two hard drives
 - Boots from floppy or hard disk
 - Supports 96 TPI and 3 1/2" disk drives
 - Can use any ST506 type hard drive - 5 to 64 Meg
 - 32 character type-ahead keyboard buffer
 - Automatic screen blanking (not avail. on 83 series)
 - 12 disk formats built-in, unlimited configurable
 - Full automatic disk relogging with QP/M
 - Internal real-time clock support
 - No software assembly required
- Includes manual, format, configuration, diagnostics, sysgen, diskette customization utility, AND hard disk utilities. Available for '83 and '84 series Kaypros.

KayPLUS ROM Set, specify model \$ 69.95

KayPLUS ROM Set with QP/M \$125.00

Parts and accessories for the Kaypro

Kaypro 2X Real-time Clock parts kit \$ 29.00

Kaypro 2X Hard disk interface parts kit \$ 16.00

Kaypro 10 or '84 series Hard Disk host board \$ 49.00

Kaypro four drive floppy decoder board \$ 35.00

Complete parts and repair services available

Xerox 820

PLUS2 ROM and X120 Double Density Board by MICROCode Consulting and Emerald Microware

About had it with single density diskettes on your Xerox 820-1? Get unsurpassed versatility with our X120 Board and PLUS2 ROM package.

- Run up to four floppy disk drives at once
- Mix 8" and 5 1/4" at the same time
- Software compatible with Kaypro and Xerox 820
- Built in drivers for most serial and parallel printers
- Get mini-monitor functions and auto-boot capability
- 19 built in disk formats, including Xerox and Kaypro
- Includes custom disk format definition program
- Banked ROM BIOS for more space in your TPA
- Composite video adaptor on X120 board
- Runs 48 TPI diskettes on 96 TPI drives
- Supports real time clock from Z80-CTC
- Works on the Xerox 820-1 and Big Board I
- Both ROM and X120 board are required for operation

PLUS2 ROM Set and X120 Board A&T \$114.95

PLUS2 ROM Set and X120 Bare Board \$ 49.95

PLUS2 ROM Set only \$ 39.95

120 Bare Board only \$ 15.00

*** Special *** 2 boards for \$25, 5 for \$50

Other kits, parts, and packages available

Parts and accessories for the Xerox 820

Xerox 820-2 CPU Board - new \$ 75.00

Xerox 820-2 Floppy Controller board - new \$ 65.00

Xerox 820-2 CPU board w/ Floppy Controller \$125.00

Xerox 820-1 CPU board - new \$ 75.00

Xerox 820 complete high profile keyboard \$ 65.00

Xerox 820 bare high profile keyboard - new \$ 25.00

Xerox 820 5 1/4" drive cable \$ 9.00

Xerox internal video cable w/brightness control \$ 9.00

Xerox 820 power supply \$ 35.00

Power connector, specify board or cable \$ 2.50

Xerox parallel printer interface cable \$ 35.00

Dual Half Height 5 1/4" Disk Drives - DSDD,

in cabinet with standard Xerox cable \$265.00

Complete parts and repair services available



P.O. Box 1726, Beaverton, OR 97075



(503) 641-0347



30 day money back guarantee on all products.

Reader Service Number 10

IBM PC

CP/M, NorthStar, Macintosh, Apple II, MS-DOS, and PS/2 - Don't let incompatible diskette formats get you down, read them all with your PC.

UniForm-PC by MicroSolutions

How often have you wished you could use your CP/M diskettes on your PC? Now you can access your CP/M disks and programs on your MS-DOS computer just as you would a standard MS-DOS diskette. Once the UniForm driver is installed, you can use standard DOS commands and programs right on your original diskette without modifying or copying your files. UniForm-PC allows you to read, write, format, and copy diskettes from over 275 CP/M and MS-DOS computers on your PC, XT, or AT. With UniForm-PC and the Compaticard, you can use 5 1/4" high density, 96TPI, dual format 3 1/2" (720K/1.44 meg. - PS/2), and even 8" drives.

UniForm-PC by MicroSolutions \$ 64.95

Uniform for Kaypro and other machines \$ 64.95

Compaticard by MicroSolutions

Meet the Compaticard, THE universal disk drive controller card. This half card will let you run up to 16 disk drives (4 per Compaticard) on your PC or XT, including standard 360K, 96 TPI, high density (1.2 meg, dual speed), 8" single or double sided (SD or DD), and dual format 3 1/2" drives (720K/1.44 - PS/2). The combinations are almost unlimited. Comes with its own MS-DOS driver and format program for high density and 3 1/2" diskettes. Use it with UniForm-PC for maximum versatility. 8" adaptor and additional cabling available.

Compaticard Board \$169.95

Compaticard with UniFORM-PC *** Special *** \$225.00

Compaticard with UniFORM-PC & high density or

3 1/2" drive *** Special *** \$350.00

MatchPoint-PC by MicroSolutions

The MatchPoint-PC board for the PC/XT/AT works with your standard controller card to let you read and write to NorthStar hard sector and Apple II diskettes on your PC. INCLUDES a copy of the UniForm-PC program, as well as utilities to format disks, copy, delete, and view files on Apple DOS, PRODOS, and Apple CP/M diskettes.

MatchPoint-PC Board \$169.95

MatchMaker by MicroSolutions

Now you can copy your Macintosh diskettes right on your PC/XT/AT with the MatchMaker. Just plug your external Macintosh drive into the MatchMaker board and experience EASY access to your 3 1/2" Mac diskettes. Includes programs to read, write, initialize, and delete files on your single or double sided Mac diskettes.

MatchMaker Board \$139.95

MatchMaker w/External Mac Drive \$325.00

Frustrated because your PC can't speak CP/M?

UniDOS by Micro Solutions

Run CP/M programs on your PC? Of course. UniDOS is a memory resident program that can use the NEC V20 CPU chip to actually RUN your favorite 8080 programs. Use UniDOS with UniForm-PC, and automatically switch to CP/M mode as you log on your CP/M diskette. Switch to emulation mode to run Z80 code programs or for systems without a V20. UniDOS directly converts video and keyboard emulation for Kaypro, Xerox 820, Morrow, Osborne, VT100, and eight other displays. All standard CP/M system calls are supported. Note: The NEC V20 CPU is a fast, low power, CMOS replacement for the 8088 CPU chip that includes a full 8080 instruction set as well as the standard 8088 set. Systems using an 8086 may substitute a V30 chip.

UniDOS by MicroSolutions \$ 64.95

UniDOS w/UniForm and V20-8 chip \$135.00

UniDOS Z80 Coprocessor Board by MicroSolutions

This 8 Mhz. Z80H half-card will run your Z80 and 8080 code programs at LIGHTNING speed on your PC or AT. Functions just like the UniDOS program, except NO V20 or emulation mode is required to run your programs. Now includes UniForm-PC!

UniDOS Z80 Coprocessor Card \$169.95

shifted in, both start bits have shifted into bit heaven, and we have eight bits of data.

When the keyboard interrupt occurs, the 8255 parallel chip's Port B, bit 7 (PB7) is low. Since this bit connects to the LS322's active-low output enable, the 8255's Port A can read the keystroke from the LS322. After the keyboard interrupt handler (in the ROM BIOS) reads the keystroke, it sends a keyboard acknowledgment to the LS322 — again through PB7.

This temporary high disables the LS322's outputs and, after being inverted by an LS04, clears both the LS322 and the S74 in preparation for the next keystroke.

Parallel System Control

So, parallel bits PB0 and PB1 on the system parallel port control the speaker. What about the bits on the 8255 we haven't yet discussed?

PB2 was originally intended to disable the keyboard by zapping the serial data stream before it got to the LS322. But it's not implemented on the XT so you can use this bit for your own hardware projects.

PB3 controls which 4 out of the 8 system configuration switches (SW-1) the system is reading via the 8255's pins PC0-PC3. When PB3 is low, the system reads switches 1-4. When it's high the system can read 5-8. Of course, the switches are read right after power-up or a reset and then ignored, so PB3 is available for any other system control tasks (e.g., speed selection) after the initial boot.

PB6 feeds a 7407 hex buffer whose output is tied to KBDCLK. A low on the open-collector output of the 7407 ensures that nothing connected to it will be able to go high. Since the output of the 7407 connects to KBDCLK, a low on PB6 shorts the KBDCLK line to ground. Setting PB6 high allows KBDCLK to reach the shift register. So, making PB6 low does a very nice job of shutting off the keyboard.

RAM

Random access memory can be very simple or very complex.

Static RAM is very simple. You give it an address (any address), data, and tell it to read. Later (assuming you haven't turned the power off) you can give it the same address and it'll give you back the same data. Unfortunately there is no static RAM in an XT.

Dynamic RAM is not very simple. First, you have to divide the address into

two pieces. Then you feed those two pieces, very precisely one at a time, to the RAM. Then you supply data, and tell it to read.

Okay, that's one added step. Not so bad. But now let's read our data back. First address, second address, read. It's gone. It instantly (well, almost) forgot everything we told it. Dynamic RAM will remember data for up to 100 ms. Maybe. Unless you rewrite the data, or reread it (which interestingly enough, rewrites the data back into the cells), it goes away.

Refreshing is a special kind of rereading. So every, say, 2 ms you do a refresh. Right? Well, yes and no. Actually, you have to do 128 refreshes every 2 ms.

Someone has to put an address on the bus and say REFRESH! On the PC Jr., that someone was the processor. On the XT, as we've already mentioned, it's the DMA controller. Why 128? It turns out that most 64K and 256K dynamics are divided up into 128 groups of cells. You can read (refresh) a whole group with one command.

Dividing An Address In Half

Let's look at how the address is divided in half. The LS158 is called an 8 to 4 multiplexer. For each two inputs it has one output. The output is determined by one of the inputs. A select signal called ADDRSEL selects which of the two inputs will be connected to the output.

But let's back up a bit. The processor (or DMA controller) puts an address on the bus and punches either -XMEMR (external memory read) or -XMEMW (external memory write). Right away the -RAS line for the appropriate bank of RAM goes true (low). RAS stands for row address select. (The high-order address bits select the proper bank of memory.)

Then after a 60 ns delay (to let the first address soak in) the ADDRSEL line changes (to make the other half of the address available to the RAM). 40 ns later the -CAS line (column address select) for the same bank goes low telling the RAM to latch in the rest of the address.

Meanwhile the -WE (write enable) line is low if we're writing to memory or high if we're reading from memory.

Parity

Parity is simple. Sorta. We're dealing with 8 data bits and one parity bit. The parity bit is used to make sure the total number of "1" bits is odd.

Let's say you write the following byte into memory: 00000111. Okay, that has

three "1"s. Three is odd so the parity bit would be "0". 00000111 + 0 is 000001110. Now, with parity we've still got an odd number of "1"s. Of course if you'd written 10010011, the parity circuit would have noticed that you had an even number of "1"s and set the parity bit to "1".

Later, when you're reading memory, the parity circuit reads all nine bits. If there's an odd number of "1"s it's happy. Otherwise it drives the PCK (parity check) line high.

The heart of this circuit is the parity chip, an S280. Its "i" input (from the parity bit) is forced low during a write (so the 8 data bits determine whether its EVEN output goes high). If there is an even number of "1"s, then even goes to "1" and that "1" gets written into the parity bit. Thus the total number of "1"s written into memory is always odd.

During a read, XMEMR (not a typo — we have both active-low and active-high XMEMRs in this circuit) goes high letting the parity bit reach the S280's i input. A correct byte (and parity bit) will be odd, forcing the 280's "odd" pin high. Otherwise "odd" goes low and passes (inverted) through an LS27 where it gets clocked out as an error by -XMEMR.

Editor's Note:

Schematics are available from IBM. Or at least they used to be. I've heard from several folks that their copies of the XT technical reference package did not contain schematics.

In case you can't get schematics from IBM or don't have a spare \$100 kicking around, we're finishing up a single-sheet wall-sized schematic of the XT's main board. The schematic will be \$15, postpaid. (Sandy's entering my drawings into Autocad as I write this.)

How easy it'll be to use will depend a bit on how closely your system matches the original XT. The basic circuit should be nearly identical (otherwise the IBM ROM BIOS wouldn't run in the clones).

You will, however, find that board layout often differs a bit from clone to clone, with some appearing identical to the original. Major signal lines will nearly always connect to the same pins on the larger (24-40 pin) chips. However, the U numbers assigned to ICs will usually be different from the original, and individual gates within packages of gates will often be wired differently.

◆ ◆ ◆

Turn your IBM PC/XT/AT or compatibles into a powerful Lab./Engineering I/O station . . .

PC-LabCard SERIES

Data Acquisition & Control
Lab Automation
Engineering Tool, Education

**PCL-712 Multi-Lab (12 Bit)
A/D + D/A + DIO + Counter**

PCL-712 Multi-Lab is a low cost, single board solution for data acquisition and control, with all the following functions in one card:

- **Analog Input (A/D converter)**
 - 16 single-ended channels, 12 bit
 - Input range: +5V to -5V, 0 to 8.2V or 0 to +5V
- **Analog Output (D/A converter)**
 - 2 Channels, 12 bit
 - +5V ~ -5V/+8.2V ~ -8.2V, unipolar or bipolar
- **Digital I/O**
 - Input: 16 channels, TTL compatible
 - Output: 16 channels, TTL compatible
- **Counter**
 - 3 Channels of timer/counter

\$295⁰⁰

**PCL-748
IEEE-488 Interface Card**

- Turn your IBM PC/XT/AT to a powerful, easy-to-use IEEE-488 controller
- Powerful, easy-to-use software command set. Few arguments and simple initialization
- All software driver is in on-board firmware. Requires no additional disc software
- On-board RAM for working space. No system memory space is needed
- BASICA and BASIC compiler are supported as standard
- C and PASCAL language support packages available as options
- High data transfer speed

\$395⁰⁰

**PCL-714 Super-Lab (14 Bit)
A/D + D/A + DIO + Counter**

PCL-714 Super-Lab is a valuable single board solution to your data acquisition and control requirements.

- **Analog Input (A/D converter)**
 - 16 differential channels
 - 14 bit, 25,000 samples/sec
 - Input range: +5V to -5V, 0 to 8.2V or 0 to +5V
- **Analog Output (D/A converter)**
 - D/A Channels: 1 standard, 1 optional
 - 14 bit, ±5V or ± 8.2V range
- **Digital I/O**
 - Input: 16 channels, TTL compatible
 - Output: 16 channels, TTL compatible
- **Counter**
 - 3 channels of timer/counter

\$495⁰⁰

**PCL-725 Relay Actuator
and Isolated D/I Card**

PCL-725 offers 8 programmable SPDT relays for ON/OFF control or small power switching, and 8 channel opto-isolated inputs which can be used as digital input under noisy environment or floating potential.

- **Relay Actuator Outputs**
 - 8 channels, SPDT form C
 - Contact rating: 1 Amp/120 V
- **Isolated Digital Inputs**
 - 8 channels, isolated or non-isolated jumper selectable.
 - Type of opto-isolator: 4N25

\$240⁰⁰

**PCL-750
Prototype Development Card**

PCL-750 is an excellent alternative to pure breadboard card for prototype developing of PC add-on cards.

- Built-in basic circuit for a great deal of time saving
- Large breadboard area (3290 holes) for user circuits
- Independent memory and I/O address decoders included
- Address sizes of memory and I/O are jumper selectable
- All bus signals are buffered, marked and ready for use.

\$74⁰⁰

**PCL-720 Digital I/O
and Counter Card**

- **32 Digital Input Channels**
 - TTL compatible
 - Low loading: 0.2 mA at 0.4V
- **32 Digital Output Channels**
 - TTL compatible
 - Driving capacity: Sink 24 mA, Source 15 mA
- **Timer/Counter**
 - INTEL 8253 chip
 - Breadboard area for flexible user configuration

\$160⁰⁰

ALSO AVAILABLE:

- | | | | |
|--|--------------|-----------------------------------|------------|
| • Stepping Motor Control Card..... | \$395 | • Relay Output Board (16-CH)..... | 169 |
| • RS-422 Interface Card..... | 130 | • PC-LABDAS Level 1..... | 195 |
| • OPTO-Isolated D/I Board (16-CH)..... | 110 | • PC-LABDAS Level 1 & 2..... | 495 |
| • Wiring Terminal Board..... | 45 | • UNKELSCOPE Level 1..... | 125 |
| • Quick Connect Wiring Module..... | 125 | • UNKELSCOPE Level 1 & 2..... | 549 |

3 CONVENIENT LOCATIONS:

HSC Electronic Supply of Santa Rosa
6819 S. Santa Rosa Ave.
Cotati, CA
(707) 792-2277

HSC Electronic Supply
5549 Hemlock St.
Sacramento, CA
(916) 338-2545

3060 COPPER RD., SANTA CLARA, CA 95051

WE SHIP C.O.D.

MAIL ORDERS

Store Hours:
Mon-Fri 8:00-7:00
Saturday 9:00-5:00

Call NOW! (408)732-1573

TERMS: Minimum order \$10.00. California residents add 7% sales tax. Prepaid orders sent freight C.O.D. or call for charges. Shipping will be added to credit card and C.O.D. orders. Prepaid orders over \$100.00 use money order or certified check. Please do not send cash. Some items limited to stock on hand. Prices subject to change.

Simulating A Bus

A/D Conversion Through The Printer Port

Okay all you couch potatoes who've been following Bruce's hardware exploits. I know you've been dreaming of monitoring everything but the dog, but you haven't so much as stuck the soldering iron in the microwave.

Well, here's your chance to show your stuff. Break out the solder and warm up your irons, this is a full-blown, watch it on the screen (your specialty) project. First you'll build a bus and then you'll drive it over about 10 feet of ribbon. (Bet the neighbors'll respect you after they see you do that.)

Get ready! I'm going to show you how to turn the 8-bit output-only port into an input-output bus by modifying a \$21 parallel printer card.

Then we'll connect the bus to a nifty 8-bit, 11 channel analog-to-digital converter (A/D) chip, and measure the real world.

The wiring is simple, but the software is tricky because we have to make the printer card look like a microprocessor bus. To speed things up, I'll exploit some of the amazing in-line assembly code abilities of Turbo C.

Printer Board Modification

It will be helpful if you have issue #38 handy (*Micro C* Nov./Dec. 1988) — that's where I fully analyzed and diagrammed my cheap printer board. I got mine from MicroSphere, but I suspect most low-cost ones are similar. The board was manufactured in Taiwan by "dtk," #P11-109, copyright 1985.

If you're in doubt about compatibility, compare the schematics of your board to the schematics in issue #38. If they match, you're home free. (Okay, \$21 isn't quite free.)

The card's 8-bit data port is wired for output, but this project requires WRITING AND READING an 8-bit bus.

The A/D chip I'm using is designed to run on a microprocessor bus, so we're going to simulate the bus using the printer card. To do this, we need to make a simple modification (the one I mentioned in issue #38, page 31).

Find pin 1 on the LS374 (this pin is the "output enable," and is tied to ground). Take an X-acto knife and cut a small section out of the trace to this pin — remove the section so you're sure the connection has been broken (test it with a continuity checker).

Now find pin 7 of the LS174 (the output of an unused J-K flip-flop). Connect a small piece of wire between this pin and pin 1 of the LS374. It's best to do the connection on the "circuit side" (the side opposite the "component side"). I "tin" (get solder on) the ends of the wires first, then just touch the wire to the pin with the soldering iron.

Now we have a bi-directional 8-bit port. Figure 1 shows port locations and functions and gives the same information as issue #38, except for the addition of the "PBUS DIRECT" bit, which controls the direction of the bi-directional port we just created. (I'll call it the "printer bus").

I changed my board's I/O BASE address to 238 hex (described in issue #38) so it can't possibly conflict with normal printer boards.

Bus Limitations

To put multiple chips on the same data bus, the chips must be well behaved. In other words, when they aren't talking, they must release the bus so other chips can talk on it.

To release it, a chip puts its bus outputs into "high-impedance" or "Tri-state" (a trademark of National Semi). This means they no longer force the bus lines high or low — it's as if they'd disconnected themselves.

Both the printer card and the A/D chip can be tri-stated. When we want to read the printer bus, we set the printer

card to input so it's listening, not talking, and let the A/D talk on the bus. To write to the printer bus, we do the reverse.

Some chips can drive a bus harder than other chips. In our case, the printer card has lots of drive (since it's designed to send information over 12-foot cables). But the A/D chip doesn't have much drive at all — it's designed for a simple local bus.

This caused me problems. I was using an eight-foot ribbon cable between my DB-25 connector at the printer card end and my super-strip at the A/D end. The printer data port also has a capacitor on each line.

The printer port could easily drive the capacitors and the cable (so it works fine for "output only" projects), but the A/D chip couldn't. Not only did I have to cut off all the capacitors on the data port of the printer card (they're all in a row right next to the DB-25 connector), but I had to shorten my cable to two feet.

If for some reason you don't want to cut off your capacitors, or you want to put the A/D at a remote location, you need a bus transceiver chip (74LS245) to boost the A/D's output.

For cabling which is in the hundreds of feet, use the TTL-compatible variety of high-power CMOS: 74HCT245. You'll also need to replace the LS374 and LS244 on the printer board with the HCT variety and cut off the capacitors. Everyone else can just cut off their capacitors.

No Power Supply?

If you don't have a five-volt power supply handy, you can tap the power supply from the PC. Drill a little hole in the faceplate the DB-25 connector mounts on, run a wire through and solder it to the five-volt bus (+12 volt signals aren't used on this board). On my board, there's an electrolytic capacitor with a "+" sign next to it — that's a good place. With a little more dexterity, you could also wire 5V into one of the unused DB-25 lines.

The A/D Chip

Figure 2 shows the wiring connection between the DB-25 connector and the TLC532 A/D converter chip. I suggest ribbon cable with a clamp-on male DB-25 connector to get the signals to your breadboard (for circuit-building techniques, check my articles in previous issues of *Micro C*).

The chip is a complete package. I've done two other projects with A/D converters and the circuitry for this is the simplest, because the chip has a built-in multiplexer (so we can sample up to 11 input channels). Plus, it uses the system clock; and 6 of the lines can be used as analog or digital inputs.

You can buy the chip in single quantities for around \$8 from your local Marshall Electronics office (Note: minimum order is \$25).

This chip was originally manufactured by National and Motorola, but Texas Instruments (TI) increased the speed quite a bit. The TI data sheets don't contain quite as much information (National and Intel usually have the best data sheets).

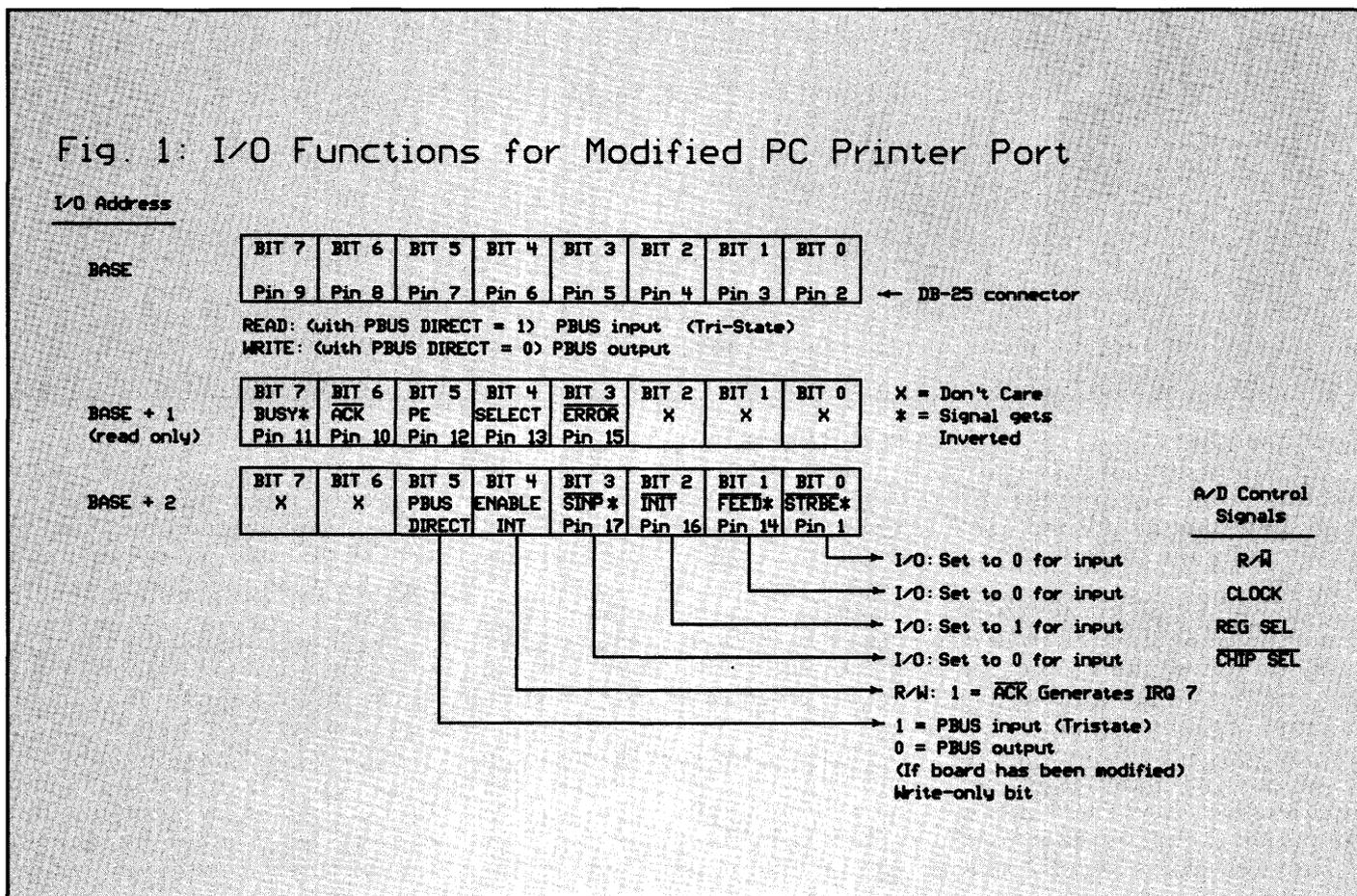
Voltage References

The chip has a positive and a negative voltage reference. The positive reference may be from 2.5 volts up to the supply voltage (5 volts). The negative reference

may be between ground and 2.5 volts. I've shown the most common configuration (Figure 2).

The A/D takes the analog input voltage and turns it into a number between 0 and 255 (0xff hex) which represents the voltage of the input relative to the negative and positive references. The connection shown allows analog input voltages between 0 and 5V.

If you want to use a smaller input range, or condition the analog input signals using an op-amp (see issue #35, April/May 1987), you can adjust the positive and negative references to within 1 volt of each other, as long as neither crosses the 2.5V mid point. If you



want better accuracy, you'll need to add a voltage reference (issue #34, Feb./March 1987).

(Editor's note: Issues #34 and #35 are now sold out at Micro C — see end of article for Bruce's alternative.)

Sample-and-hold

This chip has a "sample-and-hold," which takes a snapshot of an analog input signal. Thus the chip is not fooled by changes in the signal during conversion from analog to digital. The sampling and holding happen automatically when you request a conversion.

the connections.

There's also a reset line. *But*, we're out of output lines from the printer card! This worried me until I realized we'd never put data on the bus *and* have the chip select line high at the same time.

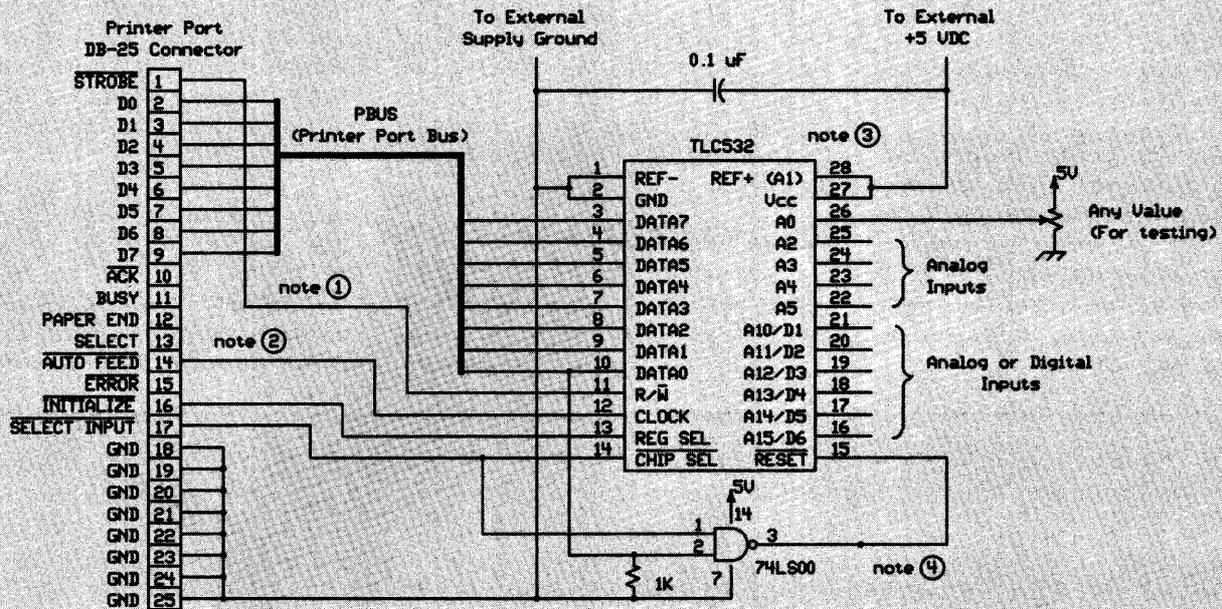
I took a data line, NANDed it with the chip select line, and used the result to drive the reset pin (the 1K resistor on the DATA0 line makes sure that line is low when it's tri-stated). The reset line must be low for three clock cycles after power up to put the chip in a known state; see the reset() function in Listing 1.

bus). The high byte is transferred on the first clock cycle, and the low byte on the second.

If you actually connect this to a processor bus, connect address line 1 (instead of 0) to the register select line. This way, the processor will think it's writing two successive memory locations when it's actually making two writes to one location (which is what we want).

The functions digital_data() and conversion() in Listing 1 demonstrate how to do the bus reads and writes.

Fig. 2: A/D Converter Interface to Modified Printer Card



- Notes:
- ① This line has a capacitor on it, so we can't use it where rise/fall times are critical
 - ② All inputs are TTL-compatible, except for the clock, which is CMOS. AUTO FEED has a pull-up on its output, so it works OK.
 - ③ You can do a test conversion on pin 28 (A1). It's equal to the reference voltage (0xff).
 - ④ I ran out of output lines from the printer port, so I had to simulate one (see text). The resistor keeps DATA0 pulled to a '0' when the bus is tri-stated.

Simulating A Bus

This A/D is designed to be directly interfaced to a microprocessor bus. We've got the data bus coordinated with our new printer bus, but we need to add control lines.

The printer card has four more output lines which connect to the read/write pin (to tell the chip whether it's being read from or written to), the register select pin (to choose internal register 0 or 1), the chip select pin (to tell the chip "yes, we are talking to you") and the clock pin, which is used not only for the logic interface, but also to drive the analog-to-digital conversion process. Figure 2 shows

16-bit Transfers

Figure 3 shows the registers for the A/D chip. We write to the control register to select the analog input channel and to tell the chip to start a conversion. We can read two result registers (differentiated by the register select line) — one for the analog result; and one for the digital result and the address of the pin we last performed an A/D conversion on.

Surprise! These are 16-bit registers! To transfer data, we must do the equivalent of a 16-bit transfer from an 8-bit microprocessor (or, in the case of the 8088, a 16-bit processor with an 8-bit

Software

Listing 1 shows all the functions necessary to set up and drive the converter. A driver program (See Listing 2) turns off the cursor and repeatedly displays the digital and analog values on the screen.

There's also a makefile (See Listing 3) so you can build the whole thing, including the in-line assembly (if you own MASM), just by typing "make."

I've included debugging code which can be turned on by putting "#define DEBUG_FLAG" at the beginning of the file. After each I/O step, the program will wait for you to press a key. You can

check all the outputs with an inexpensive logic probe — that's how I debugged the hardware.

For graphics details, check issue #39 (Jan./Feb. 1988) — Larry Fogg shows you how to put dots on a Hercules graphics screen, and Gary Entsminger does dots on an EGA. So you can put it all together and create an oscilloscope!

Next Time

Next issue I'll show you how to build your own PC-bus interface card using a plug-in board covered with super-strips.

By using the cable configuration I showed last issue to connect the two machines, you can start up Procomm on the second machine and log into the first one. If you want to change directories and hunt for files, you can start a DOS shell on the first machine from the second. I was moving files back and forth at 19.2 Kbaud. It wasn't a local-area network, but it didn't cost a fortune, either.

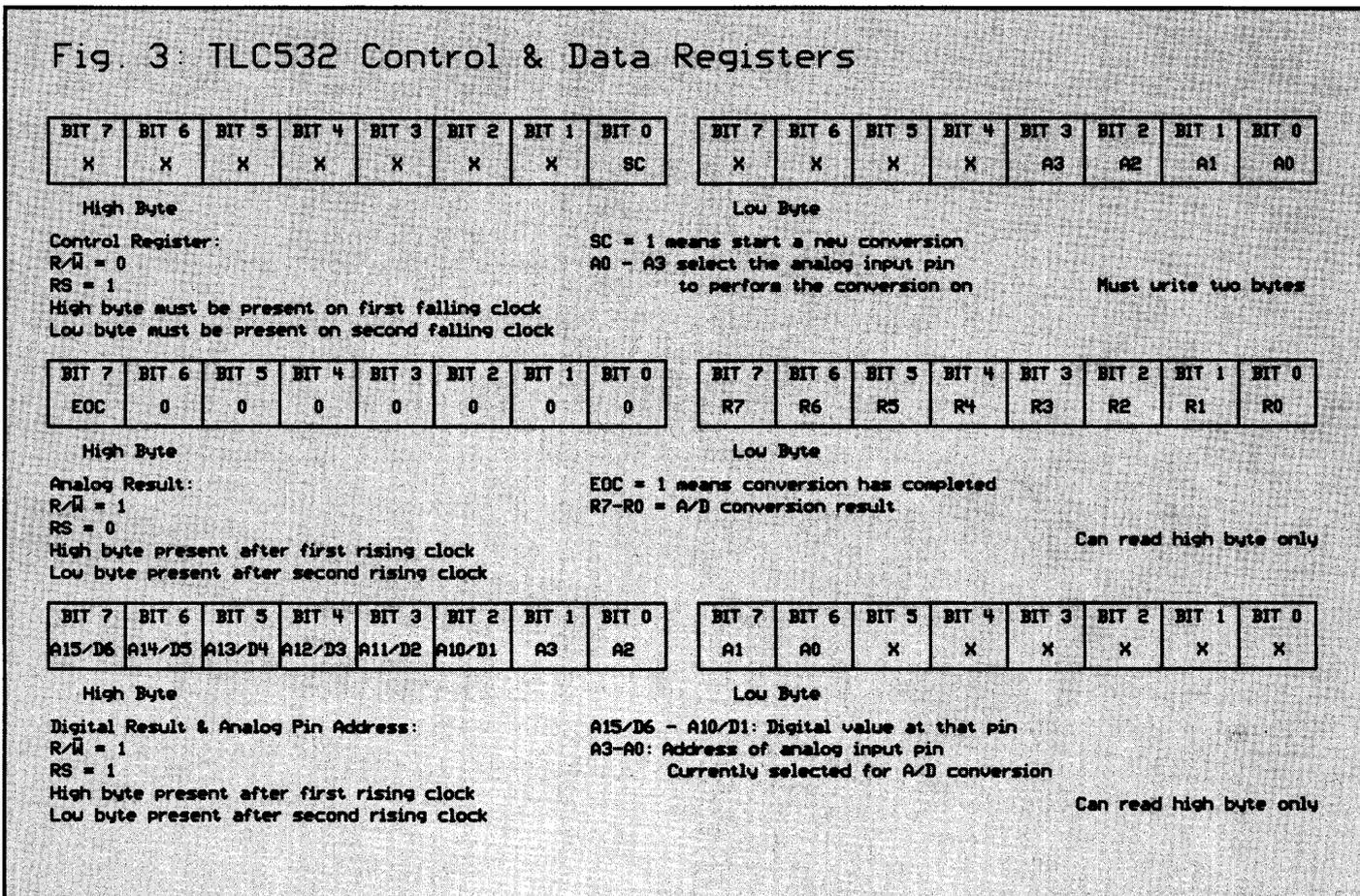
C++

The C++ charge continues! The best news is the forthcoming \$99 (!!) compiler

so far (mouse support, for instance).

Advantage C++ has just released version 1.2, with support for Microsoft Windows, OS/2, and source-level debugging with Microsoft CodeView. Sounds great, though I haven't tested it.

In the minicomputer world, the GNU project (aka the Free Software Foundation: FSF) has version 1.18.2 of their native-code C++ and source-level debugger out and being tested by everyone (this is a real community project). It was developed by Michael Tiemann (a real nice guy, and SMART!) on a Sun 3, but



And Furthermore

Notes and information about various projects and related subjects have been piling up, so here are a few notes to bring you up-to-date.

I mentioned in the last issue that I use communications programs to transfer files back and forth between my two computers. Since then, I've discovered a particularly nice system: Procomm has something called "host mode"; which you can use to set one machine up as a simple bulletin board system (BBS). (Note: Procomm is still shareware but Datastorm's new Procomm Plus is regular buyware.)

from Zortech which generates native code. This means you only need one compiler.

With translators, you had to buy a regular C compiler (usually Microsoft) in addition, and get the two to work together.

Not only is translating a hassle, but you lose debugging information in the process. Zortech C++ will work with CodeView. With a price like that, I'm considering writing more of my project code in C++. The compiler was written by Walter Bright, who created the Datalight C compiler. I'm helping to test a beta version, and I like what I've seen

people are porting it to other machines (but probably never a segmented-architecture machine).

Oregon Software sells a native-code C++ for the Sun. If you don't trust free software, wait a bit; I'll be trying it out. I especially look forward to their debugger.

Another C++ article is in the works for the object-oriented programming issue (*Micro C #44*); it will be a programming project so you can see "how to do it." And I'll be talking about C++ at SOG VII.

My e-mail Address

For those of you on a mail network, here's how you can reach me:

arpanet:

eckel@sperm.ocean.washington.edu

UUCP:

uw-beaver!sperm.ocean.washington.edu!ec

kel

bitnet:

eckel@sperm.ocean.washington.edu@UWAVM

I try to answer all my mail promptly, but bits are inevitably faster than paper. (Even though the addresses are longer.)

And Finally — A Book

I often refer to introductory material in previous articles; this can be a problem if you're a new subscriber. The solution: I've put together a book of all my articles (including some not published in *Micro C*, and some material not published anywhere) and a disk with all the source code from the articles.

To get it, send a check for \$30 (plus 8% if you live in Washington State) to Eisys, 1009 N. 36th Street, Seattle, WA 98103. Satisfaction Guaranteed or your money back.



Listing 1 - Functions For TLC532AIN

```
/* Functions for the TI TLC532AIN
11-channel A/D converter. The chip is reset,
the TTL-level digital inputs at pins 16-21 are
read, and the 8-bit analog values at pins 16-25
are read. A sample driver can be found on the
Micro C BBS (DRIVER.C). Bruce Eckel, Eisys
Consulting, 1988 */
/* Since this file contains in-line assembly, it
must be compiled with the command:
"tcc -c -B adc.c" to create a ".obj" file. A
makefile is available on the BBS to automate all
this using Turbo C's "make" */

#undef DEBUG_FLAG /* define this for debugging */

#define BASE 0x238 /* address of printer card,
set by jumpers */

#define DATA BASE
#define CONTROL (BASE+2)
/* macro to put a '1' in a binary position: */
#define BIT(x) (1 < x)
/* set PBUS to input/tristate: */
#define PBUS_IN BIT(5)
#define PBUS_OUT 0
#define READ 0 /* read/write line high */
#define WRITE BIT(0) /* read/write line low
(signal inverted) */
#define CLOCK_HIGH 0 /* signal inverted */
```

The Pocket PC

Organizer II offers:

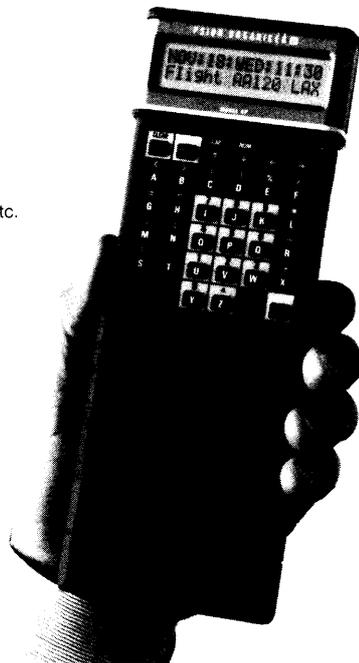
- Lotus 1-2-3 compatibility
- Word Processing
- Data Base
- Finance
- Bar Code Readers
- RS-232 Interface
- 32K Expandable to 320K Memory
- Plugs into office systems
- Excellent for Lawyers, Doctors, Salesmen, etc.
- Weight 9 oz.

Psion Organizer II XP	249.95
16K Datapak	29.95
32 Datapak	49.95
64K Datapak	89.95
128 Datapak	149.96
32K Ram Pak	89.95
Finance Pak	49.95
Maths Pak	49.95
Pocket Spreadsheet	79.95
Spelling Checker	49.95
Word Processor	49.95
Data Base	59.95
RS232 Lead (Comms-Link)	99.95
AC Adaptor	19.95

\$3.00 shipping and handling.
3% surcharge on credit card orders.

Sharp's, Inc.

Rt. 10, Box 459
Mechanicsville, VA 23111
(804) 746-1664 9am - 4pm



**DISK FORMAT
CONVERSION**

XenoCopy-PC

PC-DOS program
lets your PC

\$79.95 + \$5.00 S/H
Sales Tax if CA.
 

READ / WRITE /

FORMAT / DUPLICATE

Disks from over 300 other micros

Upgrades available from previous versions
for only \$25.00 Call for Authorization

To Order Contact:

XENOSOFT™

2210 SIXTH ST. BERKELEY, CA. 94710

(415) 644-9366

Reader Service Number 39

```

#define CLOCK_LOW BIT(1)
#define REGISTER_0 0 /* signal not inverted */
#define REGISTER_1 BIT(2)
#define SELECT BIT(3) /* signal inverted */
#define DE_SELECT 0

/* global control byte to hold the status of
BASE+2, so we can change one pin at a time: */
unsigned char ctl_val;

/* prints a byte as ones and zeroes (for
debugging with a logic probe) */
void print_binary(unsigned char c) {
    int i;
    for (i = 7; i >= 0; i--)
        /* note "ternary" if-then-else: */
        printf("%c", c & (1 < i)? '1': '0');
}

#ifdef DEBUG_FLAG
void DEBUG(char * TEXT) {
    printf(TEXT);
    printf("  ctl_val = ");
    print_binary(ctl_val); printf("\n");
    getch();
    /* so you can stop after every step;
    press a key to continue */
}
#else DEBUG_FLAG
/* do nothing if we aren't debugging
(generates a compiler warning) */
void DEBUG(char * TEXT) {}
#endif DEBUG_FLAG

/* Note: when changing a bit without affecting
the others, the non-zero version of the signal
must be used. It is ORed to make the bit go
high, and it's bitwise inverse (-) is ANDed to
make the bit go low. */
#define COMMAND(VALUE) \
    (outportb(CONTROL, (VALUE))); DEBUG("command")
#define DATA_OUT(VALUE) \
    (outportb(DATA, (VALUE))); DEBUG("data out")
#define DATA_IN (inportb(DATA))
#define CLOCK_DOWN \
    COMMAND(ctl_val |= CLOCK_LOW); \
    DEBUG("clock low")
#define CLOCK_UP \
    COMMAND(ctl_val &= ~CLOCK_LOW); \
    DEBUG("clock hi")
#define BUS_IDLE \
    COMMAND(ctl_val=PBUS_IN|CLOCK_LOW|DE_SELECT);
/* BUS_IDLE: clock should always be left in the
"low" state. PBUS should always be left
tri-stated (input). Chip select should always
be high (not selected) */
#define CLEAR BUS_IDLE; CLOCK_UP; CLOCK_DOWN;
/* 'CLEAR' resets the tlc532's internal byte
pointer so it points to the high-byte register.
This occurs after the chip gets a full clock
cycle when it is de-selected. */

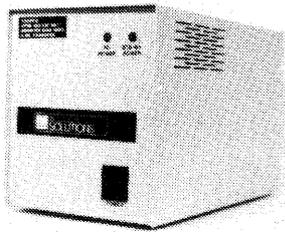
void reset_adc(void) {
    /* call this once on power-up */
    int i;
    DATA_OUT(0xff); /* raise DATA lines */
    /* lower reset line & enable PBUS output: */
    COMMAND (ctl_val = PBUS_OUT | CLOCK_LOW |
        DE_SELECT); DEBUG("reset low");
    /* now we have to raise and lower the clock
    3 times (kind of like a magic spell) */
    for (i = 0; i < 3; i++)
        { CLOCK_UP; CLOCK_DOWN; }
    /* Now return the bus to the idle state */
    BUS_IDLE;
}

unsigned int digital_values() {
    /* read the values from the digital pins */
    unsigned int result;
    COMMAND (ctl_val = PBUS_IN | READ | CLOCK_LOW
        | REGISTER_1 | SELECT );
    /* raise the clock & read the high byte */
    CLOCK_UP;
    /* shift it into the high byte: */
    result = DATA_IN < 8; DEBUG("data in");
    /* cycle the clock and read the low byte */
    CLOCK_DOWN; CLOCK_UP;
    /* don't disturb the high byte: */
    result |= DATA_IN;
    /* reset the bus & internal byte pointer */
    CLEAR;
    return result;
}

unsigned int conversion(int input_line) {
    unsigned int result;
    /* number of clocks for a conversion: */
    int clock_counts = 28;
    COMMAND (ctl_val = PBUS_OUT | WRITE |
        CLOCK_LOW | REGISTER_1 | SELECT );
    CLOCK_UP;
    /* set bit 0 of high byte for conversion: */
    DATA_OUT(1);
    /* Clock in the high byte: */
    CLOCK_DOWN; CLOCK_UP;
    /* select pin to convert: */
    DATA_OUT(input_line & 0xf);
    BUS_IDLE;
    /* So byte pointer is reset during
    conversion. Clock is low */
    /* If you don't own MASM, replace the inline
    code with the following: */
    /* while (clock_counts--)
        { CLOCK_UP; CLOCK_DOWN; } */
    /* Here's the in-line assembly to speed up the
    clocking. If you want the conversion()
    function to run as fast as possible, write
    the whole thing in assembly. */
    /* register DX holds the port number: */
    _DX = CONTROL;
    /* register AL holds the output data: */
    _AL = ctl_val;
    /* register CX for decrement/looping */
    _CX = clock_counts;
    do_clocks:

```

UNINTERRUPTABLE POWER SOURCE



MICRO

SOLUTIONS protects your equipment and your data from **power outages** and **brownouts**. Our power systems provide the **fastest switching speed** in the industry (2 ms ± 1).

EMI/RFI filtering and surge/spike protection all in one affordable unit. 1 year warranty on all units. Available in a size to suit your needs.

200 watts	\$290.00
350 watts	\$360.00
550 watts	\$410.00
800 watts	\$710.00
1000 watts	\$810.00

Includes shipping to your door in the continental U.S.. As specialists in overseas systems, we can supply 220 volt units. Call or write for details.

SOFTWARE SPECIAL



See what qualifies as GREAT SOFTWARE! ACT! does for people what 1 2 3 does for numbers. ACT! helps you manage your contacts with the important people in your life.

Call today for more information - or send a \$10 (refundable) deposit to receive a copy of the ACT! video demo tape. Your deposit will be refunded - whether you purchase ACT! or not - when the video is returned in good condition.



WE SHIP WORLDWIDE

C.O.D.



Dealers Supported

Drawer B Riner, VA 24149

1-800-323-4829 (703) 382-6624

Call 24 hours - 7 days a week

Reader Service Number 24

```

/* raise the clock line (bit 1):*/
asm or  al, 10b;
/* output the control value:*/
asm out dx,al;
/* lower the clock line: */
asm and al, 11111101b;
/* output the control value: */
asm out dx,al;
/* decrement cx and loop if not zero: */
asm loop dx_clocks

COMMAND (ctl_val = PBUS_IN | READ | CLOCK_LOW
          | REGISTER_0 | SELECT );
/* raise the clock line & read high byte */
CLOCK_UP;
/* shift it into the high byte: */
result = DATA_IN < 8; DEBUG("hi data in");
/* cycle the clock and read the low byte */
CLOCK_DOWN; CLOCK_UP;
result |= DATA_IN; DEBUG("low data in");
/* reset bus and internal byte pointer */
CLEAR;
return result;
}

```

END OF LISTING

Listing 2 - Driver For A/D Converter.

```

/* Driver for the TI TLC532AIN A/D converter functions.
Digital inputs are constantly displayed. Pressing the
a channel number (0-5, a-f) displays that channel. */

#include <dos.h>
#define VIDEO 0x10          /* video interrupt */
#define VC peek(0x40, 0x63) /* 6845 base register */
#define BIT(x) (1 << x)    /* macro to set a bit */

/* Declarations for TLC532AIN functions in file ADC.C */
void reset_adc(void); /* call this once on power-up */
unsigned int digital_values(void); /* read digital values */
unsigned int conversion(int input_line); /* perform a
conversion on a pin */
/* Move cursor to a place on the screen */
void gotoxy(int x, int y) { /* from Turbo C ref manual */
union REGS regs;
regs.h.ah = 2; /* set cursor position */
regs.h.dh = y;
regs.h.dl = x;
regs.h.bh = 0; /* video page 0 */
int86(VIDEO, &regs, &regs);
}

/* To understand these, see Fogg's 6845 article in issue #39 */
/* Turn cursor flashing off and on */
void cursor_off() { outportb(VC,10); outportb(VC+1,14); }
void cursor_on() { outportb(VC,10); outportb(VC+1,11); }

main()
{
int i = 1; /* sample ch 1 first, it's the power supply */
unsigned int digital, analog;
reset_adc();
cursor_off(); /* makes display much easier to look at */
while (1){
gotoxy(0,1);
printf("sampling channel %d ",i); conversion(i);
while (!kbhit()) {
gotoxy(0,2);
printf("address: %d \n",
(digital_values() >> 6) & 0xf);
analog = conversion(i);
printf(" EOC = %d ", analog & BIT(15) ? 1:0);
}
}
}

```

```

printf(" conversion = 0x%x\n", analog & 0xff);
printf("digital values: ");
print_binary( digital_values() >> 10);
}
i = getch();          /* get the character from kbhit() */
if (i == 27) break;   /* quit on an ESC */
if (i <= '9' && i >= '0') i -= '0'; /* convert ascii */
if (i <= 'f' && i >= 'a') i -= 'a' - 10; /*a-f -> 10-15*/
i &= 0xf;             /* other chars get masked */
}
cursor_on(); /* turn cursor back on when finished */
}
END OF LISTING

```

Listing 3 - Make File For Compiling Bruce's Code

```

# A makefile for use with Turbo C's "make" program. This will
# compile the code (which includes in-line assembly) from Bruce
# Eckel's A/D converter article in issue #42 of Micro C.
# All you have to do is type "make".

# where to look for the library files:
TCLIB = c:\turboc

# The target file is adc.exe, and it depends on the .obj files.
# Second line tells how to make the target from the .obj files

adc.exe : adc.obj driver.obj
        tcc -oadc.exe -L$(TCLIB) adc.obj driver.obj

# .obj files depend on .c files. In the case of adc.c, we
# need to invoke the compiler with the in-line assembly option

adc.obj : adc.c
        tcc -c -B adc.c

driver.obj : driver.c
        tcc -c -IS$(TCLIB) driver.c
END OF LISTING

```



Rafting Barbeque Computer tech talk

July 14 - July 16
Bend, Oregon

Register by July 1st!

See you there!!

At SOG ...

Desktop Publishing Seminar:



This is a one-day, hands-on seminar that should give you a good idea what it takes to get into PC desktop publishing in a big way, or a small way.

We'll show you how Publisher's Paintbrush works with a scanner to digitize and edit pixel images.

You'll see Designer in action. This is a vector drawing package so you'll get maximum quality from your laser printer or typesetter. (Especially when you're resizing an image substantially.) But, working with vectors is quite different from working with pixels and there'll be times when the ease of working with Paintbrush will make it the best choice.

We'll talk about text editors, covering some of the reasons you would (and wouldn't) want to use a formatting text editor like Wordstar or Wordperfect for creating your original text files (vs using straight ASCII).

We'll talk about the little things that make a big difference in final quality. Typesetters vs laser printers. Screened photos vs scanned photos. Quality printing vs quick printing. It's quite a list.

Finally, we'll take the images and the text that we've created right before your eyes and we'll combine them into a single document. (We'll show you how to do it with Pagemaker and with Ventura Publisher so you can see the differences.)

Then you'll join a smaller group for a hands-on design session, and we'll use the designs as a starting point for a quick discussion on graphics.

Handouts will include our hardware and software recommendations for putting together desktop systems. We'll include suggestions for those of you who are: creating ads, producing handbills, small newsletters, newspapers, small or large technical manuals, and magazines. Join the fun!

The West Coast Computer Swap Meet

Or: Where To Get Your Cheap Rolexes

The last time Micro Cornucopia exhibited at the West Coast Computer Faire, our booth was in the men's room on the second mezzanine (we shared a stall with FOG). That was six years ago.

This year we had a ten by ten on the main floor (immediately in front of the johns) so I was ready to rub shoulders with Microsoft, Borland, and Intel. Unfortunately, they weren't there. Of course, they'll be at SOG VII (couldn't resist a dig), but that's a different story.

It used to be the West Coast Computer Faire. The name hasn't changed, at least not officially, but this year the Faire part was pretty well hidden among the swap tables.

There were a few certifiably techy types: Trilogy. Programmer's Journal. RBBS-PC. Dr. Dobb's. Micro/Systems Journal. Marshall Magee (with his shareware menu prog.). Merrill Lynch (oops, they belong in the next section). Micro Cornucopia (Micro who?). Paul Mace...

And, the technical types there were pretty well disguised as otherwise sane hackers who donned dark suits and ties. The Micro C booth was probably the only one entirely unsuited for the occasion. (Think about it, it'll come to you.)

The first three days Gary Entsminger, Larry Fogg, and Bruce Eckel played the three musketeers while I wandered the show floor looking for likely techies (ill fitting shoulders, uncomfortable collars...).

It turned out that those three were so out of control Thursday, Friday, and Saturday, that by Sunday (when they left me to man the booth alone) people were coming by just to catch the act. (You can come to SOG and catch them at their best.)

The Swap Meet

The rest of the Faire was a swap meet. I'd be the last to complain about a swap meet (you wouldn't believe what I paid for this genuine diamond. It was even manufactured in the U.S.!). But I don't confuse swap meets with Faires.

PC-SIG

PC-SIG was showing its new marketing idea. It's packaging shareware just like commercial software, complete with color packages hanging on racks. I understand that Sears will be displaying the racks. Just plunk down your Sears card, and you can purchase jeans, wrenches, tires, and PC-SIG's software.

(Hold on a minute, this isn't PC-SIG's software, this is shareware. Purchasers haven't paid for the program, just for the rack, the packaging, and the paper bag. Will those people pony up another \$50 if they like it? Would they send Calvin Klein a \$10 sharewear fee if the jeans fit?)

Classy Vendors

Meanwhile busy vendors sold watches ("get your watches here, best price in the show") and jewelry ("our gold plating is real gold plating"). Hey, that's almost as outlandish as airline ads in BYTE. (You say there *are* airline ads in BYTE???)

You could purchase encyclopedias, "Fill out a card for a chance on a set of books." (And a guarantee on a salesman.) Or you could purchase stocks. (I told you Merrill Lynch belonged down here. Way down here.)

There must have been 20 booths peddling cheap floppy disks (27 cents each for 5 1/4", \$1.10 for 3 1/2"), 30 showing XT/AT/386 boards, another 30 selling disks of shareware (PC-SIG was the highest priced of the bunch), and a dozen others moving heavily discounted commercial software.

"Save 90% on the latest and greatest! Don't you dare pass up this booth. These

are deals you'll never see again. Sealed boxes. The freshest programs."

Freshest, my eye, they were old versions or remaindered (no longer marketed or supported) titles. Still, I bought some. Let's see, three Infocom games — Xor's Thoth (action list manager) — Blu Chip, a portfolio manager — and something else. Here it is, a centipede that crawls around the screen eating screaming software vendors.

"Three packages for \$25, take your pick. Any of the bins."

Bins? We're talking plastic wash tubs. What an end for programmers' dreams of fame and fortune.

I remember Bend's 4th of July craft faire. During the first few years it was craftspeople showing their wares and families selling home-baked specialties. But, as the crowds grew, the hand crafted wares were replaced by the black felt merchants (specializing in tigers), the assembly line basket brokers, and the driftwood clock manufacturers.

Oh, I enjoy grape snow cones, all right, but I miss the year-to-year unpredictability of Grandma Marti's brown-guey supreme.

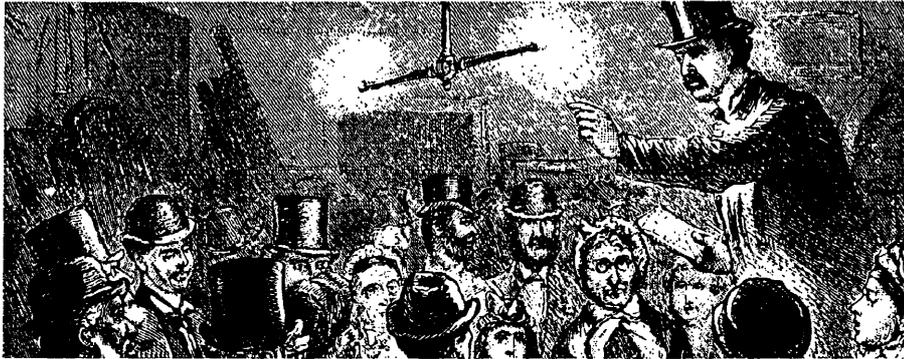
Another View

Most of the exhibitors I spoke with were disappointed with this year's Faire. They were disappointed with the attendance and with the paucity of new products. So, on a lark I called Jim Warren, the founder of the show:

"It's different than the old days. It's hard to tell if it would have changed, anyway. I enjoyed (this year's) faire. I've enjoyed them all a lot more since I sold it. Now I can attend the conference sessions and see all this shit.

"However, I thought the crowds seemed awfully light. They're (the Interface Group) saying the count was higher than ever before, but I didn't see them.

"I didn't notice all that much dif-



ference in the exhibits, but it did bother me that there weren't many micro booths. Those booths always had the flakiest, shakiest, weirdest collection of really exciting displays.

"Those booths attracted the best crowds, had the best deals, the best technology, the newest products, and you were dealing with the creators.

"Without the creators it's possible to kill the Faire. They are the people who created the Faire."

Another thing Jim and I talked about was Greyhound Exhibitions, the show's, uh, exhibitionist. Greyhound told me they would charge me \$70 if I had the Micro C office send me a small package, Federal Express. That's \$70 just to accept the package. But, it was the only way I could get it delivered at the Faire.

Exhibitors couldn't use anything with wheels to move materials to their booths. Everything had to be hand-carried, and Greyhound enforced it, at least until the show was over. Setup (without hand-trucks) took a day and a half. Teardown (with the little beauties) took two hours.

"I've thought about starting another show, but it's a royal hassle. I don't want the responsibility," Jim said.

"I was sorry to see the light crowd, sorry to hear the complaints from exhibitors, but my session was really fun."

Note: Jim Warren is coming to SOG VII. Join him for rafting, eating, and long evening

discussions about life, the universe, and everything, in Bend, July 14-16.

Discoveries

I wandered the show floor for three full days. I found that it's often easier to discuss the answer to the age old questions (see above) when exhibitors aren't defending themselves from heavyweight questions:

"Do you supply MS-DOS programs on 3 1/2" so I can run them on my Mac?"

"Are clones really compatible?"

"If I plug this 8088 into the Z80 socket, can I run PC software on my Kaypro?"

"Is this real?"

Copy II PC

While the crowds were pretty thin (no heavyweight programmers), I sidled up to one of those discount booths and purchased a Copy II PC Deluxe Option Board. Of course, you're thinking I got it so I could duplicate (almost) every piece of copy-protected software in the office.

For the record, let me state that's a false and malicious thought. (Shame on you!) I've had one of Central Point's earlier boards for a couple of years and I've already duplicated (almost) every piece of copy protected software in the office. Companies give me software to review and then they copy protect it to keep me

from testing it on different systems. Boo.

In addition to copying the uncopyable, this new board was supposed to read, write, and format 400K and 800K Macintosh disks. (Macintosh disks?) The instructions said that all I needed was the board, a standard IBM 720K 3 1/2" drive, and a clone to plug the whole mess into.

Sure folks. There's more to reading and writing Mac disks than understanding the data format. After all, Apple increases spindle speed as the head approaches the center so the velocity of the media under the head doesn't drop so drastically. But the option board is dealing with a single-speed drive, so it's got to understand more than a single data rate. Can it really do that?

Yep.

Their little half-length board works like gangbusters. It reads Mac files, writes Mac files, creates directories (I mean folders), removes directories, displays directories, diskcopies from one Mac disk to another, and copies files between Mac and MS-DOS disks.

Of course the files may need a little futzing. You have to turn the <CR>'s in the Mac text files into <CR><LF>'s for the PC editors (and vice versa). But Vedit can handle that, no sweat.

They are offering upgrades for those who already own the older boards. Send in your older board, or your master software disk, or your registration card and they'll send you one of the new boards for \$59 plus \$5 s/h. Or they'll sell you one outright for \$159 plus \$5 s/h. (Discount price was \$99.95.)

Copy II PC Deluxe Option Board
Central Point Software
9700 SW Capitol Highway
Portland, OR 97219
(503) 244-5782



Working With PCX Files

Inside the Most Popular Graphics File Format

This article began several months ago with a conversation. Sam was looking for information on the PCX graphics file format. I was looking for an article on PCX. Well, he figured out the format and we've got an article.

The Picture file format used by ZSoft for their PC Paintbrush program has become one of the more popular formats around. I took an interest in it because I needed to get diagrams into Ventura Publisher.

The .PCX format turned out to be easy to work with. The files can be loaded and written quickly, using a set of simple functions. Screen update is fast, and the screen can easily be panned around a large picture. Drawing operations for multi-color formats aren't as fast, but that's due to the data structure that I've chosen.

I'd like to start by describing PCX pictures in general terms. The Header and Color palette are complicated enough to work through in detail. I'll finish with a description of the modules that I've written to generate, write, read, and display PCX pictures. You'll probably want to make many improvements to them, but they should get a PCX-related project off the ground for you.

The Basic Structure

A PCX file consists of a 128-byte header, followed by one or more compressed picture planes. In memory, each plane is basically a bit map, each bit usually corresponding to a pixel on the screen. The number of bits per pixel, pixels per byte, bytes per line, lines per plane and planes per picture are all stored in the header.

On an EGA board, there are four planes (see Figure 1). Bit (X,Y) in each plane corresponds to pixel (X,Y) on the screen. The binary value of the four (X,Y)

bits in each plane is used to index into the color palette. On the CGA & HERC boards, only 1 plane is used. For the 720x348 HERC and 640x200 CGA modes, each bit in the plane corresponds to a single pixel. Using the 320x200 by 4 color CGA mode, there is again only one plane, but each pixel is made up of two adjacent bits.

It's possible to make a 1000x1000 by 2 color drawing which can be displayed on a CGA board 640x200 dots at a time, then show it on a Hercules board 720x348 dots at a time, but a picture isn't always so portable. A 320x200 CGA picture has to be translated before it can be shown on another type of monitor.

Drawing the picture on the screen is easy — all that's needed is a block move of each line of the picture to VRAM (video RAM). Of course there has to be some interpretation of the header information, and the pictures don't look quite the same from one type of video board to another because of differences in dot width and height.

The Header

The structure for the header looks like this:

```
typedef struct (  
    unsigned char red, green, blue;  
) TRIPLET;
```

```
typedef struct (  
    char maker,  
    version,  
    code,  
    bpp;  
    int x1, y1, x2, y2,  
        hres, vres;  
    TRIPLET triple[16];  
    char vmode,  
    nplanes;  
    int bpl;  
    char __unused[128-68];  
) PCXHDR;
```

The fields maker, version, code and vmode have fixed values — 10, 5, 1, & 0. They represent the code for the program that generated the picture, its version number and the type of compression that was applied to the picture planes; vmode is currently ignored; and __unused is reserved for future expansion.

hres and vres represent the number of horizontal and vertical dots on the video board. nplanes is the number of picture planes, bpp is the number of bits per pixel and bpl is the number of bytes to store a horizontal line of the picture. (x1,y1) and (x2,y2) refer to the upper left and lower right corners of the picture. For a standard 640x350x16 EGA picture, we would get the following: hres=640, vres=350, x1=0, y1=0, x2=639, y2=349, nplanes=4, bpp=1 and bpl=80.

PC Paintbrush takes these numbers very seriously. If you create a graphic file with Paintbrush installed for a HERC board and then reinstall Paintbrush for an EGA board, it will no longer recognize the graphic file.

That's a bit picky considering that a HERC picture is black and white, and an EGA board supports those colors.

So, if you plan to use the pictures that you generate with a commercial graphics program, be sure to specify one of the common video boards.

To enlarge or reduce a picture, you only need to change (x2, y2) and bpl. The easiest way to add colors is to add planes. Although the software that I've written only supports a few of the standard configurations, you shouldn't have trouble modifying it.

The Color Palette

The palette information is stored in the array of 16 triples. It's the only part of the PCX format that's really tricky — being totally board dependent. Perhaps Shannon at ZSoft explains it better, but here's how I understand it:

The IBM EGA adapter has 16 single-

byte color registers. Each register holds a two-bit intensity value (from 0 to 3) for each color gun.

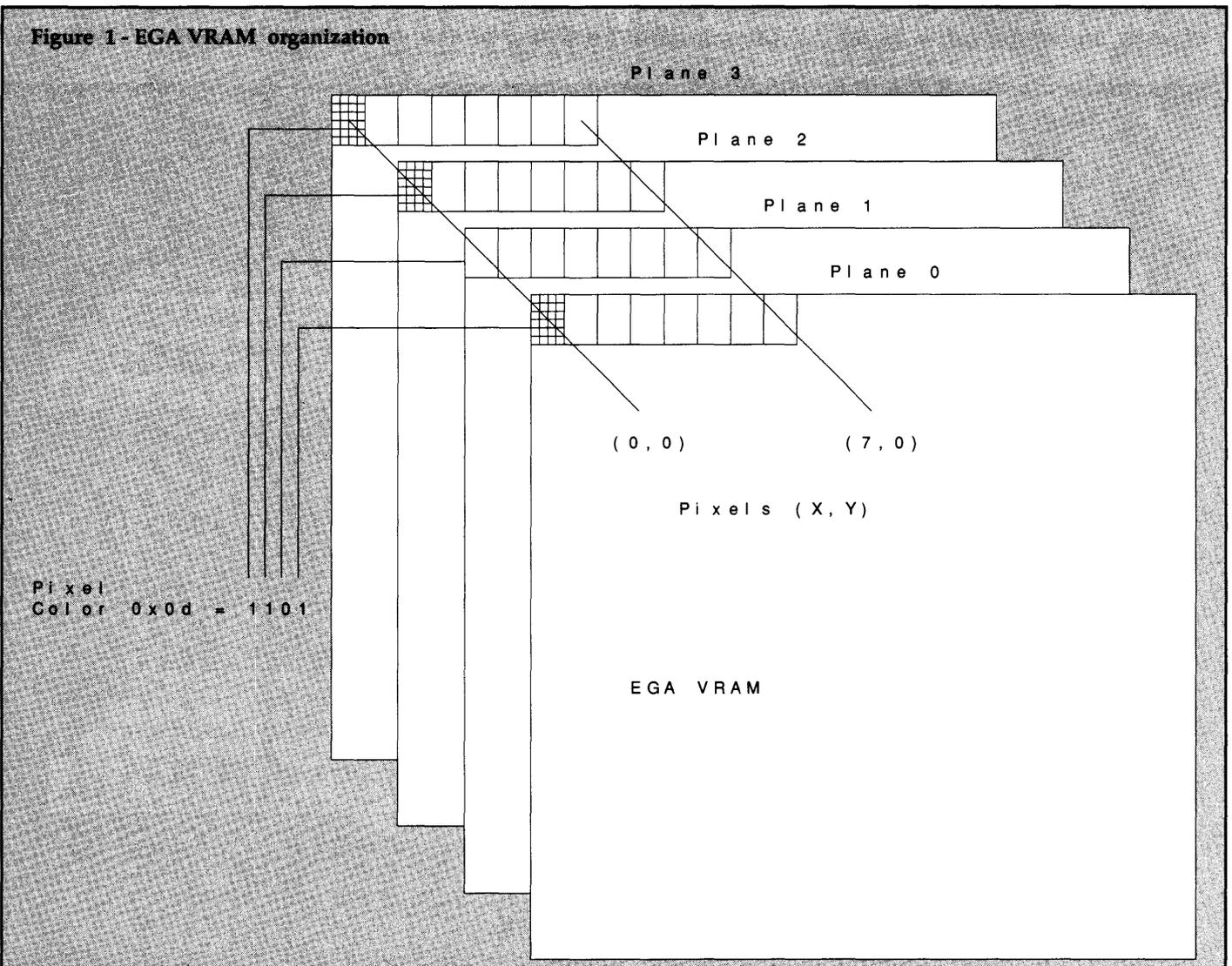
The intensity values are stored like this: "xxRGRgb." The first two bits, 7 & 6, are ignored. The next three bits are the high bits of the intensity values for the Red, Green & Blue guns. The last three bits are the low bits of the intensity values. Thus, each of the 16 registers contains one of 64 possible color combina-

tions.

The PCX header has a 16-element palette table, one element per color register. Each element is a three-byte structure (TRIPLET) which holds an intensity value for each color gun. To set the color for a palette register, the Red, Green & Blue intensity bytes contain the color intensity (after it's been multiplied by 85). (I have no idea why we multiply by 85.) To set the EGA palette register,

we take the intensity value, divide by 85, twiddle the bits, then take the easy route to the palette register by calling the BIOS Set Palette function.

The CGA board has a Color Select Register at I/O location 0x3d9. The bit configuration for this register is "xxCAIRGB." The first two bits are ignored. The C bit selects one of two predefined palettes, called Color Sets. Set 1 is Background, Green, Red & Brown. Set



2 is Background, Cyan, Magenta & White.

The Background color is one of 16 colors formed using the IRGB bits. I is an intensity bit, R, G & B are enables for the Red, Green and Blue color guns, respectively. If the A bit is set, it "Selects (an) Alternate, Intensified set of colors in Graphics Mode" (see ref. 1.) To me, this translates into 4 color sets and one of 16 background colors.

Storage of the CGA palette settings in the PCX header is simple: The first byte of the first element in the palette table contains a background color, multiplied by 16. The first byte of the second element in the palette table apparently contains a three bit value, multiplied by 32. (I haven't been able to figure out what to do with the third bit.) At any rate, there are 8 such settings in the PCX headers palette table, but it seems to me that only one of them can be in effect at a time.

There are probably many more palette formats for other boards, but I don't know of any documentation on them. In the code that accompanies this article, the `pcx_set_palette()` function has a `load_palette` flag which must be set in

order to get the routine to tinker with the palette registers. I didn't bother much with the palette for a number of reasons; the first one being that the palette is automatically set each time a graphics mode is selected; another being that Ventura Publisher ignores it.

Compressing A Plane

When in a file, each plane in the picture is compressed. The compression is very simple; designed simply to take advantage of the fact that a byte value will often repeat.

The read byte algorithm works like this: Read a byte. If the two high order bits are not set ((`byte & 0xc0`) != `0xc0`), then return the byte. If the two high order bits are set ((`byte & 0xc0`) == `0xc0`), then the remaining bits represent a repeat count (`count = byte & 0x3f`) and the next byte is the value that repeats. As such, the repeat count can never be greater than `0x3f`, or 63. The write byte algorithm is obviously the opposite operation.

The Software

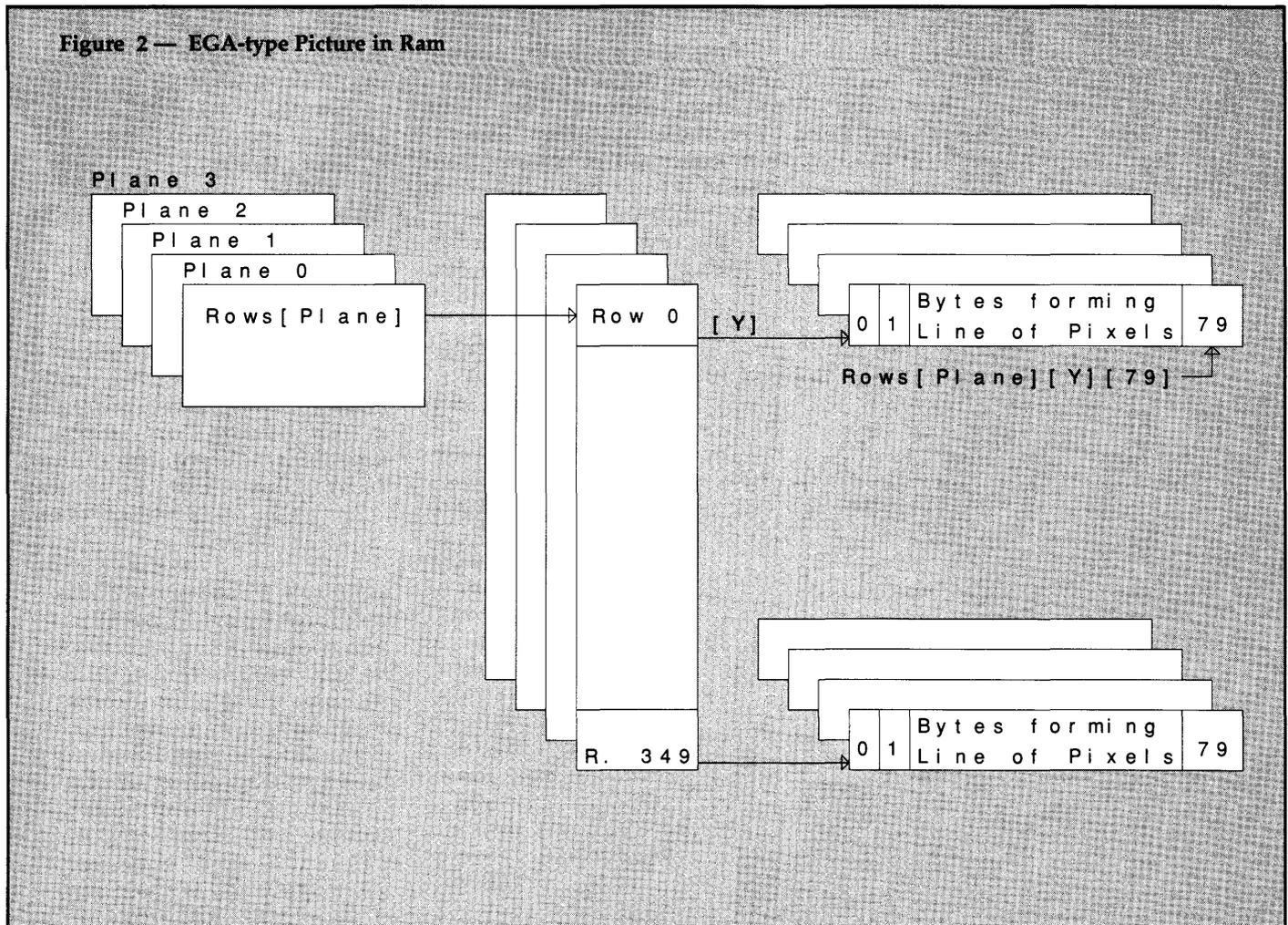
Originally, what I wanted to do was

copy schematics into Ventura. I sat in front of the system for many sleepless nights decoding DXF files, only to find in the end that the GEM files Ventura creates from the DXF files don't properly scale text.

To make a long story short, I wrote most of a DXF to PCX converter, then (balking at the prospect of having to write a font editor, create fonts, and write the code to scale them) scrapped the idea — it was less time consuming to just chop the schematics into small blocks. In the mean time, I'd written a (simple) board independent graphics package, several test routines, and a slide show program. As you've just guessed, I don't support text.

Everything was compiled using Manx Aztec C 86, V. 4.10a. With the exception of the code that handles flicker on the CGA board, it should compile without complaint on most systems. Note, however, that it'll only work in a large data model; not only because of the huge amount of RAM that it uses, but because of the way I accessed the VRAM.

There isn't enough space to list all the code, so I'll document the software here,



New, Lower Prices for CP/M

- VEDIT Version 1.40 \$49 (Single file, no windows)
- VEDIT PLUS Version 2.32 \$79 (Multiple file, no windows)
- VEDIT PLUS Version 2.33 \$95 (Current version with windows)

```

TEXT LINE: 15 COL: 16 FILE: PHOTO 203 INSERT E1
WINDOW 0
/* Main loop - displays the ma
do {
  scrlines = SCRNLINES;
  scrlwidth = SCRWIDTH;
  clrscr(screen(scrlines-20));
  show( main menu );
  ret_val = getrange( rm_pro
process( ret_val, (new ved
) while ( ret_val != EXIT_OK )
)
if ( new vedit && (table in !=
printf( crt_sel );
if (yesno( "" )) setcrl( ar
else outcrlf();
)
WINDOW $

```

DIRECTORY C:\VEDIT\NEW
COMPARE VDM CU203 VDM MAIL VDM MENU VDM PRINT VDM
SORT VDM STRIPV VDM Z80 8086 VDM



#1 PROGRAMMABLE EDITOR

FREE Fully Functional Demo Disk *

Stunning speed. Unmatched performance. Total flexibility. Simple and intuitive operation. The newest VEDIT PLUS defies comparison.

Try A Dazzling Demo Yourself.

The free demo disk is fully functional - you can try all features yourself. Best, the demo includes a dazzling menu-driven tutorial - you experiment in one window while another gives instructions.

The powerful 'macro' programming language helps you eliminate repetitive editing tasks. The impressive demo/tutorial is written entirely as a 'macro' - it shows that no other editor's 'macro' language even comes close.

Go ahead. Call for your free demo today. You'll see why VEDIT PLUS has been the #1 choice of programmers, writers and engineers since 1980.

Available for IBM PC, Tandy 2000, DEC Rainbow, MS-DOS, CP/M-86 and CP/M-80. (Yes! We support windows on most CRT terminals, including CRT's connected to an IBM PC.) Order direct or from your dealer. \$185.

Compare features and speed

	BRIEF	Norton Editor	PMATE	VEDIT PLUS
'Off the cuff' macros	No	No	Yes	Yes
Built-in macros	Yes	No	Yes	Yes
Keystroke macros	Only 1	No	No	100*
Multiple file editing	20+	2	No	20+
Windows	20+	2	No	20+
Macro execution window	No	No	No	Yes
Trace & Breakpoint macros	No	No	Yes	Yes
Execute DOS commands	Yes	Yes	Yes	Yes
Configurable keyboard Layout	Hard	No	Hard	Easy
'Cut and paste' buffers	1	1	1	36
Undo line changes	Yes	No	No	Yes
Paragraph justification	No	No	No	Yes
On-line calculator	No	No	No	Yes
Manual size / index	250/No	42/No	469/Yes	380/Yes

Benchmarks in 120K File:

2000 replacements	1:15 min	34 sec	1:07 min	6 sec
Pattern matching search	20 sec	Cannot	Cannot	2 sec
Pattern matching replace	2:40 min	Cannot	Cannot	11 sec

VEDIT and CompuView are registered trademarks of CompuView Products, Inc. BRIEF is a trademark of UnderWare, Inc. PMATE is a trademark of Phoenix Technologies Ltd. Norton Editor is a trademark of Peter Norton Computing Inc.

* Demo Disk is fully functional, but does not readily write large files.

Reader Service Number 7

Call for 286 / XENIX Version Fully Network Compatible

- Simultaneously edit up to 37 files of unlimited size.
- Split the screen into variable sized windows.
- 'Virtual' disk buffering simplifies editing of large files.
- Memory management supports up to 640K.
- Execute DOS commands or other programs.
- MS-DOS pathname support.
- Horizontal scrolling - edit long lines.
- Flexible 'cut and paste' with 36 'scratch-pad' buffers.
- Customization - determine your own keyboard layout, create your own editing functions, support any screen size.
- Optimized for IBM PC/XT/AT. Color windows. 43 line EGA.

EASY TO USE

- Interactive on-line help is user changeable and expandable.
- On-line integer calculator (also algebraic expressions).
- Single key search and global or selective replace.
- Pop-up menus for easy access to many editing functions.
- Keystroke macros speed editing, 'hot keys' for menu functions.

FOR PROGRAMMERS

- Automatic Indent/Undent for 'C', PL/I, PASCAL, etc.
- Match/check nested parentheses, i.e. '{' and '}' for 'C'.
- Automatic conversion to upper case for assembly language labels, opcodes, operands with comments unchanged.
- Optional 8080 to 8086 source code translator.

FOR WRITERS

- Word Wrap and paragraph formatting at adjustable margins.
- Right margin justification.
- Support foreign, graphic and special characters.
- Convert to/from WordStar and mainframe files.
- Print any portion of file; selectable printer margins.

MACRO PROGRAMMING LANGUAGE

- 'If-then-else', looping, testing, branching, user prompts, keyboard input, 17 bit algebraic expressions, variables.
- Flexible windowing - forms entry, select size, color, etc.
- Simplifies complex text processing, formatting, conversions and translations.
- Complete TECO capability.
- Free macros: • Full screen file compare/merge • Sort mailing lists • Print Formatter • Menu-driven tutorial

CompuView

1955 Pauline Blvd., Ann Arbor, MI 48103 (313) 996-1299, TELEX 701821
 Reader Service Number 7

and you can download the files from one of the following sources: The file "\$\$\$PCX.ARC" will be on the Micro C RBBS long before you see this (and on the Micro C Issue #42 disk); and I'll put it in one of the IBM forums on CompuServe, in thanks to the many people who put together the EGA.ARC file (see ref. 2.) Finally, there'll be a copy in the LISTINGS/IBM.ARC section of BIX.

The Functions

Basically, the functions can be divided into three groups. The lowest level contains the board level driver modules. There's one each for the HERC, EGA, and CGA boards. The PCX module is set up as just another video board. Each module has the same collection of functions, each function name prefixed with the board name.

The next level up contains what I call the VGR modules (Video GRaphics). It's a simple interface between the driver modules and the next higher level.

This level contains a bunch of macros to allow access to the driver level while still being able to switch drivers at runtime. Using these macros, you can write object draw functions which are independent of the low level drivers. So far, I've put in Bresenham's line drawing algorithm and an inefficient but functional irregular polygon fill. A circle drawing algorithm from the May '83 issue of *Doctor Dobbs Journal* is likely to go in next.

The Picture level is the highest. The functions at this level operate on the following picture structure:

```
typedef struct {
    PCXHDR hdr;
    char **rows[4];
} PCXPIC;
```

(I should warn you that the following few paragraphs will be particularly boring. If you're not interested in how I chose to handle the pointer and bit twiddling details, skip to the next heading...)

The PCXHDR is obvious enough, but the rows array deserves some explanation. Figure 2 shows how I visualize it when drawing a 640x350x16 EGA-type picture. A plane is a pointer to an array of rows, each of which is a pointer to an array of characters. That array of characters contains the bits which will become a horizontal line of pixels.

This approach is inefficient because it uses several 32 bit pointer operations, but it's easy to work with. As you can see, I've hardwired a limit of 4 planes, and the arrays are dynamically allocated.

For all but the 320x200x4 CGA modes, we can read the bit at position (x,y) on a plane using the following expression:

```
!!(pcx_cplic->rows[plane][y][x>>3] &
(0x80 >> (x & 7)));
```

The global variable `pcx_cplic` is the pointer to a PCXPIC picture (`cplic == current picture`). It's used to tell the pixel routines what picture to work on.

For the HERC board, or the 640x200x2 CGA mode, you deal with one plane (zero). The 640x350x16 EGA mode forces you to repeat the operation with all four planes. Since `rows[plane]` is a pointer to an array of rows, `rows[plane][y]` is a pointer to the y^{th} row.

All that's needed now is the byte to look at in the row, and the bit in that byte. The low order three bits of x ($x \& 7$) select the bit, and the remaining bits of x ($x \gg 3$) become the byte offset for the row. The above expression would return 1 if the bit at (x,y) is set, or zero if it's not.

The 320x200x4 CGA mode is slightly more complicated since two adjacent bits are used to select a color. The expression to return the color of the pixel at location (x,y) in that mode would be:

```
i = (x & 0x03) << 1;
color = (pcx->rows[0][y][x>>2] &
(0xc0 >> i)) >> (6-i);
```

Since each byte represents 4 pixels, $x \gg 2$ is used as the byte offset for the row, and $x \& 0x03$ is the number of the pixel in the byte. Multiplying that number by 2 gives i , the number of bits from bit 7 that we skip over to get to the first of the two bits that select the color of the pixel. $0xc0 \gg i$ makes a mask for those bits.

The big difference between this expression and the previous one is that we need a two-bit result, so we can't use the `!!` operator to clean up after the bit-wise AND. Shifting to the right $6-i$ bits does the job.

The only difference between the two expressions above and the ones used in the EGA and CGA modules is that there aren't any planes. The EGA board uses a plane register to select each plane and all four planes use the same memory locations. The HERC board has two pages, which can be thought of as unrelated planes. I visualized each horizontal row of pixels as being an array of columns, so the arrays are called columns instead of rows. The array of pointers to each row is dynamically allocated. The pointers are initialized with sequential row addresses

for the EGA board, and scattered addresses for the HERC and CGA boards.

Although the pointer operations and bit twiddling never get more complicated than that, some of the expressions aren't much fun to read. Fortunately, all the low level operations are ready to go. You can just throw them into your library and use them. With that said, let's get down to function names and parameter lists...

Using The Functions

Note that the declarations here don't match the ones in the modules. Many functions that don't return anything are declared as returning `int`. Most parameters to most of the functions are not range-checked.

```
void allocf(char *p);
void map_not(int *map,int len);
```

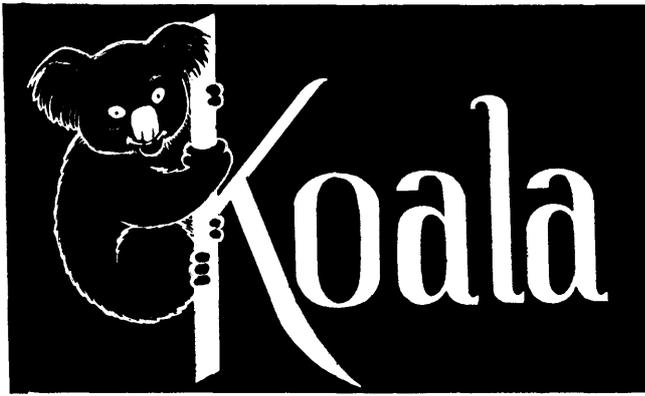
These two come out of my library. `allocf()` checks the pointer p to see if it's NULL. If so, it doesn't do anything. If not, it calls `free()`. If `free()` returns an error code, `allocf()` generates an error message and the program halts. `map_not()` inverts len ints in the bit-map pointed at by map .

```
void cga_movmem(char far *s,char far
                *d,intn);
void cga_peekb(char far *p);
void cga_pokeb(char far *p,char b);
```

These functions are the same as `movmem()`, `peekb()` and `pokeb()`; except that they wait for one of the CGA's horizontal retraces before access, to avoid flicker. The `cga_movmem()` function is a straight block move, so you'll have to check for overlapping blocks if you plan to move data from one part of CGA VRAM to another.

```
void ega_select_plane(int plane);
int pcx_select_plane(int plane);
```

The EGA VRAM holds 4 planes, each mapped into the same memory space. `ega_select_plane()` is used to select one or more. If the function is passed a negative number, it uses the absolute value of the number as an enable mask (ie: if $plane == -0x0f$, all 4 planes are selected simultaneously). This is handy when clearing the screen since a single call to `setmem()` can then clear all 4 planes! If $plane$ is not negative, then it must be from 0 to 3, and only that single plane is enabled. `pcx_select_plane()` selects the plane which will be used by the `VGR_ROW()` macro (while drawing on a PCX picture).



KOALA COMPUTER, INC.
The Friendly Computer

!!LOOK!! Greenbar 11½x14⅞ paper.

Only 2000 cases left. This is a one time deal and we bought it all. Case approx. 35 lbs. Two holes punched at top for binders. Your choice 2 part or 4 part. 2 part \$11. 4 part \$10. When this is gone — no more!!

XT Style Keyboard manufactured by Cherry.

LEDs in NUM and CAP LOCK keys. Complete with coil cord and plug but without case. Our Service Department sells these as an alternative to repair the old style PC keyboards. Plus you end up with a keyboard that has a much nicer feel than the regular clone keyboard. If you are a touch typist this is a nice upgrade and KOALA Beary Beary inexpensive. \$25. 2 lb.

Monochrome Display Adapter.

(MDA) with parallel printer port. These are manufactured in Japan and we think they are better quality than the usual components from the Pacific Rim. \$35. 2 lb.

Not a mistake and no Bull — but quantities are limited. Expand your 256K system to 640K. **384K expansion card** parallel port, serial port, game port, clock calendar, and 384K of memory installed and tested. \$160. 3 lb.

Large diskette boxes with metal keylock.

This is the size (6¾x14 inches) that are called 100 diskette capacity. We store less in them ourselves but to each his own. \$12. 3 lb.

10 DSDD 5¼ Diskettes with hub rings, labels, Tyvek sleeves and write protects in a Dial-N-File box. Container cover flips to form an easel. Turn the lever on the front and the diskettes are separated for easy selection and extraction. Hear lots of nice comments about these! \$12.50 a box. 10 boxes \$110. 1 lb.

HELP!

The command DOS* forgot! **DOS HELP by Flambeaux.** Easier and faster than the DOS manual. Usage examples are very good too! List \$35. Our price \$20. 1 lb.

Nice Print.

Take you all evening to sample all the Fonts in this one. Roman, Script, Olde English, etc. List \$95. Our price \$70. 2 lb.

Six-foot parallel printer cable.

Not the cheap ones, but the DB25 with the knurled fastener (Centronics parallel on the other end of course). \$10. 1 lb.

Six outlet power strip.

With 3 prong plug, switch, circuit breaker, and MOV surge suppressors. Why pay up to \$75 for this protection when you can buy for the Beary Friendly price of \$15. 3 lb.

Dbase Application in Business.

ISBN 0-8359-1241-8. A 335 page 8½x11 book published by Reston Publishing Company, Inc., A Prentice-Hall Company, \$4. 2 lb.

TERMS — Cashier's check — immediate shipment; Personal checks, Money Orders, and others must clear. No C.O.D. No Collect Calls. All prices FOB Torrance, CA. All items shipped UPS — Contact your local UPS office for freight charges. Approximate shipping weights in product description.

213-316-5866, (9 to 5 PST)

4306 Torrance Boulevard, Torrance, California 90503 (offices only)
Quantities limited to stock on hand on most items. Add \$2 handling to each order.
California residents add 6½% sales tax. See note above regarding freight charges.

```
void ega_set_palette(char reg, char
red, char green, char blue);
```

Sets one of the 16 EGA palette registers. reg can have a value of from 0 to 15. red, green and blue set the intensities for each color gun. Values range from 0 to 3.

```
int herc_set_page(int page);
```

The HERC driver will default to page 0, but it will let you select either of the pages.

```
int *pcx_init(void);
cga_init(void);
ega_init(void);
herc_init(void);
```

```
int VGR_HRES,
VGR_VRES,
VGR_NCOLORS,
VGR_NBPL; /* #bytes per line */
```

The `_init()` routines initialize all static local variables, allocate space (if it hasn't been allocated by a previous `_init()` call) and tell the VGR module to use the module that was INITed. The VGR module, among other things, holds the four ints shown above. The HERC and EGA modules always set these values since they're constants. The PCX module will set them only if `pcx_cpvc` is not null, therefore pointing to a valid PCXPIC picture. The CGA module sets them after a `VGR_MODE()` call. The `_init()` functions each return OK or ERROR, with ERROR usually resulting from an out of memory condition.

```
int VGR_MODE(int m);
```

Once the VGR module has been set up, using the appropriate `*_init()`, you must set the mode. Since the `_init()` call set up the array of pointers to functions in the VGR module, you can use the #defined macros in `vgr.h`. `VGR_MODE()` is one of those macros. `m` must be one of the following:

```
MODE_TEXT0 - text, 80x25
MODE_APA0 - APA, 640x350x16
MODE_APA1 - APA, 720x348x2
MODE_APA2 - APA, 640x200x2
MODE_APA3 - APA, 320x200x4
```

When drawing on a PCX picture, the mode is used to tell the PCX module whether it's producing a one or two-bit per pixel picture; so available PCX modes are effectively `MODE_APA0` or

`MODE_APA3`. Available EGA modes are `MODE_TEXT0` or `MODE_APA0`; for the CGA it's `MODE_TEXT0`, `MODE_APA2` or `MODE_APA3`; and for HERC it's `MODE_TEXT0` or `MODE_APA1`. ERROR or OK is returned with ERROR indicating an invalid mode number.

```
void VGR_CLEAR()
void VGR_SET(int x,int y,int c)
void VGR_CLR(int x,int y)
void VGR_XOR(int x,int y,int c)
int VGR_GET(int x,int y)
void VGR_ROW(int r,char far *p,int n)
void VGR_MOVE(char far *s,char far
*d,int n)
void VGR_PEEKB(char far *p)
void VGR_POKEB(char far *p,int b)
```

These VGR_ macros allow access to the board independent functions. `_CLEAR` clears the screen. `_SET` sets the color of the pixel at (x,y) to c. If c is zero, it clears the pixel.

It's a good idea to not confuse the two operations since the HERC module ignores the value of c. `_CLR` sets the color of the pixel to zero. `_XOR` does a bit-wise exclusive or of the color of the pixel with c. Again, the HERC module ignores c. `_GET` returns the color of the pixel. `_ROW` moves n bytes from location p to the VRAM row r. `_MOVE`, `_PEEKB` & `_POKEB` either call `movmem()`, `peekb()` & `pokeb()`, or `cga_movmem()`, `cga_peekb()` & `cga_pokeb()`; depending on the driver module you're using.

```
void vgr_fill(int x,int y,int color);
void vgr_line(int x1,int y1,int
x2,int y2,int color);
void vgr_rectangle(int x1,int y1,int
x2,int y2,int color);
void vgr_point(int x2,int y2,int
color);
```

`vgr_line()`, `_rectangle()` and `_fill()` do what they sound like. `vgr_point()` does one of two things. If color is -1, the values (x2, y2) are copied to the local static values (x1,y1). If color is not -1, `vgr_point()` calls `vgr_line()`, then copies (x2,y2) to (x1,y1). This makes it possible to draw lines between a series of points.

```
int vgr_get_board(void);
int vgr_mode(char mode);
```

`vgr_get_board()` tries to figure out what kind of video graphics board is in use. The return value is one of `TYPE_UNKNOWN`, `TYPE_EGA`, `TYPE_CGA`, `TYPE_MDA` or `TYPE_HERC`. These values are defined in

`vgr.h`. `vgr_mode()` calls the BIOS set graphic mode function; INT 0x10, AH==0. See ref. 2, page A-46 for more info on that BIOS call.

```
PCXPIC *pcx_init_pic(int hres,int
vres,int nplanes);
void pcx_free_pic(PCXPIC *pic);
void pcx_invert_pic(PCXPIC *pic);
```

`pcx_init_pic()` dynamically allocates space for a PCXPIC structure, and returns a pointer to it. `hres` & `vres` are the number of dots horizontally and vertically; `nplanes` is the number of planes in the picture. It dynamically allocates arrays, returning pointers to them. If it runs out of memory, it de-allocates all arrays and returns NULL.

`pcx_free_pic()` frees up all the space allocated for a picture.

`pcx_invert_pic()` inverts all the bits in each row of the picture.

```
int pcx_read_pic(PCXPIC *pic,FILE
*fp);
int pcx_getc(int *c,int *n,FILE
*fp,int maxn);
int pcx_write_pic(PCXPIC *pic,FILE
*fp);
int pcx_xputc(int c,FILE *fp);
int pcx_putc(int c,int n,FILE *fp);
```

These functions operate on a file which has been opened by one of the STDIO buffered file open functions. `fp` is the file pointer returned by the open call. `pic` is a pointer to a PCXPIC picture structure. These functions return OK or ERROR.

`pcx_read_pic()` loads a picture from the file. The `pic` structure can either be statically allocated, or dynamically allocated (using `pcx_init_pic(0,0,0)`). `pcx_read_pic()` automatically allocates picture arrays. `pcx_write_pic()` writes the picture to the file.

`pcx_getc()` reads a character or pair of characters from the file. The character is placed in the integer pointed at by `c`, and a repeat count of one or more is placed in the integer pointed at by `n`. `maxn` is the maximum repeat count that the calling function wants to receive.

`pcx_write_pic()` calls `pcx_xputc()`, which counts the number of times it receives a given byte value, then calls `pcx_putc()` with that number. If `pcx_xputc()` is called with `c == -1`, the buffered byte value and count are passed immediately (flushed) to `pcx_putc()`. `pcx_putc()` writes the byte `c` to the file, using a repeat count of `n`. `n` may have a value of up to 32767; `pcx_putc()` will

recursively call itself with $n \leq 63$ until the correct total has been written to the file.

```
void pcx_showpic(PCXPIC *pic,int
hoffs,int voffs,int load_palette_flg);
```

This function uses VGR_ROW to copy rows from the pic picture to the video board driver module in use. hoffs is an offset from the start of each row in pic. voffs is an offset which is applied to the row number. These two offsets allow the screen to be panned around a large picture. The load_palette_flg, when set, allows pcx_showpic() to set the hardware palette registers from the PCXPIC palette table. Consider the palette functions to be unreliable.

The Test Routine

As I write this, PCX.EXE is running and bug free. It's a test routine which exercises the modules. If executed without command line parameters, it tries to figure out which video board is available and if it sees a CGA, HERC, or EGA board, it creates and displays a simple test picture (see Figure 3).

If it's invoked with a board name, CGA (640x200x2), EGA, HERC or CGA2 (320x200x4), it creates a picture for that

board, and tries to display it on that adapter.

If it's invoked with a board name followed by a file name, the picture is saved, using the specified file name, after it's been displayed.

PCXSHOW.EXE, the slide show program, is in need of some upgrading — it was written before the latest improvements to the PCX/VGR modules. It'll be ready before you see this, and will be included with the other files.



Figure 3 - Picture Generated by the PCX.EXE test routine.

Bibliography

- 1. The IBM Personal Computer XT Technical Reference Manual, April 1983 revision. Part Number 1502237.
- 2. The file EGA.ARC, which was found somewhere on CompuServe, probably in one of the IBM forums.
- 3. ZSoft Technical Reference Manual. ZSoft Corporation, 1950Spectrum Circle, Suite A-495, Marietta, GA 30067. Tel. (404) 428-0008. Many thanks to Shannon.
- 4. A Hercules Primer, by Larry Fogg, *Micro Cornucopia*, Issue #39, Jan-Feb '88, page 26.
- 5. Tidbits, by Gary Entsminger, *Micro Cornucopia*, Issue #39, Jan-Feb '88, page 84.
- 6. Language Connections, by Gary Entsminger, *Turbo Technix*, Jan-Feb '88, page 136.
- 7. MicroEMACS by Dan Laurence & others. Public domain text editor, with C sources. Available on BIX in LISTINGS/IBM.ARC.



ALL SALES SUBJECT TO THE TERMS OF OUR 90 DAY LIMITED WARRANTY. FREE COPY UPON REQUEST.

CANON 80 COLUMN PRINTER - \$29.95
 ORIGINALLY MANUFACTURED FOR THE PC JR, BUT WITH OPTIONAL CONNECTOR WILL WORK WITH PC, XT, OR AT. REQUIRES SERIAL I/O. THIS THERMAL PRINTER IS QUIET AND USES EASY TO GET 8 1/4 IN. ROLLS OF PAPER, 50 C.P.S., UPPER AND LOWER CASE, PLUS GRAPHICS. ORIGINAL LIST PRICE \$199.00. ADD \$3.00 FOR PC/XT CONNECTOR. ADD \$5.00 UPS.

THE NEW ZRT-80 CRT TERMINAL BOARD!
 A LOW COST Z-80 BASED SINGLE BOARD THAT ONLY NEEDS AN ASCII KEYBOARD, POWER SUPPLY, AND VIDEO MONITOR TO MAKE A COMPLETE CRT TERMINAL. USE AS A COMPUTER CONSOLE, OR WITH A MODEM FOR USE WITH ANY OF THE PHONE-LINE COMPUTER SERVICES.

FEATURES:

- * Uses a Z80A and 6845 CRT Controller for powerful video capabilities.
- * RS232 at 16 BAUD Rates from 75 to 19,200.
- * 24 x 80 standard format (60 Hz).
- * Optional formats from 24 x 80 (50 Hz) to 64 lines x 96 characters (60 Hz).
- * Higher density formats require up to 3 additional 2K x 8 6116 RAMS.
- * Uses N.S. INS 8250 BAUD Rate Gen. and USART combo IC.
- * 3 Terminal Emulation Modes which are Dip Switch selectable. These include the LSI-ADM3A, the Heath H-19, and the Beehive.
- * Composite or Split Video.
- * Any polarity of video or sync.
- * Inverse Video Capability.
- * Small Size: 6.5 x 9 inches.
- * Upper & lower case with descenders.
- * 7 x 9 Character Matrix.
- * Requires Par. ASCII keyboard.

\$89.95 A&T ADD #ZRT-80 \$50
 (COMPLETE KIT, 2K VIDEO RAM)

OUR BEST SELLER!

FOR 8 IN. SOURCE DISK OR PC-XT FORMAT 5 1/4 IN. ADD \$10

Digital Research Computers
 P.O. BOX 381450 • DUNCANVILLE, TX 75138 • (214) 225-2309

Call or write for a free catalog on Z-80 or 6809 Single Board Computers, SS-50 Boards, and other S-100 products.

TERMS: Add \$3.00 postage. We pay balance. Orders under \$15 add 75¢ handling. No C.O.D. We accept Visa and MasterCard. Texas Res. add 8-1/4% Tax. Foreign orders (except Canada) add 20% P & H. Orders over \$50 add 85¢ for insurance.

ShareWare™

BROWN BAG Software™

HARD DISK MADE EASY

PowerMenu™

- Pop-up User Menus
- File Security
- Colorful and "Goof Proof"
- Runs Your Program with One Keystroke
- Great for Networks
- All DOS Functions without "Computerese"
- DOS Window

PowerMenu™ is Brown Bag Software's DOS operating environment. It creates colorful pop-up stacking menus that run your programs at the press of a single key. Programs can be password protected if you wish and access to "Raw DOS" can be restricted. **PowerMenu™** also features a powerful file manager that displays files and directories. Multiple files can be "Flagged" for "One Pass" copy erase or more. Both novices and Power Users will appreciate **Power Menu's™** speed and ease of use. Takes just 2.5K of RAM! A **ShareWare™** version is available on CompuServe™, Genie™, Delphi™, Bix™, and The Source™, or directly from the Brown Bag BBS (408) 371-7654.

\$89.95 Plus \$10 Shipping and Handling.

1-800-523-0764
 IN CALIFORNIA 1-800-323-5335 or (408) 559-4545

VISA, MasterCard accepted or mail your check to:
 File 41719, Box 60000, San Francisco, CA 94160-1719

SHAREWARE DISK AVAILABLE FOR \$10.00

© Copyright 1988, Telemarketing Resources, Inc.



Zortech's C & WATCOM's C

And Manipulating MS-DOS

By Scott Robert Ladd

P.O. Box 61425
Denver, CO 80206
(303) 322-7294

Scott remains buried under the droppings of his UPS driver. That means, of course, there are more interesting updates to the benchmark tables. Also, he manipulates global MS-DOS variables as a way of passing information from one program to the next.

New things continue to happen in the world of C. There's a new compiler from Canada, a few bug fixes, and some upcoming products of interest. As usual, a newly updated version of the compiler benchmark table is available on the Micro Cornucopia Bulletin Board at (503) 382-7643 or on Micro C's Issue #42 disk for \$6.

Microsoft

In February, Microsoft announced new versions of *all* of their programming languages, including C. Microsoft Optimizing C 5.1 not only fixes the bugs from version 5.0, but adds OS/2 compatibility and a faster library. By the time you read this, it should be shipping.

The powerful Microsoft Editor is included. Extensions can be added to the editor by writing new functions in C. It can handle multiple files and allows the user to run programs (including compilers) from within it. The editor, like most of the other utilities, has a version which runs under OS/2 and a version which runs under MS-DOS.

Version 5.1 produces faster programs than before. Display output has really been improved; the Display benchmark compiled with 5.1 is 30% faster than it was when compiled with 5.0. All of the proposed ANSI-standard pre-defined macros (such as `__TIME__` and `__DATE__`) are available. Several pragmas (compiler directives) are new with 5.1. Most of these control the output of program listings.

QuickC has undergone a minor revision (to version 1.01), which includes the faster 5.1 libraries. Unfortunately, most of the other 5.1 enhancements (the new pragmas and pre-defined macros) are not available in QuickC. The problem I mentioned last issue with a specific Western Digital disk controller has

been fixed. The new QuickC is *not* compatible with OS/2.

Zortech C and C++

As is usual in this business, by the time the May/June column saw print, it was out-of-date. Where's a cheap time machine when you need one?

Walter Bright no longer markets his excellent Optimizing C compiler through Datalight. Instead, a company called Zortech will be marketing the compiler in the United States. Zortech and Walter have a longstanding relationship, since Zortech (under its old name Zorland) has been selling the compiler in Europe for several years.

In mid-May, Zortech will introduce his new C++ compiler for \$99, followed by the release of an upgraded version of the regular C for \$49.

Yes, I did say C++ — this will be the first compiler for C++ under MS-DOS. The final product should be excellent, judging from my beta-copy. For those of you in the dark, C++ is an object-oriented extension of K&R C. To be honest, I haven't worked much with C++, but everything I hear makes me want to learn more.

The Optimum C compiler I have (also a beta) produces code even faster than before. I find it amazing that Walter continues to find ways to improve his already excellent product.

Both compilers support the segment ordering and naming conventions used by Microsoft's products, so you can use Microsoft CodeView for debugging Zortech C++ and C programs. Two other improvements will be better manuals (produced by Zortech) and an extended library. The head of the American division of Zortech told me that former Datalight customers will be able to get support from Zortech.

Zortech Inc.
361 Massachusetts Ave.
Arlington, MA 02174
(800) 848-8408

WATCOM

Just when it would seem to be futile for another C compiler to enter the market, an old hand at mainframe software jumps right in. WATCOM is based in Waterloo, Ontario, and is well-known for its IBM-mainframe compilers (especially WATFOR).

Editor's note: Rumor has it they'll be coming out shortly with a flexible new language called WATEVER (WATELSE would they call it?) and I understand they have an even more dramatic product on tap. It's a non-fattening liquid refreshment for programmers called WATER.

Now, they've introduced a complete C development system for IBM compatibles.

Actually, there are two C packages from WATCOM. The first is Express C, an inexpensive (\$75) prototyping compiler with an integrated environment. The other is a full-blown development system containing the optimizing C 6.0 compiler and tools for \$295. Express C is included in the more comprehensive package, much like the way Microsoft includes QuickC with their optimizing compiler.

Express C, unlike QuickC, cannot be used by itself to generate professional-quality programs. An empty program generates an 83K executable, for example. Express C has built-in debugging functions, and is targeted primarily at the educational market.

The main compiler is another story. Here we have a high-powered, optimizing compiler. They throw in a complete set of tools, including a linker, librarian, disassembler, and source-level debugger. Both the main compiler and Express C adhere to the December 1987 ANSI standard. While the library is a bit small, it is very complete and compatible with Microsoft C.

WATCOM C rearranged the pecking order in my benchmarks. It finished first (speed-wise) in the Dhrystone, AllocMem, and Sort tests, and usually finished in the top five on the other benchmarks.

Compile speeds are too slow for my taste, but it produces small executables. The editor is miserable (in my lowly opinion), but that's about my only complaint. WATCOM's editor reminds me of the first full-screen editors on mainframes, where (for example) you use a function key to insert a new line.

It takes guts to enter the compiler wars. WATCOM C has jumped in with a very good product.

WATCOM PRODUCTS, INC.

415 Phillip St.

Waterloo, Ontario CD N2L 3X2

(519) 884-3141; TLX: 06-957520

MIX Power C

Last issue, I mentioned problems when compiling my benchmark suite with Power C; these have all been fixed in version 1.1.0. The latest results are in the tables on the Micro C RBBS. While it isn't as good as its ads claim (is anything ever that good?), you get a good, ANSI-standard C compiler for less (\$20) than the price of most programming books. If you just want to C what this is all about, you might look into Power C.

The Death Of C?

You may have seen the ads for Stony Brook Modula-2: "I come not to praise C, but to bury it!" Yes, Dave (our enthusiastic editor), I know this is a C programming column. However, I'm an on-again, off-again fan of the Modula-2 language, and I just *had* to check out the benchmark claims in the ad.

For the uninitiated, Modula-2 is an outgrowth of Pascal. Both were designed by Niklaus Wirth. Modula-2 is a strongly-typed language (just like Pascal), and has support for multitasking. While I hate the way it does I/O (a different statement for every data type? Give us a break Mr. Wirth!), Modula-2 has some interesting features.

Most Modula-2 compilers generate mediocre code. The Stony Brook compiler is an exception. Not only will the compiler and libraries fit on just one 360K disk, but it generates small, fast programs. This compiler even produces programs which work with Microsoft Windows and OS/2. While I don't expect it to "bury" C, it certainly is a contender.

Stony Brook Software

Forest Rd.

Wilton, NH 03086

(603) 654-2525

A Speedy Tip

Many C compilers (including Turbo C and Microsoft C) have options for generating code for more advanced microprocessors than the 8088 or 8086 found in most PC compatibles.

If you have a NEC V-20 in your machine (you do if you use a recent Kaypro PC), you can set the compiler switches to generate instructions for the

80186 microprocessor. This will make your programs slightly quicker and smaller. Remember, though, that if you do this, the program generated will not run on the old 8088/86 machines! But for your own internal programs, it's kind of nice...

BBS Of The Issue

For those of you interested in using your modems to contact the world of programming, you might want to check out the C BBS, run by Barry Lynch at (703) 998-8377 (FidoNET node 1:109/713).

Barry's board is dedicated to C programming. He has areas for Microsoft and Borland addicts, and carries an excellent selection of public domain and shareware source libraries and utilities. While Barry asks for a \$10 per year registration fee, his board is worth the money.

Barry is also the producer and editor of a freely distributed electronic newsletter called *C News*. Several boards participating in FidoNET carry this newsletter for downloading. It includes news, reviews, code examples, and some pretty good articles. Definitely recommended!

Book Of The Issue

If you want to do work with serial communications, check out a book called *C Programmer's Guide to Serial Communications*, by Joe Campbell, published by Sams. This is an exceptional book filled with theory, practice, and code examples. It covers not only how your serial port works, but *why* it works.

Editor's second note: Now if someone would tell us why it doesn't work...

Laptop C

Recently, through a quirk of fate, I ended up with a Zenith Z-181 laptop computer (named "Merlin" by my wife). It's a long story, but I've found that I actually like the little machine. It certainly comes in handy when I'm traveling or while I'm at a meeting.

The thought occurred to me that it would be nice to be able to do C programming with the Zenith. My primary criteria was that the compiler be usable from a pair of 720K drives without my needing to swap disks. It can be done, but only with a little trepidation and work.

Of the compilers I have for review, only four would fit with enough space left for source code. These are: Borland

Turbo C, Mark Williams Let's C, Microsoft Quick C, and MIX Power C.

Having been spoiled by a fast hard disk in my main machine, the compile times seemed awfully slow. Those little 3 1/2 inch drives are slower than regular 5 1/4 inch drives. Other than that, I was able to do my compiling without any problems.

Editor's third note: I spent a whole term running Aztec C from the single 720K drive (and 300K RAM disk) on my Kaypro 2000. It was great. I had the class assignment written and running before the period ended. (I was writing, compiling, and, especially, debugging C routines while the instructor was finishing his discussion of the problem. When he asked for questions, I was ready for him.)

Environmental Control

Let's get down to coding!

Last issue (#41) I talked about how MS-DOS and the PC manage memory. As promised, I'm going to build on that information, showing you how to access and manipulate the MS-DOS environment.

The environment is a list of null-ter-

minated text variables maintained by MS-DOS. Each environment variable has a name, customarily in uppercase, an equal (=) sign, and the associated string value.

An example of an environment variable is PATH, which has a list of directories as its value (e.g., PATH=C:\;C:\BIN), and is used by MS-DOS to search for executable programs not found on the default drive/directory. Other common environment variables are COMSPEC (the path and name of the command processor, usually COMMAND.COM) and PROMPT (which contains the definition used in the last PROMPT command).

A number of applications programs use environment variables to hold information on configurations and the locations of components. Most compilers require several variables to be defined in the environment, to give search paths for libraries, include files, and compiler programs. The more software products you have installed on your PC, the more items will need to be stored in the environment.

COMMAND.COM creates the primary, or global, environment when a PC boots up. Some items are stored in the environment through special internal MS-DOS commands; PATH and PROMPT are two such examples. The other way to store environment variables is through the SET command. The command SET TEMP=C:\TEMPDATA would add a new variable to the environment named TEMP, with a value of C:\TEMPDATA.

The first problem to crop up when using the environment is a lack of space. The default environment size is from 128 to 160 bytes, depending on your version of MS-DOS. Although the MS-DOS manuals claim that the environment will automatically expand if it needs more room, it also points out that this expansion can't happen if any memory-resident (TSR) applications have been loaded.

From my experience, the size of the environment when it is created is the size you get to work with thereafter. If we want to have a bigger environment, our choices depend on our version of MS-DOS.

If you have a version prior to 3.10, you're out of luck unless you own a copy of a Microsoft compiler. They provide a special program called SETENV, which patches COMMAND.COM so that it installs a larger environment.

However, if you have version 3.10 or greater, add the following line to your CONFIG.SYS file:

```
SHELL = d:\path\COMMAND.COM /P /E:nnn
```

where d:\path is the drive and directory where you keep COMMAND.COM, and nnn is the size of the environment you want.

Under DOS 3.1, nnn is the number of paragraphs of environment space you want, a paragraph being 16 bytes. When using DOS 3.2 or later, nnn is the actual size of the environment in bytes, and will be rounded up by the system to the nearest 16 bytes.

The maximum size of the environment is 32K bytes, which is far more than you'll ever need. I set mine at 512 bytes (32 paragraphs), and use a number of environment-hungry programs without running out of room.

I gave the format for the Program Segment Prefix (PSP) last issue (#41, May/June 1988). This is the 256-byte area DOS builds in memory immediately before it loads your program (referred to hereafter as the application).

When the application is executed, a copy of the environment is made, and a pointer to this copy is stored in the program's PSP at offset 0x2C. We'll call this the "environment pointer." The copy is made from the environment of the parent program, which is normally the COMMAND.COM loaded at boot time.

Most MS-DOS C compilers have a library function called getenv() which can be used to retrieve the value of a variable from the local copy of the environment. Some libraries even provide a function (usually called putenv()) to store new variables in this area.

You can use the pointer in the PSP if you want to write your own functions to access the application's local copy of the environment. Unfortunately, the local environment is temporary; it goes away when the program finishes. And, modifications to the global environment are not reflected in the local copy.

Let's say you have a program which wants to leave a message behind for other programs. For instance, a configuration program might load setup info into the global environment. Perhaps you want a program to remember where it was when you restart it. Since the local environment is temporary, how can we put a variable into the global one?

68000

and 6800/2/8/9

SOFTWARE

SK*DOS - a powerful DOS for the 6809 (\$75) or the 68000 (\$140, incl. an editor, assembler, Basic, utilities, code for a boot ROM, etc.)

HUMBUG - a monitor/boot ROM, \$50 - \$75.

OTHER SYSTEM SOFTWARE including assemblers, text formatters, editors, spell checkers, languages, etc., all very reasonable.

HARDWARE

A wide selection of single-board computers and systems, from \$275.

COMBINATIONS

Package deals of fast and powerful computer plus DOS and more, from \$350.



SOFTWARE SYSTEMS CORP.

BOX 209 • MT. KISCO, NY 10549

914/241-0287

Reader Service Number 40

There are no built-in MS-DOS functions for accessing the global environment from a user-written application; in fact, MS-DOS doesn't even provide a documented way to *find* it! So, we're back to sneaking through the unknown MS-DOS features.

Looking at offset 0x16 in the application's PSP, we find a pointer to the PSP of the application's parent. If the application was executed from the MS-DOS prompt, this will point to the PSP of COMMAND.COM, which (gosh!) is the owner of the global environment.

So, we should be able to look in the application's PSP, find the PSP location of its owner (COMMAND.COM), and then use the environment pointer in COMMAND.COM's PSP to find the global environment. While this method works in some versions of MS-DOS, it doesn't work in all of them. Some versions store zero in the environment pointer, effectively hiding where COMMAND.COM keeps the global environment. Now what?

At this point, we're down to brute

force. By carefully examining the MS-DOS memory map, I've found that versions of MS-DOS which "zero" the environment pointer keep the environment in the memory block immediately following COMMAND.COM. Since we know the format of MS-DOS memory control blocks (see issue #41), and can "leapfrog" through the chain of them, we can find the global environment.

GLOBENV.C

This issue's module works with MS-DOS versions 2.1 through 3.30, inclusive. Named GLOBENV.C, it's a stand-alone unit which you can link with programs so that they can manipulate the global environment. I have successfully compiled (and used) it with Microsoft, Borland, and Zortech (Datalight) compilers.

(Scott's code is available on the Micro C RBBS or on the Issue #42 disk.)

GLOBENV defines three public functions — `getgenv()`, `putgenv()`, and `delgenv()`. The first three retrieve, store, and delete variable definitions from the global environment. `Findgenv()` is an in-

ternal function which is called by the first invocation of any of the public functions. It locates the global environment using the methods listed above. You should be able to understand how the module works by following the comments.

Wrap Up

Well, that wraps it up for this issue. In the future, I'll be covering some aspects of C which are not MS-DOS related, so as to not leave out those of you with CP/M or Macintosh PCs.

If you have any questions about this column (or C programming in general), please feel free to leave a message on the Micro C RBBS or to drop me a letter.

Next issue I'll be writing about C database libraries, along with my usual update on the who, what, where, and why of C programming. See ya then!

◆ ◆ ◆

ADD TO THE POWER OF YOUR PROGRAMS WHILE YOU SAVE TIME AND MONEY!

CBTREE does it all! Your best value in a B+tree source!

Save programming time and effort.

You can develop exciting file access programs quickly and easily because CBTREE provides a simple but powerful program interface to all B+tree operations. Every aspect of CBTREE is covered thoroughly in the 70 page Users Manual with complete examples. Sample programs are provided on disk.

Gain flexibility in designing your applications.

CBTREE lets you use multiple keys, variable key lengths, concatenated keys, and any data record size and record length. You can customize the B+tree parameters using utilities provided.

Your programs will be using the most efficient searching techniques.

CBTREE provides the fastest keyed file access performance, with multiple indexes in a single file and crash recovery utilities. CBTREE is a full function implementation of the industry standard B+tree access method and is proven in applications since 1984.

Access any record or group of records by:

- Get first
- Get previous
- Get less than
- Get greater than
- Get sequential block
- Get all partial matches
- Insert key and record
- Delete key and record
- Change record location
- Get last
- Get next
- Get less than or equal
- Get greater than or equal
- Get partial key match
- Get all keys and locations
- Insert key
- Delete key

Increase your implementation productivity.

CBTREE is over 6,000 lines of tightly written, commented C source code. The driver module is only 20K and links into your programs.

Port your applications to other machine environments.

The C source code that you receive can be compiled on all popular C compilers for the IBM PC and also under Unix, Xenix, and AmigaDos! No royalties on your applications that use CBTREE. CBTREE supports multi-user and network applications.

CBTREE IS TROUBLE-FREE, BUT IF YOU NEED HELP WE PROVIDE FREE PHONE SUPPORT.

ONE CALL GETS YOU THE ANSWER TO ANY QUESTION!

CBTREE compares favorably with other software selling at 2,3 and 4 times our price.

Sold on unconditional money-back guarantee.

YOU PAY ONLY \$159.00 - A MONEY-SAVING PRICE!

TO ORDER OR FOR ADDITIONAL INFORMATION

CALL (703) 356-7029 or (703) 847-1743

OR WRITE



PEACOCK SYSTEMS, INC.

Peacock Systems, Inc., 2108-C Gallows Road, Vienna, VA 22180

Reader Service Number 20

C CODE FOR THE PC

source code, of course

	Bluestreak Plus Communications (two ports, programmer's interface, terminal emulation)	\$400
	CQL Query System (SQL retrievals plus windows)	\$325
	GraphiC 4.1 (high-resolution, DISSPLA-style scientific plots in color & hardcopy)	\$325
	Barcode Generator (specify Code 39 (alphanumeric), Interleaved 2 of 5 (numeric), or UPC)	\$300
NEW!	Vmem/C (virtual memory manager; least-recently used pager; dynamic expansion of swap file)	\$250
	Aspen Software PC Curses (System V compatible, extensive documentation)	\$250
	Greenleaf Data Windows (windows, menus, data entry, interactive form design)	\$250
	Vitamin C (MacWindows)	\$200
NEW!	TurboTeX (TRIP certified; HP, PS, dot drivers; CM fonts; LaTeX)	\$170
	Essential resident C (TSRify C programs, DOS shared libraries)	\$165
	Essential C Utility Library (400 useful C functions)	\$160
	Essential Communications Library (C functions for RS-232-based communication systems)	\$160
	Greenleaf Communications Library (interrupt mode, modem control, XON-XOFF)	\$150
	Greenleaf Functions (296 useful C functions, all DOS services)	\$150
	OS/88 (U**x-like operating system, many tools, cross-development from MS-DOS)	\$150
	ME Version 2.0 (programmer's editor with C-like macro language by Magma Software)	\$140
	Turbo G Graphics Library (all popular adapters, hidden line removal)	\$135
	American Software Resident-C (TSRify C programs)	\$130
	PC Curses Package (full System V, menu and data entry examples)	\$120
	CBTree (B+tree ISAM driver, multiple variable-length keys)	\$115
	Minix Operating System (U**x-like operating system, includes manual)	\$105
	PC/IP (CMU/MIT TCP/IP implementation for PCs)	\$100
	B-Tree Library & ISAM Driver (file system utilities by Softfocus)	\$100
	The Profiler (program execution profile tool)	\$100
	Entelekon C Function Library (screen, graphics, keyboard, string, printer, etc.)	\$100
	Entelekon Power Windows (menus, overlays, messages, alarms, file handling, etc.)	\$100
	Wendin Operating System Construction Kit or PCNX, PCVMS O/S Shells	\$95
	C Windows Toolkit (pop-up, pull-down, spreadsheet, CGA/EGA/Hercules)	\$80
	Professional C Windows (windows and keyboard functions)	\$80
	JATE Async Terminal Emulator (includes file transfer and menu subsystem)	\$80
	MultiDOS Plus (DOS-based multitasking, intertask messaging, semaphores)	\$80
	WKS Library (C program interface to Lotus 1-2-3 program & files)	\$80
	Professional C Windows (lean & mean window and keyboard handler)	\$70
	Quincy (interactive C interpreter)	\$60
	EZ.ASM (assembly language macros bridging C and MASM)	\$60
	PTree (parse tree management)	\$60
	HELP! (pop-up help system builder)	\$50
	Multi-User BBS (chat, mail, menus, sysop displays; uses Galacticom modem card)	\$50
	Heap Expander (dynamic memory manager for expanded memory)	\$50
	Make (macros, all languages, built-in rules)	\$50
	Vector-to-Raster Conversion (stroke letters & Tektronix 4010 codes to bitmaps)	\$50
	Coder's Prolog (inference engine for use with C programs)	\$45
	C-Help (pop-up help for C programmers ... add your own notes)	\$40
	Biggerstaff's System Tools (multi-tasking window manager kit)	\$40
	PC-XINU (Comer's XINU operating system for PC)	\$35
	CLIPS (rule-based expert system generator, Version 4.1)	\$35
	TELE Kernel or TELE Windows (Ken Berry's multi-tasking kernel & window package)	\$30
	Clisp (Lisp interpreter with extensive internals documentation)	\$30
	Translate Rules to C (YACC-like function generator for rule-based systems)	\$30
	6-Pack of Editors (six public domain editors for use, study & hacking)	\$30
NEW!	Crunch Pack (a baker's dozen of file compression & expansion programs)	\$30
	ICON (string and list processing language, Version 6 and update)	\$25
	LEX (lexical analyzer generator)	\$25
	Bison & PREP (YACC workalike parser generator & attribute grammar preprocessor)	\$25
	AutoTrace (program tracer and memory trasher catcher)	\$25
	C Compiler Torture Test (checks a C compiler against K & R)	\$20
	Benchmark Package (C compiler, PC hardware, and Unix system)	\$20
	TN3270 (remote login to IBM VM/CMS as a 3270 terminal on a 3274 controller)	\$20
	A68 (68000 cross-assembler)	\$20
	List-Pac (C functions for lists, stacks, and queues)	\$20
	XLT Macro Processor (general purpose text translator)	\$20
	Data	
	WordCruncher (text retrieval & document analysis program)	\$275
	DNA Sequences (GenBank 52.0 including fast similarity search program)	\$150
	Protein Sequences (5,415 sequences, 1,302,966 residuals, with similarity search program)	\$60
	Webster's Second Dictionary (234,932 words)	\$60
	U. S. Cities (names & longitude/latitude of 32,000 U.S. cities and 6,000 state boundary points)	\$35
	The World Digitized (100,000 longitude/latitude of world country boundaries)	\$30
	KST Fonts (13,200 characters in 139 mixed fonts: specify TeX or bitmap format)	\$30
	USNO Floppy Almanac (high-precision moon, sun, planet & star positions)	\$20
	NBS Hershey Fonts (1,377 stroke characters in 14 fonts)	\$15
	U. S. Map (15,701 points of state boundaries)	\$15

The Austin Code Works

11100 Leafwood Lane

Austin, Texas 78750-3409 USA

Free shipping on prepaid orders

acwlnfo@uunet.uu.net

Reader Service Number 04

For delivery in Texas add 7%

Voice: (512) 258-0785

BBS: (512) 258-8881

FidoNet: 1:382/12

MasterCard/VISA



Rafting And Large Screens

Carrying It Off When You're Up In The Air

By Laine Stump

% Redhouse Press
Merkez PK 142
34432 Sirkeci
Istanbul, Turkey

Laine packs up the tent and prepares to return to Turkey for white water rafting. Unbeknownst to the editors, he also discusses RLL hard drives, OS/2, and large screens. (Anything to make a splash, I suppose.)

The scenery is changing once again. Last night I finally tore down the free standing dome tent in my bedroom. I had set it up last Saturday after I bought it from REI in the Twin Cities, but gave in and dismantled it after calling Pan Am to reconfirm my reservation for next Thursday.

That's right. I'm back at it again. I've been stocking up in preparation for my latest Turkish trip and my bedroom looks like a sporting goods store. Having the tent down and bagged up helps, but there's still that stack of orange life jackets in the corner. And the Therma-Rest mattress rolled up next to the six rolls of Ektachrome 200. (The bed was banished to the living room weeks ago.)

Now if my new fiberglass shaft paddles would just show up. I leave in four days and I'm starting to get anxious! And as Dean (Klein) and Earl (Hinrichs) have both told me, I don't want to be up the creek without a (Those guys! They're also the ones who sent a letter to me one Thanksgiving when I was living in Ankara, saying that was the only time of year they thought about me. Right around dinner time, as a matter of fact. Just as they were carving up the)

(A note added Monday morning: the paddles have arrived! Hooray!! Now I can start figuring how to pack them...)

I received an updated letter today from Rich, one of my partners in crime who is already in Turkey. Along with asking me to pick up a pole for his tent from REI, he informed me that *both* of the other two crew members he has recruited are journalists (so is Rich!) Looks like we'll be fighting over the National Geographic credits.

By the time this gets printed, the trip will be over (or aborted). Look forward to an account

of our (hopeful) adventures during our month long expedition on "as many Turkish rivers as possible in one month," or "Midnight Deliverance." (One good thing about it, they don't even *have* pigs in Turkey. It's against the religion.)

What?????

"How will I compute?" you ask. Well, I have some plans. First, I still have the Zenith 181 portable that I bought last fall. While it isn't much on computing power, it does have a nice screen and it can run four hours on its batteries. I can also run it off a car's cigarette lighter (a much better use of cigarette lighters, if you ask me).

My three months at PC Tech haven't left me unscarred, however. As the weeks here in Lake City have slipped by, I've found myself more and more addicted to high speed machinery. But whatever machine I have must be very small and light (easy to carry, easy to smuggle, and look like a portable radio or tire pump).

I looked around and found that what I needed (80286, at least 10 MHz, 2 MB of RAM, 20 MB winchester) wasn't available in battery-powered form, so I started looking around for the smallest AC powered machine I could find (and afford). What I ended up with isn't my idea of the "perfect machine" (after all, it doesn't work on the beach, in the car, or on the river), but it is just about as close as I could come.

First, the cabinet. You've no doubt been noticing those McTek ads the last few months featuring the "portable" case? Well, McTek has found a new case that they're selling for the same price as the case in the ad, but it has a much nicer (larger) screen. Except for the difference in display color, it looks identical to the Compaq 386 portable.

The McTek cabinet has an LCD display of 640 x 400 dots, but in text mode those dots are "doubled" and use the standard, ugly CGA character font (8 x 8) instead of an EGA character font (8 x 16, or something). I'm sure the video controller can handle it, the desig-

ners just weren't thinking. Oh, and while it has a contrast control, it doesn't have one for brightness. The screen is generally too bright. I'd like to tone it down.

It's still the best I've seen, though. Gary Entsminger turned me on to this cabinet, and I'm forever grateful. Well, at least until a newer model hits the streets.

And for brains — do you think I could get away with putting anything other than a PC Tech board in this thing??? I'd never make it out of Lake City alive!

So instead of forfeiting my chance at collecting Social Security, I installed a "passive backplane" board (devoid of anything except sockets and power connectors) in the cabinet. Then I stuffed in a prototype of one of those new "286 on an expansion card" boards that people keep whispering about, but can't openly announce for fear of being reprimanded or having a dirty sock stuffed in their mouth. ("Pssst! It's called the 'X24.' Pass it on.")

This sucker is *fast*, too! As I'm sitting at home on Sunday writing this, my "very early" prototype is still poling along at 10 MHz, 0 waits (shades of Venice), but the newer models are running at 16 MHz, 1 wait state (Monte Carlo) now. And I've been promised an upgrade before I leave on Thursday.

By the time I hit the airport, I'll have a 16 MHz 286, 2 MB of RAM (expandable to 4 MB on the CPU card), 20 MB of winchester (oh, and "by the way," don't forget the 34010 video board — sans monitor, unfortunately) all packed into a little black shoulder bag that will slide right under my seat. The final score? About 1.25 lb. per MHz.

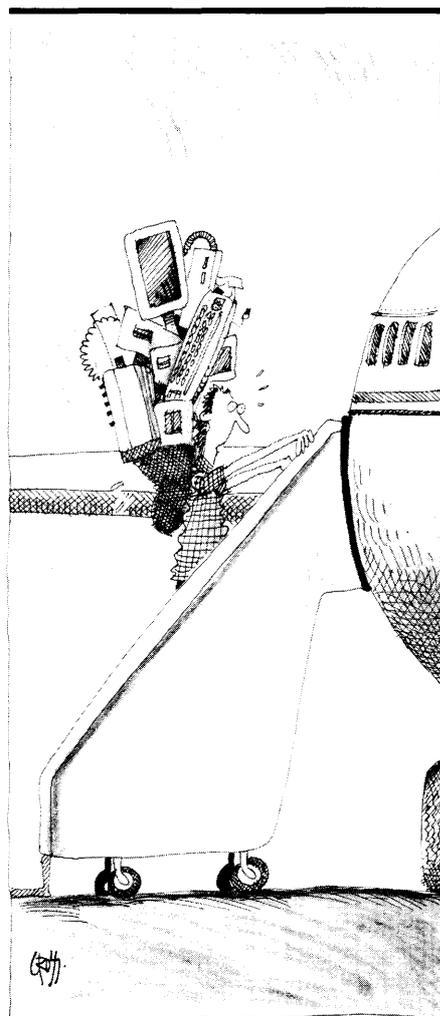
How much did it cost??? About the same as a fully outfitted Avon Redshank inflatable. At least the way I outfit Avon Redshanks.

I think Gary is going to have more to say about portables (and inflation) in this (or the next) issue. Last time I talked to him he was babbling something about Colorado mountains and 286 clones, so I'll leave most of the details to him.

Updates

I have been in communication with Logitech since last issue. They sent me a newer version of the Point editor which fixes some of the bugs I had found. Some are still there, but the programmer says that he's working on them. He also agreed that Point should be made

By the time I hit the airport, I'll have a 16 MHz 286, 2 MB of RAM, & 20 MB of winchester all packed into a little black shoulder bag.



compatible with any size of video display. Especially after I told him how easy it was.

Something was also mentioned about an "Ohwesstu" version, or something like that. He didn't expand on that one, but I'm guessing it must be a special version for programming in some Central African pygmy dialect that uses all glottal stops and tongue clicking

sounds, and traverses the page diagonally when written.

Personally, I can't see much use in something so obscure as that, but I suppose Logitech sees a market in it, otherwise they wouldn't be wasting the time. Or maybe it's part of one of those "Live Aid" assistance programs or something.

They should spend time working on development tools for that new operating system Microsoft released last fall, instead. That's what I think.

Ohwesstu

Being that I'm getting interested in Central African dialects myself (I just picked up a copy of "See It and Say It in Ohwesstu" at B. Dalton's yesterday), I've been talking to a few people about it. Not the drivel about "Is it worthwhile?" or "The OS/2 applications I've seen are sludge!" or even "Well, they're a few years late, I think." I don't care about all that.

Regardless of how good or bad the stuff written for it is, how late it is, etc., the fact is that OS/2 is here. And we've got to deal with it. It does no good to bitch about operating systems development lagging behind the hardware or any of that. It's all under the bridge. Forget about that, look at it as an opportunity.

It may not be ideal, but it has some powerful features and it's an emerging standard. And most important, it exists (finally). And you have to admit that it's better than DOS (I know I'll catch some shrapnel on that one, but that's what I think). *Editor's note: People who live in inflatable boats should avoid shrapnel.*

Anyway, lately what I've been talking to people about is: "Does your machine BOOT OS/2???" For a surprisingly large number of those who purchased early, cheap AT clones, the answer is a devastating "NO!"

Many things can contribute to this incompatibility. Among them are winchester controllers with their own ROM BIOSes, incompatible motherboard BIOSes, incompatible (put your own here). It's really kind of scary.

One of my reasons for wanting a 286 machine for this trip was that I want to do some serious looking at OS/2. Since OS/2 can kind of be equated with "large libraries" and "big programs," I had wanted to use an RLL controller on my machine and get 30 Megs for the price of 20 plus a little. But when I slapped an RLL controller in the box (admittedly, an XT style OMTI controller), the machine refused to have any-

thing to do with OS/2 at all. The AT style OMTI we have wouldn't boot OS/2 either.

I finally became paranoid and switched back to the standard old Western Digital 1002-WA2 (or whatever it is) — that big-ugly-slow controller that, whatever faults it has, is fully compatible with OS/2.

OMTI is rumored to be making a new version of their AT controller, and Western Digital has promised a prototype of their new RLL AT controller to Dean. But in the meantime, I'm stuck with 20 Megs on a 30 Meg drive. (And that's just one example of what's been happening to people all over the country.)

What's my advice? I won't give any pitches about brands or the like here, I'll just say, "Be careful! And get a guarantee!!!" It's not a happy day when you find out that your two grand is down the tubes just because some engineer cut a corner somewhere on the design of your motherboard.

RLL Drives (And Controllers)

Talk of RLL brings me to the subject of the reliability factor of RLL drives. I have heard rumors that others writing in this issue will attempt to badmouth RLL technology, so I thought I should put in my opinion, too. (Like I've said so many times before: "Everyone's entitled to my opinion.")

When RLL drives were first released, I thought they were too good to be true. After all, you can't get vodka from a water buffalo in Des Moines (geez, I love making up sayings).

I was kind of like the guys who thought that Orbel and Wilver Wright were doomed to failure ("If God had meant 611 track drives to have 30 megabytes, he would have given them 6 heads!"). I took the attitude of "wait and see." Or maybe, "point and laugh."

Matter of fact, I nearly openly called my brother a fool when he went out and bought a Seagate 238 RLL drive. That was almost a year ago now, and Cecil's 238 is still going fine (last I heard).

Working at PC Tech, I got my first introduction to RLL on a daily basis. My "home" system has a Miniscribe 8438 (3.5 inch, 30 Meg RLL drive) and an OMTI 3527 controller (SCSI bus RLL controller). Neither has given me any problems, and PC Tech is recommending that drive/controller combination to its customers (along with the OMTI 5527 for those who want a controller to

plug into the PC bus). In the three months I've been here, I haven't heard of a single problem with these drives.

Of course, Miniscribe isn't exactly the cheapest drive on the market. And OMTI 3527's don't go for a song either.

This is what I think: There is nothing wrong with RLL drives and controllers (just like there is nothing wrong with 3.5 inch floppy drives, or 9600 bps modems). You just have to be careful about quality. If you get a good brand from a reliable distributor, it's going to be as solid as any MFM hard disk. But if you buy from a bargain basement, watch out!

And it's not just the drives you have to be careful about. Watch the controllers, too. I don't know much about other brands of controllers, but the OMTI controllers seem to be working fine (on XT class machines). If you find some no-name that's selling for \$20 less, stop and think for a minute about the possible aggravation that \$20 could cause you later on when your system fails at 3 a.m. the night before that big demo at UNISYS.

I guess if cheap, bottom of the line, ugly scum is really what you want, though, I would be wary of buying RLL.

Pop Of The Stack

Anyway, while I was talking to the programmer at Logitech, I decided I should tell everyone how to write programs that could take advantage of large screens automatically. Since it's so damned easy, there's no reason why everyone shouldn't do it. What follows is a (slightly) polished version of what I have sent off in the mail to a couple of people.

Programming For Large Screens

NOTE: The following procedure is compatible with EGA and VGA, as well as the original CGA, MDA, and Hercules (in text mode). It's also compatible with the PC Tech 34010 Monochrome Graphics Adapter. Other manufacturers' adapters should be compatible with this, but I don't know for sure. If they aren't, they should be. Using this method is easy, and it's based on existing standards.

Background

Most large screen text displays are simple extensions of either CGA or MDA. This means that their video buffers start at the same address (B8000 or B0000, respectively) and occupy contiguous words up to the end of the

screen.

Each word consists of a character and an attribute byte. Lines are "NumColumns" words apart (where NumColumns is usually 80, but sometimes 40, and could really be any size, up to 65535, depending on the hardware). Using this standard as a base allows these high performance video adapters to work with software supporting only CGA or MDA.

Since most display adapters in the past were either CGA or MDA, and since software developers (myself included) were lazy, most programs assume the presence of one of these cards, and that ugly number "25" spread like an unchecked plague across hundreds of thousands of lines of code.

Now that many people have screens capable of displaying more lines (the EGA, for example can display 43 lines, the VGA 50, the PC Tech 66, the MDS Genius 66, the Moniterm Viking ??), they're happy. But wait! There is an ugly spectre in the shadows! That elusive computer version of 666. The computing world's "Number of the Beast" - 25. AAIIGGGHHH!!!!!!

Programmers, being the machine hogs they are, saw this problem. And many of them, when they got their shiny new EGA card, decided to do something about it. Since the EGA had a 43 line mode, we were treated with a few programs that could work in 25 line mode, and 43 line mode. Nothing else.

VGA has now become popular. With this popularity comes the expected wave of programs with 25 line, 43 line, and, yes, all new "made for VGA" 50 line mode. Obviously these programs were well written; they can support different screen sizes. Obviously, they're not seeing the whole picture. What happens if I happen to own a machine that has a 42 line display? What about 66 lines? And I haven't even mentioned the possibility of 90 columns, 132 columns, etc.

And I'm being simplistic. What if every card on the market had a different number of columns? More realistically, even if the largest commercially viable (cheap) screen is 43 lines by 80 columns today, what if tomorrow somebody comes up with an incredible new technology? A technology, for instance, that supports multiple hardware font sizes on a screen with a user programmable resolution up to 200 by 200 characters?

Wouldn't you like it if your software had built-in support, already working,

ERAC CO.

8280 Clairemont Mesa Blvd., Suite 117
San Diego, California 92111
(619) 569-1864

AT

Motherboard 6 & 10 Meg
Zero Wait State
8 Expansion Slots
1 Meg RAM On-Board
Math Co-processor Option
Phoenix Bios
200 Watt Power Supply
Hercules Compat. Video Board
Parallel Port
2 Serial Ports Active
Game Port
Clock/Calendar
Hard Disk & Floppy Controller
20M Hard Drive
1.2M 5 1/4" Floppy Drive
360K 5 1/4" Floppy Drive
5061 Keyboard
Case with Turbo & Reset,
Hard Drive Light and
Keyboard Disable Switch
Amber Graphics Monitor

\$1581
EGA ADD \$449
40M HD ADD \$150
6 & 12 MHz ADD \$73

BABY AT

Motherboard 6 & 10 Meg
Zero Wait State
8 Expansion Slots
80286 Processor
Math Co-Processor Option
1 Meg RAM On-Board
Phoenix Bios
200 Watt Power Supply
Hercules Compat. Video Board
Parallel Board
2 Serial Ports Active
Game Port
Clock/Calendar
Hard Disk & Floppy Controller
20M Hard Drive
1.2M 5 1/4" Floppy Drive
360K 5 1/4" Floppy Drive
5061 Keyboard
Mini AT Case with Turbo &
Reset, Hard Drive Light and
Keyboard Disable Switch
Amber Graphics Monitor

\$1531
EGA ADD \$449
40M HD ADD \$150

XT/TURBO

Motherboard
5 & 8 MHz Switchable
8088 — V20 Optional
Optional Co-processor
8 Expansion Slots
ERSO or Bison Bios
640K RAM
150 Watt Power Supply
Hercules Compat. Video Board
Parallel Board
2 Serial Ports Active
Game Port
Clock/Calendar
Hard Disk and
Floppy Controller
20M 5 1/4" Hard Drive
2 ea. 360K 5 1/4" Floppy Drive
AT Style Keyboard
Standard Slide Case
Amber Graphics Monitor

★ ★
\$999
EGA ADD \$429
40M HD ADD \$150
5 & 10 MHz ADD \$21

UNINTERRUPTIBLE POWER SUPPLY \$149

ELGAR MODEL SPR401 — THESE SUPPLIES MAY HAVE SOME MINOR COSMETIC DAMAGE, BUT ARE ELECTRICALLY SOUND. 400VA SQUAREWAVE WAVEFORM. RUN ON INTERNAL OR EXTERNAL 24VDC BATTERY WHEN LINE GOES DOWN. TYPICAL TRANSFER TIME = 12MS.

Battery Supplied — Not Guaranteed

NEW 24V INTERNAL BATTERY \$75

KAYPRO EQUIPMENT

9" Green Monitor \$35.00
Keyboard 75.00
Hard Disk Cable Set (4) 15.00
PRO-8 Mod. to your board 149.00
Host Interface Board 15.00

KAYPRO IC'S

81-189 Video Pal \$15.00
81-194 RAM Pal 15.00
81-Series Character Gen. ROMs 10.00
81-Series Monitor ROMs 10.00

POWER SUPPLIES

0-8VDC 100A Metered \$249.00
Volt & Current Regulated

5V/1A, -5V/2A, 12V/1A,
-12V/2A, -24V/0.5A 9.90

HOURS: Mon. - Fri. 9 - 6 — Sat. 10 - 4
MINIMUM ORDER — \$15.00

TERMS: VISA, MasterCard, Certified Checks, Money Order, NO COD. Visa and MasterCard add 3%. *Personal checks must clear BEFORE we ship.* Include shipping charges. California residents add 6 1/2% Sales Tax. For more information please call.

CPU & SUPPORT CHIPS

MC68000-8 CPU \$8.00
Z80 CPU75
Z80A CPU 1.50
Z80 CTC 1.50
Z80A PIO 2.00
Z80A SIO 5.00
8088 6.50
8089-3 6.50
D8284A 2.50
4164-15 1.90
4164-12 2.10
1793 6.00
1797 7.00
ICL7107 LCD Driver 7.00
6845 5.00
VC3524 Switching Regulators 5.00
1458 Dual Op-AMP70
LM2877P 4W Stereo Amp Dual 2.50
MB81464-15 2.75
2716 3.00
2732 3.25
2764 3.50
27C128-1 9.00
74HC0038
74LS12530
74LS37350
74LS17430

SWITCHERS

5V/9.5A, 12V/3.8A, -12V/8A \$39.00
5V/3A, 12V/2A, -12V/4A 19.50
5V/6A, 12V/2A, -12V/1A 29.00
5V/6A, 24V/1 1/4A, 12V/6A,
-12V/6A 29.00
5V/10A 19.00
5V/20A 24.00
5V/30A 39.00
5V/75A, 12V/8A, 24V/5A 55.00

MISCELLANEOUS

Z80 Controller w/8-bit A/D Conv. . \$15.95
Nidc Pack 12V/5AH 6.50
Joystick 4 Switches 1" Knob 5.50

TEST EQUIPMENT

OSCILLOSCOPES

TEK 7403N/7A18N/7B50A 60 MHz .. 695
TEK 455 50 MHz Dual Trace 595.00

ANALYZERS

TEK 491 10MHz - 40 GHz \$4500.00
Nicolet 500A 1 Hz 0-100 KHz 1800
Biomation 805 Waveform Analyzr . 259.00
HP1600A/1607A Logic Analyzr ... 1000.00

DBASE BOOK OF BUSINESS APPLICATIONS by Michael J. Clifford

Reg. \$19.95 **NOW ONLY \$3.95**

on the day that product was announced? Wouldn't you especially like it if this support were compatible with existing standards as well, and was just as easy to implement (easier even) than the "hardcoded" method?

I humbly submit my method for supporting variable screen sizes in application programs. The techniques used comply with the standards implied by the Enhanced Graphics Adapter Technical Manual. They will work under any version of MS-DOS using any IBM standard card (CGA, MDA, EGA, VGA), as well as any other that follows the simple rule of keeping its "NumColumns" variable in the IBM ROM BIOS dictated location, and the "BottomLine" variable in the location used by the EGA ROM BIOS.

Before I describe this, a disclaimer. I do not attempt to convince you that this standard is the ideal standard. I even think that it is a *bad* standard and that it could be done much better. But, I didn't make the standard. I'm just reporting on it.

I have no idea why NumColumns is a word and BottomLine a byte; I don't have the foggiest why NumColumns is represented as 1 to n and BottomLine is 0 to (n-1). That's not within my jurisdiction. I can merely tell you that they are, and how to use them.

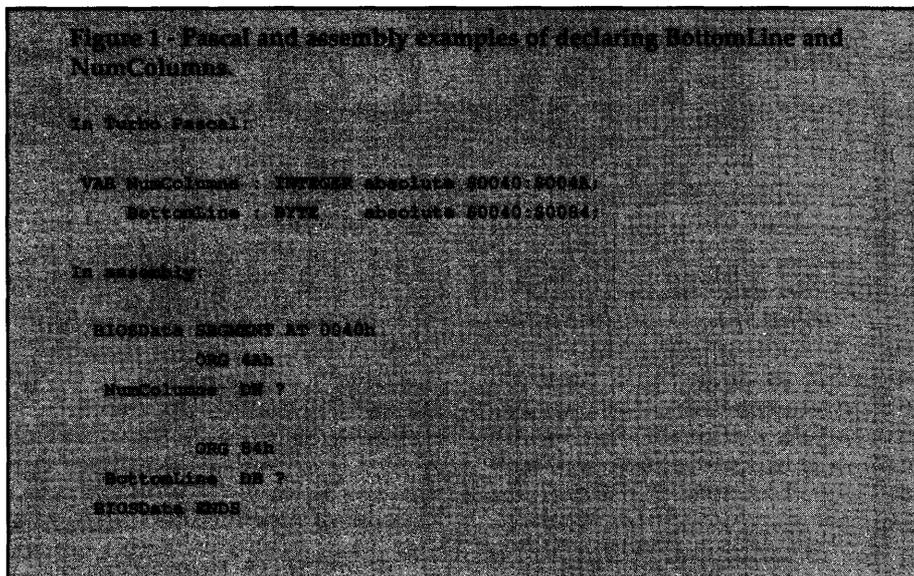
Details

Okay. Let's say you have a program that's going to be doing a lot of screen output. Maybe a few windows that you want to move around. Lots of text. You want to be able to use as much screen real estate as you can get your hands on. What to do....

First — don't hard code *anything!* Not even array bounds. If you do have any array that's dependent on screen size, allocate it at run time (and maybe even be prepared to change its size). This really isn't as hard as it seems. Instead of putting 25s (and 80s) all over your program, just reference a variable.

What variable? BottomLine, of course. Bottomline is a byte stored in the EGA BIOS scratch RAM area at 0040:0084h. It is set equal to the number of lines on the display minus one. So if your display has 43 lines, it will say 42; 65 means 66, etc. Many times, the "-1" is what you want, anyway. Just declare BottomLine as a variable in your program and use it instead of a constant. (See the examples in Figure 1.)

The contents of BottomLine is also returned (along with some other junk)



by the EGA BIOS call INT 10h, function 1130h. You can try looking here for the screen size as well. But many cards, in their efforts to not be mistaken for an EGA, don't implement this call. (On the other hand, there may be some cards that implement this call, but not the BottomLines variable at 40:84 itself! Maybe you should check both...)

There is one exception, though!! Since BottomLine didn't exist until EGA came along, older cards don't store anything at that location (and they don't implement function 1130h either). If this is the case, BottomLine will = 0; just check for this condition and change 0 to 24. (The Award AT BIOS does this for you at boot time).

And what about different column sizes? That's easy. NumColumns is stored in the word at 0040:004Ah. It is also returned by the GetVideoMode ROM BIOS call, INT 10h, function 0Fh. NumColumns is in 1 to n format, (i.e., if you are dealing with an 80 column monitor, NumColumns = 80). NumColumns was supported even in the original IBM ROM BIOS, so it should always be set correctly, no matter what kind of card is installed.

Now we know how big the screen is. Where is it? Usually, video cards that are pretending to be a large CGA or MDA will just use the same mode settings as their smaller cousins. Just do INT 10h, function 0Fh; if the returned mode is 7, the video buffer is at address B0000. If the mode is anything else, it's probably B8000.

One exception is the MDS Genius, where modes 8 and 9 are 66 line B0000 modes. Note that I am not accounting for using display pages other than page 0. If you want to do any of that stuff

(I've never found the need), you're on your own.

One more thing. Some video cards reset back to 25 lines whenever you do a mode change (INT 10h, function 0), so don't set the video mode unless you absolutely have to. It's reasonable to figure that whoever is running your program has a liking for the video mode they are using when they enter the program. Don't screw up a good thing.

List Of Things NOT To Do

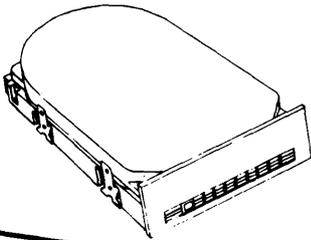
Do not do massive "compatibility checks" to assure that the installed video card actually can display the number of lines given in BottomLine. If the card couldn't display it, BottomLine wouldn't have been set in the first place.

You shouldn't, for example, do a VGA equipment check and then refuse to display 50 lines just because you can't find the palette register or anything like that. Take the word of the BIOS. It knows what it's doing. (Are you listening, Microsoft CodeView, Editor, and WORD authors???)

If no specific screen size is set, you can do other checks to look for certain cards that don't support this standard (MDS Genius, for example, does not). Only do this *after* the standard method has failed to reveal a screen size other than 80 x 25.

Do not assume a fixed screen size based on a mode command you send to the video BIOS. For example, don't do an INT 10h, function 1112h (set 43 line mono text on an EGA) and then assume that the screen now has 43 lines.

In this example, some cards do not support this function, so their screen



20 MEG HARD DRIVES

SEAGATE Model ST-225

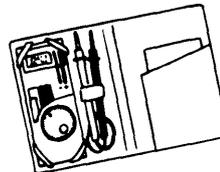
\$219.95 (full 6 Mo. Warranty)
(less face plate)

Western Digital Controller For Model ST-225 Drive
\$69.95 x/cable

Yes!!! These are for IBM compatible machines

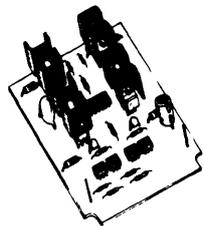
5.25" HARD DISK CONTROLLER CARD

For any Modern 5.25" Hard Drive
\$89.95 w/cable



BECKMAN DM78 MULTIMETER

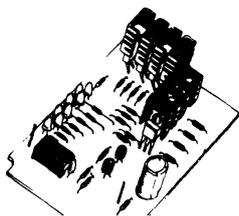
- Pocket Size
 - A-C, D-C Volts & Ohms
 - Continuity Beep Tester
 - Autoranging
 - L.C.D. Display
 - Carrying Case
- Our Biggest Seller!
\$29.95



D-C Motor Controller

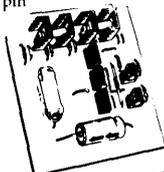
- Control 2 D-C motors with a computer or other logic source
 - For motors rated 6-24 VDC
 - Control forward/reverse/run/cw/ccw/stop
 - Up to 6 Amp starting surge, 4 Amp cont.
 - Dynamic breaking (capable)
 - Will also run most 4-lead stepper motors
- 29.95

Techna-Kit



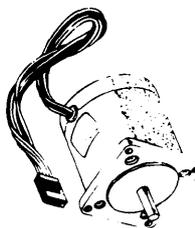
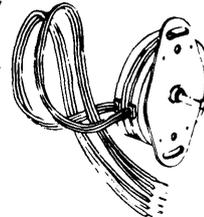
USMD-C

- Control standard 6-lead stepper motors with a computer or other logic source
 - For motors rated 1.7-12.0 VDC
 - Optical Isolation
 - Control: forward/reverse/step rate/stop
 - Industry standard 22 pin edge card connector
- \$29.95



STEPPER MOTORS

Copal #SP-57
1/4" Shaft,
7.5 deg./step,
36 Ohm, 12
VDC
\$6.95



Superior #
M061-FD02
1/4" Shaft,
35 oz." torque,
2.1A., 5VDC
\$9.95

XT CLONE

Building?
Upgrading?

We Stock The
Parts And
Hardware

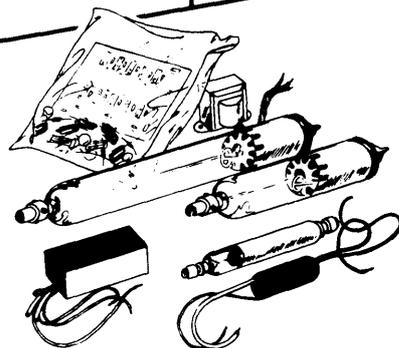
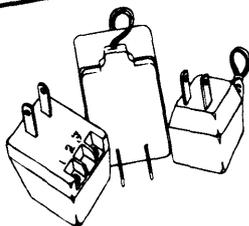
You
Need

AUTOMATION CENTER

- Use your computer to provide automation
 - 8 separate driver ports per card
 - 8 TTL/Cmos inputs
 - 1 user defined sense switch
 - 6-24 VDC
 - 4 Amps/driver (max current)
- \$29.95

REPLACEMENT WALL TRANSFORMERS

- | | | |
|------------|--|--------|
| #45-753 | Universal Out Put | \$8.40 |
| | 3 volt - 4.5 volt - 6 volt - 7.5 volt - 9 volt - 12 volt | |
| | @ 500ma DC | |
| #45-773 | 16 volt @ 1.1A AC | \$8.40 |
| #E-54749 | 4.5 volt @ 265ma DC | \$2.49 |
| #WP-410608 | 8 volt @ 750ma AC | \$2.49 |
| #YS-4510 | Nicad Charger 1.5 Volt @ 400ma DC | \$2.49 |



Lasers

- | | |
|------------------------------|---------|
| 5 MW Laser Tube | \$89.95 |
| Power Supply Kit (115VAC) | \$69.95 |
| Power Supply (wired) (12VDC) | 119.95 |
| 1 MW Laser Tube | 119.95 |
| Power Supply (Bat. Pwrd.) | 99.95 |
| 7 MW Laser Tube | 149.95 |
| Power Supply (12 VDC) | 119.95 |

(These lasers are brand new and guaranteed to have a cosmetic defect or not meet manufactures full specifications. All are tested in our lab to insure your satisfaction.)

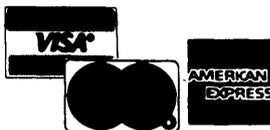
WARNING:

Voltages present and use by lasers can be lethal...Permanent eye damage could result from direct exposure to an oncoming laser beam. Only those persons qualified to handle such potentials should do so.

 **united products corporation**
DISTRIBUTORS OF ELECTRONICS SINCE 1968

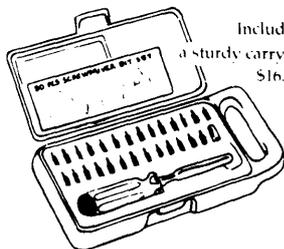
1123 VALLEY STREET • SEATTLE, WA 98109-4425

PHONE: (206) 682-5025 FAX: (206) 682-5593



M-F 9-6
SAT 9-5

30 PIECE SCREWDRIVER BIT SET



Includes
a sturdy carrying case
\$16.95

O
U
R

20TH
Year

Eco-C88 C Compiler with Cmore Debugger

Professionals prefer the Eco-C88 C compiler for ease of use and its powerful debugging features. Our "picky flag" gives you nine levels of lint-like error checking and makes debugging easy:

"I'm very impressed with the compiler, editor, and debugger. I've tried quite a few different compilers for the PC and have given up on all of the others in favor of yours. . . I've gotten to the point where I download C code from a DEC VAX/VMS system just to be able to compile it with the picky flag set at 9. It finds lots of things VMS totally ignores. . ."

JS, Oak Ridge, TN

The Eco-C88 compiler includes:

- A full-featured C compiler with 4 memory models (up to 1 meg of code and data) plus most ANSI enhancements.
- Without a doubt, the best error checking you can get. We catch bugs the others miss, making you much more productive.
- Cmore is a full-featured source code debugger, not some stripped-down version.
- Robust standard library with over 230 useful (no "fluff") functions, many of which are System V and ANSI compatible. Full source is available for only \$25.00 at time of order.
- CED, a fast, full screen, multiple-window program editor with on-line function help. You can compile, edit, and link from within CED.
- cc and mini-make utilities included that simplifies the most complex compiles.
- Users manual with over 150 program examples (not fragments) to illustrate how to use the library functions.
- Fast compiles producing fast code.

Our Guarantee: Try the Eco-C88 compiler for \$99.95. Use it for 30 days and if you are not completely satisfied, simply return it for a full refund. We are confident that once you've tried Eco-C88, you'll never use anything else. Call or write today!

Orders: **1-800-952-0472**
Info: **1-317-255-6476**



Ecosoft Inc.
6413 N. College Avenue
Indianapolis, IN 46220

ECOSOFT

Reader Service Number 9

Figure 2 - Changing Turbo Pascal 3.02 to allow larger Window()

```
C>debug turbo.com

-e16B
ssss:016B 19 42<enter> ; if the originals don't match,
ssss:016C <enter>      ; you have a different version - DON'T PATCH
-a38F
ssss:038F 19 42<enter> ;the 1st patches GotoXY()
ssss:0390 <enter>      ;the second patches Window()
-w<enter>
-q<enter>
```

size would remain at its previous value. For other cards, even more lines could be displayed. As a matter of fact, function 1112h actually means "Use 8 point typeface." The only reason it ends up being 43 lines on an EGA is that the EGA is scanning 350 lines, and $350/8 = 43.75$.

On a card with 400 scan lines (VGA), it would mean 50 lines of text, and on one with 1024 lines it would mean 128 lines. If you must use these functions, you should call the function and then recheck the size stored in BottomLine. That way you will always come up with the correct answer, even after IBM announces the EVGA+/2 Model 71, or whatever.

Bottom Line (No Pun Intended)

To determine number of columns, look at (word) 0040:004Ah or use INT 10h, function 0Fh and use the value returned in AH (note that this is only a byte!).

To determine number of rows, look at (byte) 0040:0084h and add 1, or use INT 10h, function 1130h, and add 1 to register DL. If those numbers are both 0, assume 25.

To determine the start of the video buffer, do INT 10h, function 0Fh. If the mode (returned in AL) is 7 then the buffer starts at B0000, if other numbers then it begins at B8000. (One known exception is MDS Genius — Modes 8 and 9 are 66 line at B0000).

Only look for special cards if these tests fail to find anything other than "normal" values.

Recheck the screen size and video mode any time your program does a mode set or executes another program or library routine which modifies the screen. It may change. If someone wants to switch from color 43 lines to mono 66 lines by escaping to DOS and doing a mode command, you should let them. (Your program should then use the new settings.)

Let me know about programs which support large screens in this manner. I will gladly be a beta tester, and will write about them if they work well. A growing number of us own large screens, and we're looking for software that supports them. (Why do you think we bought big screens in the first place???)

A partial list, by the way, of programs that I know to support large screens (at least up to 66 x 80) is:

Brief 2.0

WordStar 4.0

WordPerfect 4.2

Turbo Pascal 3.02 (use patch in Fig. 2)

Some other programs work in large screen modes, but only for a selected few cards (i.e., Microsoft WORD).

Good Luck, and Happy Programming. See you on the river.

COMING SOON: MASM 5.1 and The Joy of CodeView.

◆ ◆ ◆

If You Don't Have WindowDOS 2.0, You're Wasting Time!!

"When Baba Ram Dass said "Be here now, remember," designers of hard disk utilities should have paid heed. A powerful manager like XTREE can track files and subdirectories and execute DOS commands, but it isn't memory resident. Handy pop-up DOS commanders like PopDOS may be here now, but they lack the power of a full-fledged disk manager. After much meditation, the developers of WindowDOS 2.0 have come up with the best answer yet to the guru's paradox.

Until now, the closest thing to a real RAM-resident disk manager was version 1.0 of WindowDOS. If offered a full-screen pop-up menu and could rename, copy, and delete files. But it couldn't move files, format disks, or rename subdirectories—which XTREE can. Now version 2.0 is here, and it's a winner. Its RAM resident (using less than 50K) but offers all the power of a nonresident disk manager."

—Patrick Marshall, WindowDOS 2.0 Product Review, PC World, May, 1987

Once you've experienced the convenience of instant access to DOS commands, you'll never be satisfied with returning to DOS to list files, format disks, or copy, rename, or erase files. Nor will you be happy with a DOS shell, because shell programs are just as inaccessible as DOS when you are using an application program. **Only one program combines memory-residency with the power of a full-featured disk manager: WindowDOS Version 2.0.**

Features Not Found In DOS

- ◆ Sort directories in 8 ways--or not at all
- ◆ Copy, erase, and move groups of files
- ◆ Find any file in seconds
- ◆ Display default directory of any drive with a single keystroke
- ◆ Display graphic tree
- ◆ Global copy & erase commands
- ◆ Copy function prompts you to insert another disk when necessary
- ◆ Display hidden files and subdirectories
- ◆ Display file contents in various formats and page forward/backward
- ◆ Display Wordstar files in readable format
- ◆ Unique RAM Environment function shows name, size, location, and interrupts of every program in memory
- ◆ Rename subdirectories for instant reorganization
- ◆ Hide and unhide subdirectories
- ◆ See and change file attributes
- ◆ Send control codes to printer
- ◆ Switch default printer
- ◆ Password "lock" your system
- ◆ Set AT Real-Time Clock
- ◆ 5-minute screen-blanking function
- ◆ Input response macros

Enhances These Functions

- ◆ Format disks (faster than DOS)
- ◆ Make and erase subdirectories
- ◆ Copy, rename, and erase files
- ◆ Copy files to printer or COM ports
- ◆ Display disk free space and other media information
- ◆ Check and set the time and date

Benefits

- ◆ **Saves Time**—No waiting to exit or reload programs. Instant access to DOS functions whatever your current task. Easily saves 10 or more minutes a day.
- ◆ **Comprehensive**—Broad range of commands, including many not supplied by DOS. Satisfies the needs of both new and advanced users.
- ◆ **Simplifies DOS**—No need to remember exact DOS commands. Intuitive interface and "point and shoot" design saves keystrokes and prevents mistakes. Group file "tagging" avoids the drudgery of repetitive commands.
- ◆ **Security**—Capability to hide/unhide subdirectories, password "lock" a computer, and check for unwanted programs in RAM helps secure data and prevent unauthorized access.

WindowDOS 2.0 Addresses "RAM Cram" Like No Other Program!!!

1. Designed specifically to be loaded first, unlike most memory-resident programs that insist on being loaded last.
2. Uses a hot key combination that does not have an associated ASCII value—prevents key conflicts with other programs.
3. Unique RAM Environment function lets you monitor the locations, memory costs, and interactions of all programs in memory, including the currently running program. Great for power users/developers.

Other Information

- ◆ Not copy protected
- ◆ Uses only 51K of memory
- ◆ Supports EGA & Hercules
- ◆ Runs memory-resident or as a stand alone program
- ◆ Uninstall command
- ◆ PC/XT/AT/100% Compatibles
- ◆ **Order Today--Only \$49.95**

WindowDOS Associates • Box 300488-C • Arlington, Tx 76010 • 817-467-4103



Indexing Magazines

Finding An Article The Easy Way

Anthony Barcellos

P.O. Box 2249
Davis, CA 95617-2249
(916) 756-4866

Those who have seen it — and survived the experience — say that Tony's home is decorated by towering stacks of books and magazines. When Microsoft chairman Bill Gates visited Sacramento last year, he was surprised to see a 1984 TIME magazine that featured him on the cover. Tony had found it on the floor of his study. How do you improve information retrieval from a collection like that?

Devoted computer users collect magazines just about as fast as they collect floppy disks. After all, you never can tell when you might need that little assembly language routine from the Power User's column tucked away in the back of — now what issue was that?

Trouble is, magazines are harder to search for information than floppies. *PC Magazine*, for example, used to make it even harder by not publishing an index. So how do you find that article you almost remember? I have tried both sequential access (page turning) and random access (flipping the magazine open), but neither technique has much to recommend it. Especially since I usually found myself looking through the wrong issue.

Let Your PC Do The Walking

Early on, I realized that my computer might be of help. I started a small database in which I would note articles that looked interesting. By entering the title, author, magazine issue, and page number, I had an electronic index to my collection of periodicals. Unfortunately, that project turned out to be unwieldy and I fell further and further behind. Then I discovered a new software product that made me give it up entirely.

Lots of other people had the same idea, but the father and son team of Ralph and Randy Nelson applied themselves more diligently than most. Letus A-B-C was born: the Literature Evaluation Table and User Service. (Ralph Nelson admits that the acronym was created before the full name it ostensibly abbreviates.)

Using ButtonWare's PC-File, the Nelsons began to compile a methodical index of computer magazine articles. Their first disk covered 1982 and included information from four magazines: *PC Magazine*, *Softalk for the IBM PC*, *Personal Computer Age*, and *BYTE*. (Two of those magazines no longer exist.)

Two disks covered all of 1983, but then the Nelsons switched to quarterly publication of their database. The PC publication boom was on and they were trying to keep up.

The Paperless Index

Through the end of 1987, Letus A-B-C occupies a total of 39 disks. The Nelsons mail two or three new disks every quarter to their subscribers. Refined and augmented with time, Letus has migrated from the original 320K floppies to 360K (can 1.2 megabytes be far away?) and updated its file format to conform with upgrades to PC-File.

Letus provides a separate database each quarter for each magazine. For example, PCW87D is the index to *PC World* for the fourth quarter of 1987. All of the databases are defined by an identical PC-File header file. Here's the header PCW87D.HDR:

```
keyword(s) 65
author(s) 45
title 65
magazine 8
issue 25
page 4
1st line 65
2nd line 65
3rd line 65
```

As you can see, Letus provides a 65-character field for keywords, 45 characters to identify the authors, 8 characters for the name of the magazine, and so on. The Nelsons use three comment lines at the end to describe the contents of the article.

Facts On File

Ralph and Randy Nelson have followed the ebb and flow of microcomputer magazines over

the past five years. They currently scan and index the articles for *BYTE*, *InfoWorld*, *PC Magazine*, *PC Resource*, *PC Tech Journal*, *PC Week*, *PC World*, *Personal Computing*, and *Programmer's Journal*. (I can, of course, think of at least one extremely important magazine they're missing!)

The Nelsons even record interesting letters to the editor as well as the regular articles. Letus A-B-C really goes from A through Z when it comes to completeness.

While Letus is scarcely a shareware dynamo when it comes to sales through software libraries, it completely fills the niche that the Nelsons have devised for it. Letus has created its own category and no one has challenged it yet. No one need bother. One good index is all it takes to fill the bill, and Letus is a good index.

Thumbing Through The Index

To generate a quick example, I jumped into PC-File and asked for a search in the *PC World* database on the keyword "shareware." The following display appeared on-screen. It gives complete information on the referenced article, including other keywords under

which it is indexed.

Record # 112

keyword(s) J. Button, Button-ware, shareware, PC-File
author(s) E. Bender and M. Hogan
title INDUSTRY OUTLOOK - "Share and Like"

magazine PC World
issue 87/10 Oct v.5 n.10
page 122

1st line Discussion of the success of shareware software developers,
2nd line including Buttonware.
3rd line

Of course, PC-File lets you continue the keyword search after you have viewed each matching record — until you've scanned the entire database.

Finding The Key

For user convenience, the Nelsons create a file for each database that contains an alphabetical listing of all keywords used to describe the articles. Thus one can quickly search PCW87D.KWD (the keyword file) for *PC World* with a utility like Vernon Buerger's LIST before entering PC-File to

do a search.

The dedicated user can merge individual databases to create a large master index (PC-File supports this directly) and concatenate the keyword files accordingly. Since all of the databases have the same structure, it's easy to do. One Letus aficionado I know has moved it into dBASE III Plus.

How Much Lettuce?

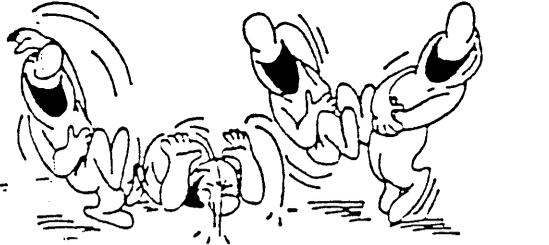
The annual subscription fee for the 1988 Letus disks is \$125, the same as it was in 1987. A prodigious amount of work goes into this disk-based magazine index. Writers, researchers, and magazine junkies like me are the target audience. For those of us who need this kind of reference tool, it's an incredible bargain.

Letus A-B-C

3790 El Camino Real, Ste. 2006
Palo Alto, CA 94306

Laser Blast

How would you like to use a laser to get the most complete access imaginable to the vast hoard of shareware? It's now possible, in the form of a CD-ROM



YOU WANT THE SOURCE?!

WELL NOW YOU CAN HAVE IT! The **MASTERFUL DISASSEMBLER (MD86)** will create MASM compatible source code from program files (EXE or COM). And the files are labeled and commented so they become USEABLE. MD86 is an interactive disassembler with an easy to use, word processor like interface (this is crucial for the REAL programs you want to disassemble). With its built-in help screens you won't have to constantly refer to the manual either (although there are valuable discussions on the ins and outs of disassembling which you won't want to miss).



MD86 is a professionally supported product and yet costs no more than "shareware". And of course, it's not copy protected.

MD86 IS ONLY \$47.50 (\$1.50 s&h) plus tax.

C.C. Software, 1907 Alvarado Ave., Walnut Creek, CA 94596, (415) 939-8153



HOMEBASE™

"An optionally memory-resident desktop organizer."

INCLUDES:

- Calculator
- DataBase
- Text Editor
- Communications Program
- Things-To-Do List
- Cut & Paste 4 Items
- Autodialer
- Calendar
- Alarm Clock
- DOS Services
- Expense Tracker

It's no wonder that *HomeBase™* was chosen Software Product of the Year by *PC Magazine!* *HomeBase™* has just about everything for everybody, and it is always available at the press of a key.

If you have a hard-disk system, *Homebase™* takes just 80K of RAM! If you use a computer, you need *HomeBase™!*

\$89⁹⁵

Just give us a call at:
1-800-523-0764

IN CALIFORNIA 1-800-325-5336 or (408) 559-4545

VISA, MasterCard accepted or mail your check to:
C.C. Software, Box 89000, San Francisco, CA 94189-1700

SHAREWARE DISK AVAILABLE FOR \$10.00

Reader Service Number 87

© Copyright 1988, Telemarketing Resources, Inc.

laser disk from PC-SIG in Sunnyvale, California.

PC-SIG has placed over 550 megabytes (there's no decimal point in that "550") comprising 15,000 separate files on a single compact disk. It's their entire software library of over 900 volumes (conventional 360K diskettes) in one small package.

Of course, the price is not quite as small. You may buy the PC-SIG CD-ROM Library for \$295, on a one-time purchase basis. For \$495, you obtain not only the compact disk, but the next update as well.

"Every update is a brand-new disk," says PC-SIG sales assistant Robin Plotts. "We add one hundred to two hundred new programs every time. An update comes out about every six months, and the next version is due in April or so." PC-SIG has thus far released four versions of the CD-ROM.

Few PCs presently sport CD-ROM drives as peripherals, so the one-disk library is not exactly a commodity item yet. Even so, it does offer a remarkable opportunity to collect the entire contents of PC-SIG's extensive shareware library in a convenient form. (Or would you rather juggle a thousand floppies?)

Not everyone is as delighted with the CD-ROM library as its creators. Some shareware authors regard the one-disk library as a step backward.

"The PC-SIG CD-ROM is disliked by some shareware authors because it 'freezes' our programs," says Neil Rubenking, author of Piano Man and a leader of the San Francisco PC Users Group. "Even if an author makes vast improvements to a program, PC-SIG will still have the old version on their CD-ROM. In 'real' life, shareware programs get updated all the time."

I asked PC-SIG librarian Brian Tuck about Rubenking's comments. "I have authors that update once a month," he acknowledged, but Tuck estimated that only about twenty-five authors were so prolific. "Out of the 900+ floppy disks that we have in the library," he continued, "each new CD-ROM contains about one hundred updated disks."

If one absolutely needs the very latest release of a program, Tuck suggests the floppy disk version, which is updated on a continuous basis. Since Rubenking guesses that the average update period for a shareware program is one year, the six-month update cycle of the PC-SIG CD-ROM may be

reasonable. (Of course, the one-year average incorporates a huge range from the monthly update to "once in a blue moon.")

PC-SIG publishes *Shareware Magazine*, which provides regular information on the updates to their growing software library. Plans call for the magazine to go from bimonthly to monthly sometime in 1988.

PC-SIG's annual membership fee of \$20 includes a subscription to the magazine and notices of library updates. A special membership of \$39 includes any five disks of your choice. Disks normally sell for \$6 each, unless they are part of a special package deal. Any PC-SIG diskette is available in either 5.25" or 3.5" format.

PC-SIG
1030D East Duane Avenue
Sunnyvale, CA 94086

Toll-free phone order line:
(800) 245-6717
(800) 222-2996 (in California)
For information or local orders:
(408) 730-9291

◆ ◆ ◆

CIRCUIT BOARDS

PCB-Edit... creates multi-layered PCB's with ease. The program includes solder mask and legend ink support, plotter and printer output, and one of the fastest CAD artwork layout packages for the IBM. Features include, 1 mil resolution, over 20 different pad styles, unlimited trace widths, plot traces at any angle, CGA and EGA support, ASCII netlist input, text support...much more.

ONLY \$99.95

After you have created your circuit board layout, send your data files to us, and we will make your double sided, plated thru holes circuit boards for only \$1.00 per square inch in single quantity. No set-up charges for PCB-Edit files, \$25.00 set-up charge for other artwork. Silk screening and gold plating available for additional charge.

SCHEMATIC DRAWINGS

PCB-Scem.. the CAD package for drawing scematics on Big Blue's machines. Supports full component libraries, rubber banding, auto part numbering, output to printer or plotter. Netlist support for PCB-Edit provided.

ONLY \$ 99.95

LOGIC ANALYZER - STEPPER

LOGATEST... 32 channel logic analyzer for the IBM. 16 bit trigger word, 80 nano second sample time.

BUILT - \$399.95 BARE BOARD - \$99.95

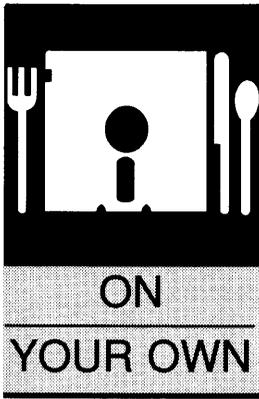
STEPPER... 3 axis stepper motor controller. Plugs into your parallel printer port.

BUILT - \$179.95 BARE BOARD - \$49.95

Call or write --

EM ENTERPRISES
PO BOX 3228
SIERRA VISTA, AZ 85636
(602) 458-4065

See you at SOG!



A Dog & Pony Show

Great Way To Get Feedback

Gary L. Scott
Decision Technology
P.O. Box 5040
Aloha, OR 97007

So you've got a winning idea. You're sure of it. However, it would be a big mistake to sit down and start coding. You haven't done your listening yet.

Last issue we talked about the development of a features matrix to help identify the features you'll need in your new product. If you're like me, you now have a list of features long enough to guarantee that you'll be coding well into the year 2000. This month's article will discuss how to pare the list down and develop a customer-oriented product.

Prioritize

First, prioritize the features list. Many people will tell you that a dart board is a useful instrument for this job. But, our goal is to develop a saleable and profitable product, so let's try something more organized.

The first step is to prototype the user interface. You could use an interpreted language such as QuickC or QuickBASIC, but I prefer Dan Briklin's Demo Maker.

The idea here is to evaluate the "look and feel" issues. You want a product that can be demonstrated. It isn't necessary to prototype the product down to the last comma. What you want is a program that will show all of the major screens as well as several of the underlying screens. The object is to allow the potential user to work through the screen linkages in as realistic a fashion as possible.

The second portion of the demonstration package is the user manual. Yes, I am suggesting that you write a preliminary user manual before you lay one line of code onto the screen. Writing the user manual and developing demonstration software will go a long way toward pointing out problem areas.

Remember, good software is consistent; ALT-C does the same thing in all screens. Nothing is more frustrating to a user (except maybe nonfunctional software) than inconsistencies in the human interface.

One of the strengths of Macintosh software is that Apple imposed certain constraints on the human interface. So, it's consistent, not only within a program, but also among programs.

Developing a demo package and a user manual before writing any code may seem like a lot of work. It is. But it will save you time in the long run. Avoiding just one major rewrite will more than pay for the effort. Another benefit of producing this demo package is that when it comes time to write the final manuals, they'll already be organized.

Also, when you go out to bid for writing the manuals, you will be able to save yourself big money by cutting the time the writer will need

Writing a manual before starting to code the product might seem backward. It's really not. It's smart.

to get up to speed (assuming you keep the demo package current during the development). I strongly recommend this because you will be constantly showing the demo to potential customers, employees, and maybe even an investor or two.

Show And Tell

Once you have your demo version running bug free and your preliminary user manual done, it's time to put together a "dog and pony" show. How extensive you want to be with this depends on where you'll be doing your market research. Obviously, if your product will sell to professionals for use in their

Software Developers

We need your program!

Do you have a program that's good enough to sell, but don't want the problems or financial risk of producing, typesetting, printing, packaging, warehousing, marketing, distributing and supporting a product?

Why start your own software house?
We've done it for you!

Merlin Publishing Group
is now accepting submissions of micro-computer software for publication.

You get:

- initial cash payments
- generous royalties
- to spend your time programming

But we can't help you if you don't submit. Call or write today for our submission guideline kit.



1240 Johnson Ferry Place, Suite A10
Marietta, GA 30068
(404) 977-6034

P. S. See us in Atlanta at COMDEX-Spring '88

Reader Service Number 35

ZRP/M™ creates

Z280®

CP/M©2.2 compatible

IBM PC

ZRP/M is an **operating system** combined with a Z280 emulator. Either standalone or with DOS present, ZRP/M provides the solid base of a genuine operating system reliably distinct from the facade created by an MSDOS interface. **All 2.2 system and CBIOS calls are supported**, 56.5k TPA, file date and time stamping, fast virtual disk, iobyte redirection, terminal emulation, color console display, auto relog, COM path, SAVE anywhere, single key phrase recall, built-in access to DOS drives. SETDISK redefines a drive to any of over 80 CP/M formats. System disk with manual \$129. Shipping \$5 (\$10 nonUS)  



118 SW First St. - Box G
Warrenton, OR 97146
(503)861-1765

business, the level of show will need to be much higher than for a new software tool.

Remember that the people you demo your product to are going to use the quality of your dog and pony show to judge your sincerity and intent (and the quality of your final product). If all you have to show is a few scratchings on a napkin and your name is Knuth, you probably will be treated seriously. But if your name is Scott, you probably won't.

The minimum level presentation is a running demo program, hard copies of all screens, a data dictionary of all user entered variables, and a laser printed manual with *no* spelling errors and no gross deviations from commonly accepted English grammar.

The next step is to practice your presentation. Just because you wrote the demo packages, don't assume that you can demo it. Practice on a friend, wife/husband, or a schoolmate.

While you are practicing and refining your presentation, identify who should see your demo.

If you are developing a specialized package, you probably already know several people in the field who would be happy to look at your ideas.

Contact these people and be up front and honest about why you want to talk with them. Make it clear that your intent is not to sell them anything. You are simply looking for evaluations of your product by "leading" members of the business community.

Explain that you will not take a lot of their time and, in fact, that your presentation takes less than forty-five minutes. But be prepared to spend several hours talking. When the presentation comes to an end, be sure and ask if you can come back at a later time and discuss the enhancements you are incorporating as a result of the interviews. If your audience was impressed, you should have no problem scheduling a second presentation.

After each presentation, you need to evaluate your results. Did someone suggest a new feature or change that you should incorporate? If so, make the change before your next presentation.

After three or four presentations to different individuals or groups, you will start to get a feeling for the features important to your potential user base. You can now make an intelligent decision on what features need to be in the first release and which features can be

treated as an upgrades.

Keep It Confidential

An important topic that needs to be discussed here is confidentiality. You are going to be showing product proposals around at a very delicate stage. If you show your product to the wrong person, you might find your product on the market months before you're ready.

If someone really wants to steal your ideas, there is very little that you can do about it. But there are some simple steps you should take to protect yourself. Have all of the people you show your product to sign non-disclosure agreements *before* you show them the product. Don't leave a paper trail. Be sure that you get back all of your copies before leaving a presentation. Ask the participants not to take notes during the presentation.

If your product is in a highly competitive field, encode an identification into each manual or working document. If you are concerned about the potential theft of your ideas by Xerox, make sure that all documents that leave your office are printed on dark red paper. It is almost impossible to reproduce dark red

ICs PROMPT DELIVERY!!!

SAME DAY SHIPPING (USUALLY)
QUANTITY ONE PRICES SHOWN for MAY 3, 1988

OUTSIDE OKLAHOMA: NO SALES TAX

640K MOTHERBD UPGRADE: Zenith 150, IBM PC XT, Compaq Portable & Plus; hp Vectra

DYNAMIC RAM			
SIMM	1048Kx8	100 ns	\$450.00
1Mbit	1048Kx1	100 ns	36.50
51258	* 256Kx1	100 ns	13.50
41256	256Kx1	60 ns	14.50
41256	256Kx1	80 ns	14.00
41256	256Kx1	100 ns	13.50
41256	256Kx1	120 ns	13.50
41256	256Kx1	150 ns	12.50
41264	+ 64Kx4	120 ns	16.95
EPROM			
27C1000	128Kx8	200 ns	\$37.50
27C512	64Kx8	200 ns	14.95
27256	32Kx8	250 ns	7.50
27128	16Kx8	250 ns	6.95
STATIC RAM			
43256L-12	32Kx8	120 ns	\$16.50
6264LP-12	8Kx8	120 ns	10.75

* COLUMN DRAM + 2 PORT VIDEO RAM 80387-8 80387-16 80387-2 80387-4 80387-6 80387-10 80387-20 \$160.00 \$245.00

OPEN 6 1/2 DAYS, 7:30 AM-10 PM: SHIP VIA FED-EX ON SAT.

WE EXPORT ONLY TO CANADA, GUAM, PUERTO RICO & VIRGIN ISLANDS

SAT DELIVERY INCLUDED ON FED-EX ORDERS RECEIVED BY: Th: Std Air \$6/3 lb Fr: P-1 \$10.25/1 lb

MasterCard/VISA or UPS CASH COD
Factory New, Prime Parts μ P80
MICROPROCESSORS UNLIMITED, INC.
24,000 S. Peoria Ave., (918) 267-4961
BEGGS, OK. 74421

No minimum order. Please note that prices are subject to change. Shipping & insurance extra, & up to \$1 for packing materials. Orders received by 9 PM CST can usually be delivered the next morning, via Federal Express Standard Air @ \$6.00, or guaranteed next day Priority One @ \$10.25!

Reader Service Number 37

masters.

Following these simple steps will decrease the chances of someone using your ideas and increase your chances of winning any possible litigation that might develop.

Alpha Release

We will refer to the first release of your product as the alpha release (what would we have done without the Greeks?). This is the first working release that you will use for demonstrations.

Once you have made sure that there are no major problems with this release, it is time to hit the dog and pony circuit again. But this time you have an additional need. Not only are you looking for input on your product, but you will be looking for a company or two to help you wring the bugs out.

These test sites are invaluable. If you have an especially good relationship with one of the companies that you have used for product evaluation, try them first. It will always be easier to convince someone who knows you to try your product. Remember that very often smaller organizations are more willing to take a chance on an untried product or concept.

Test Sites

The test site serves two very crucial purposes. The first is actual hands on evaluation and real life use of your product. Second, when you go out to market your product and the customer, not wanting to be the first person to buy it, asks who's using the product, you can say "XYZ Corp."

If XYZ Corp. is a competitor, the customer will wonder if XYZ has a competitive advantage through use of the product. If XYZ is not a competitor, the customer will feel reassured that he is not the first.

If your beta site is happy with the product, definitely ask for an endorsement. A positive letter from a customer can go a long way toward breaking down barriers for new prospects.

How do you convince a company to become a test site? There are no definitive answers to this question; I can, though, offer some definite don'ts.

Foremost, don't expect the beta site to purchase the product at retail. A charge for media cost may be appropriate, but be prepared to swallow all of the cost. Once the company makes the commitment to act as a beta site, you can feel fairly comfortable that they

So Help Me!™ A NEW AND POWERFUL TOOL FOR CREATING POP-UP WINDOWS!!

NEW VERSION! *So Help Me!* is a windowing tool that greatly simplifies adding pop-up help screens, light-bar menus, and error messages to your custom software. Create text for the windows in your program, or place it in one or more ASCII files, and then use *So Help Me!*'s Application Program Interface (API) to display them.

FREE BONUS

You'll be even more productive with *Show Me! Version III*, the \$59.95 pop-up file viewing and pasting power tool sent **FREE** with your paid order!

FEATURES

- Up to four scrollable windows displayed simultaneously with full video attribute control
- Control returned to your program after a defined time period; when a predefined key is pressed; or when any key is pressed (the key value is passed back for processing)
- When messages and menus are disk resident, all file I/O is handled including random file positioning by record numbers or by unique index marks to locate specific text
- Available in both memory-resident and linkable library formats making *So Help Me!* compatible with all languages and compilers for PC/MS-DOS 2.0 & later
- Requires as little as 20K; utilizes EMS memory; sample BASIC, C, Pascal and dBASE III+ programs provided

So Help Me! is only \$89.95 (+\$5 S/H).

To Order Call 800-634-3122.

Visa & MasterCard accepted • 90-day guarantee • No royalties
Serengeti Software • P.O. Box 27254 • Austin, Texas 78755-9954

Reader Service Number 27

will continue to use the product after release.

Even then you should consider giving them the product at release. Remember that they have a lot of time and money involved in your product, too.

Conclusion

Getting a successful product to market is full of pitfalls. Successful products are market driven, not seller driven.

What I've attempted to outline here is a method which reduces the pitfalls.

During my years in high tech, I have seen more than one product brought to market because the company knew it would sell, only to find out that the market didn't want what the seller had. On the other hand, some very successful companies have been founded using the basic techniques discussed here.

Mentor Graphics is a notable case in point. The founders of Mentor Graphics used the dog and pony show to define their first product line, and it worked.

The most important thing to remember is to listen to your customer base!

◆ ◆ ◆



PASCAL PROCEDURES

Writing Your Own Window Unit

Don't Buy A Library, Do It Yourself

By John Paul Jones

6245 Columbia Ave.
St. Louis, MO 63139
(314) 645-1596

You can purchase a package of Pascal windowing utilities or you can follow along as John creates a very solid set. Join him as he takes advantage of the new Turbo units to enhance his user interfaces.

Although I indicated last time this column would be about mixed language programming, it's turned out that it will be better to postpone that topic for a while. Instead I'll talk about application windows.

With regards to the text screen, a window is an area to which normal output with WRITE or WRITELN will be restricted. Beginning with version 2 (DOS), Turbo Pascal has had built-in support for screen windows. Try this little program:

```
program windemo;
  uses crt;      { needed for version 4 }
  var i : integer;
      ch : char;
  begin
    window(15,5,65,12);
    for i := 0 to 2000 do
      write(chr((i mod 224)+32));
      {don't allow ctl chars}
    read(ch); { wait for input }
  end.
```

The WINDOW statement in this program defines a screen area with upper left corner at x = 15, y = 5 and lower right corner at x = 65, y = 12. Rather than filling the screen with an incrementing character pattern, only the area defined in the WINDOW statement is used. When a WINDOW statement has been executed, cursor locations are window relative, not screen relative. In the example above, valid x cursor positions range from 1 to 49, and y positions from 1 to 6.

Great, but what's it good for? By itself, not much; but it can be used to develop something useful. There are several commercial productivity modules on the market which provide very sophisticated screen and window manage-

ment facilities. In many cases, there's a lot more in these packages than we need, so let's develop a Turbo UNIT to provide a useful subset of text screen routines. In this unit we have routines for:

- (1) opening a window (with or without border) anywhere on the screen
- (2) save/restore of the main screen (full display)
- (3) one line error messages (nondestructive)
- (4) clearing to end of the current window from the current cursor position

Figure 1 is the Turbo 4 source for the unit. The TYPE window_rec is exported, it defines the basic information we need to manage a screen window. The position of the upper left corner (absolute screen coordinates), width and height are the minimum information needed. We've added some fields (foreground and background colors and some boolean flags) to make our procedures a bit more versatile.

The global variable mgr_ok is also exported; if it is FALSE, the function of the remaining routines is suspect. Within the IMPLEMENTATION section are a few global declarations. TYPE screen represents the underlying structure of the screen memory in text mode. Each character on the physical screen is represented by a character-attribute pair contained in one word's storage. The 80 column X 25 row (full) screen is thus 2000 words.

The function setupscreen serves several purposes. First it calls BIOS to determine the current video mode; if it is not one of the recognized text modes, the function returns FALSE to indicate an error.

Since the different video adapters have their refresh memory located at different addresses within the processor's address space, the routine assigns the proper segment address for the mode detected. Then it allocates a screen-sized block of memory on the heap for savescr and restorescr. Finally the screen is cleared with the default foreground and background colors (white on black). This routine *must* have returned a TRUE result to mgr_ok for savescr and restorescr to work properly.

Savescr stores the current (window relative)

cursor position in the global variables saved_x and saved_y, then copies the entire physical screen contents into dynamic storage. This allows us to open a window "on top" of the current screen, then later restore the overwritten area. This is a bit "brute force" since theoretically only the area overlaid would need to be saved. Turbo 4 handles the move statement fairly efficiently, so there is no significant speed penalty for doing it this way.

Restorescr does pretty much the opposite of what savescr does, but it does not restore the cursor. The assumption is that there has been an intervening window open and use, so only the caller knows where the cursor really belongs.

Cleors is a handy little routine, included here because the critical piece of information it needs, the number of lines in the currently active window, is available. With only slight modification, it could be used for everyday full screen work.

Open_window is the routine used to set up a new screen window. It first sets the foreground and background colors, then if the save flag is TRUE, saves the physical screen. It then calculates the absolute screen coordinates for the lower right corner of the window, and if the border flag is TRUE, frames the area. The position and size values passed in the window_rec parameter become the "active" window, and the border surrounds that area. Finally the window is created and cleared if the clear flag is TRUE. The cursor is left at its default position $x = 1, y = 1$.

The final routine, error, displays a one line, reverse video, flashing error message anywhere on the physical screen. It then waits for either the programmed number of seconds (time) or a keystroke. Before exit, the window described in the window_rec parameter is activated. Normally this would be the caller's window but does not need to be. Figure 2 tests the scrnmgr unit.

No explicit close_window routine is needed, windows are "closed" automatically when another one is opened.

Since this is but a subset of a full function screen window management package, there are a few caveats and limitations. Most important, virtually no error checking is done.

Strange and wonderful things will happen if parameter values place windows or borders outside the physical screen. It's your responsibility to check

for these kinds of errors. Also, it's your responsibility to insure that the fields of your window_rec variables are initialized and updated correctly. You will also need to keep track of which window is active.

Although you can create as many non-overlapping windows as will fit the screen, you can create only a single overlapping window if you plan to restore the data in the underlying screens. (Because I save only one physical screen image.)

You could save more than one screen by simply passing pointers as parameters to savescr and restorescr. You'd use 2000 bytes of memory for each screen, and you'd need to write the routines to manage these images.

With just these simple screen/window management routines, you can do some pretty sophisticated things. I developed these routines for an application program that:

(1) in normal mode, acts as a "dumb terminal."

(2) allows opening a special function window for menu-driven functions.

(3) allows you to drop out of the program into DOS, do what you want, then EXIT back to the terminal mode without losing the screen information.

'Til next time...

Editor's note: John's code and the rest of the code in this issue are available on the issue #42 disk (\$6 postpaid, add \$2 foreign for postage) and the Micro C RBBS (503-382-7643, 24hrs, 300-1200-2400,8, N,1).



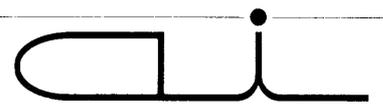
*Finally —
Affordable Intelligence.*

TINY EINSTEIN *The Expert System Shell*

- Create your own expert systems in minutes.
- With pulldown menus and windows
- Context-sensitive online help
- Free example expert systems
- Tutorial
- Interactive full-screen text editor
- DOS access from shell
- Turbo Fast execution
- Cluster, Trace, Explain
- For Diagnosing . . .
Simulating . . .
Predicting . . .
Planning . . .
Classifying . . .
Training . . .
and Monitoring systems.

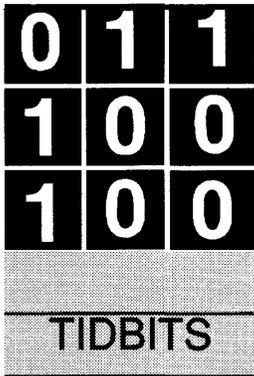
Only \$49.95! (Plus \$5 S/H)

Designed & implemented by
Gary Entsminger & Larry Fogg



ACQUIRED INTELLIGENCE
P.O. BOX 2091 • DAVIS, CA 95617 • (916) 753-4704

Reader Service Number 72



Building Fast AT Clones

And Faster Rabbits

By Gary Entsminger

1912 Haussler Dr.
Davis, CA 95616

When Gary told me he was planning to build something, I had no idea he was planning something this big about something so small. Ah well, he carries it off well.

One late Bend summer night in '86, I assembled my first XT clone (and wrote about it in the December '86, "Fast And Cheap" issue of *Micro C*, #33).

That bare bones clone cost me about \$800 (with 2 floppies, 256K RAM, serial port, parallel port, Hercules compatible graphics card, and monitor). I was tickled that so much power was so cheap (and still am, as a matter of fact).

In the two years or so since, I've learned more ways to use my clone, and more about it. And with only a few well-calculated (fair-priced, I'd say) upgrades (a 20 megabyte Seagate ST225 hard drive), a faster main board (PC Tech X16, 80186 CPU), and a cosmetic addition (an EGA adapter and monitor), it's served me well.

And yet, I'm thinking about moving up a computing notch. Consider my arguments.

First — I depend more on larger programs. I never thought I'd succumb, but I'm facing it: a few of those BIG programs are niftier than a night heron in a sunset.

I'm computing better (even programming better) thanks to some fairly major productions like Paradox, Super Calc, Statgraphics, and PROLOG. I've also gotten into the very addictive habit of suspending one program to run another one. But this kind of pleasure takes memory; 640K hasn't felt like quite enough for several months now.

The West Coast Computer Faire

So I was in the mood; and heading west, I met Dave and the Boys at the West Coast Computer Faire in San Francisco. There, I saw and heard the flair and scream of flying ATs and AT 386s.

A little more speed, I said aloud. Maybe then *Micro C* could go monthly. *Editor's note:*

Subversive talk, Gary.

Or should I get by with the power I have? Maybe add a little memory? Maybe just forget the pleasures of suspending and pseudo-multitasking? Maybe wait another year (or two) for an improved OS/2?

After all, my system has been as faithful as a Kaypro 2. It works as well now as it did in 1986 (gemlike almost, except for a half-crazed power supply that loves to blink).

Also, don't you just hate giving up something that works, even if it is no longer the cat's meow? Maybe this could be the BUT I KNOW IT WORKS ARGUMENT.

I countered quickly — But all my friends' computers are faster than mine. Even Dean Klein (the designer of my XT) drives an AT. Call this THE ARGUMENT FROM DESIGN.

A little more speed, I said aloud. And let's get on with the story.

Are you interested in upgrading your system to an AT (or building one from scratch)? Here's what's out there in the cheap, high-powered computing arena.

Cheap And High-Powered Details

If you're starting from scratch, you'll be assembling a system which consists of —

- (1) Main board
- (2) Drives & Controller
- (3) Video & Controller
- (4) Case & Power
- (5) Serial and Parallel Ports
- (6) Keyboard

Of these, the most subtle differences occur in the main board.

AT clones come in a variety of speeds and packages. They have everything from lowly 6 MHz 80286 CPUs to lordly 16 and 20 MHz 80286 and 80386 CPUs.

Inside them, there's memory — packaged either as common dual inline packages (DIPs) or as single in-line memory modules (SIMMs). The SIMMs take up much less board space than DIPs, and, generally, SIMMs are soldered in place which precludes problems with sockets.

The current crop of 80386-based clones (in-

cluding the Compaq Deskpro) are little more than glorified ATs, so I'll treat them as such for now. But the 80386 CPU has the clear long-term advantage of a 32-bit data bus, and more flexible and efficient multi-tasking.

By comparing the combination of RAM type, RAM speed, CPU speed, and wait states, you can get a pretty good indication of the overall execution speed of a system.

Wait States And RAM

A wait state is an extra clock cycle that the CPU can add during RAM or I/O access. For instance, the CPU says "hey out there, I want the byte you just received from the keyboard." If the device that's holding the byte is slow, then the data won't be available on the data bus right away. So the processor pauses for a clock cycle (a wait state) or two while the slow device gets its data together.

The 8086 bus cycle (for instance, a memory access) takes 4 clock cycles. Adding 1 wait state makes that 5 clock cycles (a 25% performance penalty). Adding 2 wait states cuts performance by 50%.

The 80386 uses only 2 clock cycles to do the same thing. Thus, 1 wait state means a 50% performance cut.

A 0 wait state system runs faster, but a system designed for 0 wait states will probably be more expensive, since you'll pay more for fast RAM and other fast chips. Everything must keep up with the CPU. Or else.

In general, a 16 MHz CPU requires 80 or 100 ns RAM chips, or the system freezes more suddenly than a dewdrop. 12 MHz will need 120 ns RAM chips. 10 MHz will need 150 ns RAM chips.

Editor's note: Gary's right, generally. But, a lot depends on the quality of the bus signals and the RAM. 100 ns RAM has valid data available on the bus about 100 ns after it receives the second half of the address. At 10 MHz, one clock cycle takes 100 ns. Fortunately RAM has a bit over one cycle to get it together and most RAM chips (especially Japanese parts) are faster than they're marked. However, it would be much safer if you used 120 ns parts on a 10 MHz, 0 wait state, system.

Wait states, RAM speed, and a few tricks of the trade will determine whether a system flies or not.

For example, a 286 running at 16 MHz, 0 wait states will (in theory) not run with 100 ns RAM. But it will run (at 0 wait states) if consecutive reads are from alternate banks.

An interleaved memory system organizes RAM into 2 banks — odd-numbered addresses in one bank, even numbered addresses in the other. Since most data reads or writes are sequential, it's unlikely we'll have consecutive memory accesses to the same bank. So the system can maintain 0 wait states. When there are consecutive accesses to the same bank, the system adds a wait state.

The Tandy 4000 (an 80386-based AT), the Wyse 386, and the McTek 286 AT use this interleaved method to get 16 MHz, using 100 ns SIMM DRAM.

Another way around wait states is to use another (more expensive) kind of memory — static RAM (or SRAM).

On the early PCs and XTs, all RAM was just DRAM (dynamic random access memory). On the AT, engineers have designed around DIP DRAM, SIMM DRAM, and SIMM SRAM.

Most 10 MHz ATs at the West Coast Computer Faire went with tried and true, old-fashioned DIP DRAM. But some 12 MHz, and all the 16 MHz, ATs I saw were using SIMM DRAMs. The 386s I saw were using either SIMM DRAMs or SIMM SRAMs.

SRAMs are usually 100 ns parts on 80386 computers. Each megabyte of SRAM consists of 4 SIMMS that mount perpendicular to the board. Each SIMM contains eight 32K by 8-bit chips.

The tradeoffs seem straightforward. You can get static RAM that's faster than the best dynamic, but, bit for bit, it's a lot more expensive.

One other item of note — I've heard of three AT boards that are designed around a passive backplane. The CPU and memory boards sit in expansion slots.

The Wyse 386, the PC Tech X24 and the CCI ST/286/12 use this design. This design permits (at least, in theory) a designer to more easily upgrade a board for faster processors and memory.

In sum, I found these options —

(1) 8 MHz 80286 CPU, 150 ns conventional RAM — at this notch you get IBM AT speed and total compatibility.

(2) 10 MHz 80286 CPU, 0 or 1 wait state, you might get by with 150 ns RAM with 1 wait state.

(3) 12.5 MHz 80286 CPU, 0 or 1 state, 120 ns RAM for 1 wait or 100 ns for 0.

(4) 16 MHz 80286 CPU, 0/1 wait state, using 100 ns SIMM DRAM and interleaved memory.

(5) 16 (and 20) MHz 80386, 0/1 or 0 wait states, depending on RAM. Static

RAM gets 0 wait states; DRAM gets 0/1.

Compatibility

According to the folks I talked with at the WCCF, a 10 MHz AT (switchable to 6 or 8) will probably handle most software and all but old (whatever "old" is) peripherals. Expanded memory cards, intended to run at 6 or 8 MHz, with slow RAM (150 ns or so) might be a problem at even 10 MHz and certainly a problem at 16 MHz.

Very early XT video (circa 84 and before) cards have had problems at anything over 6 MHz. But any fairly recent video system was probably upgraded (before you bought it) to support the faster ATs.

So, a good bet is to go for a new mainboard which allows you to upgrade to as much memory as you think you'll need ON BOARD.

And if the memory on board is EMS compatible (something else to check for!), you whip two villains (incompatibility and/or slowing down to accommodate slow expansion cards) with one check.

Drives And Controllers

The floppy drive story has changed a bit. You now have four good possibilities (if you're running DOS 3.3 or better) —

5 1/4" 360K
5 1/4" 1.2M
3 1/2" 720K
3 1/2" 1.44M

For the record, DOS 3.0 supports the first two of the above, DOS 3.2 supports the first three, and DOS 3.3 supports all four.

If you want two floppy drives, and want to be able to read everything, go for a 1.2 meg 5 1/4" and a 1.44 meg 3 1/2". The 1.2 will read 360. The 1.44 will read 720. And the extra storage is worth it.

Editor's note: Reading is one thing. Writing is a whole different kettle of vipers. Sure, those handy dandy narrow-headed little data munchers will write too, but their tracks are half as wide as their less dense little brothers. See Techtips for more info.

5 1/4" disks are cheap enough to remain popular indefinitely, but the 3 1/2" disks are a nice size and capacity.

These two and either a Seagate or Miniscribe 20 megabyte (28 ms) hard drive make a not-too-expensive and FAST system. Or add RLL and pick up

another 10 Mbytes (see Dave's discussion of RLL).

Coprocessors

If you're crunching numbers, remember that faster CPUs need faster coprocessors. The 80287 runs at roughly 2/3 the speed of the 80286. (The 8087 runs at the same speed as the 8088.)

So an 80286 running at 8 MHz can use the 80287-6 (6 MHz).

An 80286 running at 10 or 12 MHz will want an 80287-8.

And an 80286 running at 16 MHz will need an 80287-10, at least.

Wrapping Up

So let's wrap up this ramble with a few prices and suggestions.

First — if you're going with an 8 or 10 MHz mainboard (or system), piecing a system together part by part (mainboard here, drives there, video over yonder, etc.) is probably safe. But it won't necessarily save you time or money.

Editor's note: Some of those new Jr-sized AT cards can simply replace your XT motherboard. Everything else (including the hard and floppy controllers) simply plugs right in and runs. (Though you may need an AT-specific keyboard.)

Most clone companies have all the clone parts you need (at fairly consistent prices), and if you buy everything from one, you'll probably —

(1) save money;

(2) save work (they'll put the system together for you AND MAKE SURE IT WORKS);

(3) and have someone to tell your troubles to over the long haul.

If you're in the mood to upgrade to an AT, there are a few good deals.

A 10 MHz 80286 with 1 megabyte of conventional RAM could go as low as \$600 (under \$400 w/o RAM).

A very fast 16 MHz 80286 AT board with 1 meg of SIMM DRAM goes for about \$1,000.

A good price for a 386 board (alone) with two megs of SRAM is about \$1,600. (I didn't see anyone selling a 386 with 1 meg for under \$1,500.)

The best price I found on a 10 MHz, 0 wait state system, with 640K RAM (dual in-line), 20 meg hard disk, 1.2 meg floppy, Hercules compatible adapter and monitor, serial and parallel ports (on board in this case), and keyboard was \$1,399.

If you really cut corners (get two 360K floppies, skip the hard drive, get a

CGA, and less RAM), you could do it for well under \$1,000.

The lowest advertised complete 80386 system I found was \$2,675 (16 MHz CPU, 1 megabyte DRAM, 2 floppies, 1 40 megabyte hard disk, monochrome monitor and graphics, serial and parallel ports).

No single company at the West Coast Computer Faire had the best prices on all equipment. But many of the best system and board prices I found were at McTek (the Sky High of my December '86 article) in Berkeley, California. MicroSphere had the lowest priced 80386 kit.

The "name" 80286 and 80386 systems (PC Limited, Compaq, Wyse, CCST, Wells American, etc.) in 10, 12, or 16 MHz versions are considerably more expensive — starting at \$2,000 for 80286 systems and \$4,000 for 80386 systems, and going higher quickly. The Compaq Deskpro 386 tops the list at about \$6,500.

There are a few published reports of compatibility problems with the faster systems (particularly with 80386 software), but the 10, 12, and 16 MHz 286s I saw seemed to run everything.

The Rabbit

So what did I buy? I bought a Rabbit. No, not the cute furry kind, or the funny German car, but the quick, sleeker than a Kaypro 2 portable lunch box variety.

Despite the arguments, I couldn't give up my tried and true X16 desktop (I'll wait until it gives up on me), but I could rationalize a new portable system.

The key was a new supertwisted electro-luminescent back-lit LCD screen, case, and power supply called (I mean it) the Rabbit. The Rabbit has a 640 by 400 pixel screen, and is CGA compatible (and Hercules workable). It's the next best thing to a Compaq portable screen, which makes it (to my eyes) the next best LCD screen on the market. And it comes at a much better price.

Buy a case (at \$799) and build up your own system, or buy a fully equipped and tested Rabbit —

- case with a 640 by 400 LCD & power supply,
- 1.44 megabyte 3 1/2" drive,
- 20 megabyte Seagate hard disk (28 ms),
- and a mainboard (10 MHz, 0 wait state 80286) with 640K RAM on board,

for \$1,799.

It's a few hundred more than the cheapest desktops, but a few thousand less than the cheapest Compaqs. (But note: the Rabbit, like the Compaq, isn't a laptop, it's a portable; you have to plug it in. But you can carry it around. It weighs 19 pounds or so with a 20 megabyte hard drive, and it has a handle.)

The Rabbit case wasn't made to handle 5 1/4" floppies, so if you need 5 1/4" capability, you'll have to put it outside (or buy a Compaq Portable 3).

I couldn't resist hot-rodding (been talkin' with Laine Stump too long I guess), so I bought the Rabbit case, drive, etc., and then dropped in a 16 MHz, 0/1 wait state (depending on how the interleaved memory scheme works) mainboard with 1 megabyte of EMS compatible, (SIMM) DRAM on board, expandable to 4 megabytes.

It looks like it's going to be a honey of a system, and it's not the least bit furry.

Bye

In short, despite high RAM prices, you can buy a lot more computing power for the buck now than ever before.

Back in December '86, when I wrote about building a PC, it cost \$800 for a minimally equipped XT, and at least \$2,000 for a minimally equipped AT (256K, 2 360K floppies). And \$2,550 for an AT with a 20 megabyte Seagate.

This summer, you can cut lots of corners and build a minimally equipped AT for less than \$1,000, and have at least five times the power of an XT.

Really, it's not a bad time to build a clone.

References

Entsminger, Gary. "Build A PC Clone For Less Than \$800." December 1986. *Micro Cornucopia* #27. p 52-55.

Klein, Dean. "Designing With The 80386." February 1987. *Micro Cornucopia*, #34. p 6-10.

Malloy, Rich, G. Vose, G. Stewart. "The Tandy Anniversary Product Explosion." *BYTE*. October 1987. p 100.

McNierney, Ed. "The WYSEpc." *BYTE*. April 1988. p 211.

Poor, Alfred. "High-Speed ATs: Life In The Fast Lane." *PC Magazine*. April 14, 1987. p 119.

Van Name, Mark. "The PC's Limited 386." *BYTE*. December 1987. p 141.





Way Beyond Desktop

Or: The Curse Of The Automated Heads

By Irv Mullins

Rt. 1, Box 312C
Talking Rock, GA 30175
(404) 635-5750

We get a lot of correspondence here at Micro C, letters from a pretty varied group of people. But the most varied correspondent of all has to be P. Leeward. Here's his latest.

Dear Editor: All of us here look forward to readin the "On Yer Own" collum in *Micro Cornucopia*. Me and Luther and LuBell have recently been inspired by it to become independent ontrayprenures.

Our first fling at a business of our own (software sales) was a flop — remind me to tell you about that sometime — but our current endeavver shows a lot of promise.

It all started when I went out to cousin Luther's to see if th' latest *Micro C* had arrived. Cousin LuBell was at the kitchen table readin one of those grocery store magazines when I walked in. She told me that the *Cornucopia* had just been delivered, and Luther, who had been waiting anxusly for some time, grabbed th' *Cornucopia* and lit out for th' readin' room. (We all wish you'd go monthly.)

Well, I was nearly excited enough to join Luther in th' readin' room, but I remembered that Luther's library only has one seat, and presumably he was usin' that. Beside, the atmosphere in there tends to git a little "close," so I elected to stay and talk to Miz LuBell.

"Look at this," she said, pointin' to a big, black headline on th' tabloid. "GHOST OF ELVIS HAUNTS HONOLULU IN UFO!" it said. "Who could believe that kinda garbage?" asked LuBell.

I allowed as to how it had to be true, else they couldn't print it in th' paper. Anyhow, lots of people must believe it, cause they sell milliuns of those magazines ever week for fifty or seventy-five cents apiece, I noted. Well, that display of logic really got LuBell to thinkin.

Just then Luther, flushed, you might say, with exitment, came into the kitchen wavin the *Cornucopia* (it was all about Desktop Publishin') and tellin us how we could use those computers out in the barn to do our own typesettin and such. "All we'd have to buy," Luther said,

"is a lazier printer." I would have thought the one we had was lazy enough, but that's what the artickle said.

"Hummm...," says LuBell. "Maybe we could start our own checkout-line newspaper." She took out a pencil. "Might sell milliuns a week. Lessee," she said, figurin' furiously, "at fifty cents apiece, that's..."

Even befor she finished, Luther and me caught on to the fact that that would add up to a signficant sum, so we decided to start our own shopping cart weekly. We made LuBell Editor-n'-Chief because of her weighty journalistic experience: LuBell anchored the six o'clock Nigh-Witless News on the local TV station for a while (until the tube in the transmitter burnt out.) Beside, the whole thing was her idea.

At first, I was goin to hire Caleb the undertaker to help. He's famous for his layout work. He declined, sayin he already had his hands full at the moment, and at that point I suddenly lost interest in persuin' the subjek any further. We bought one of them PageMucker programs insted.

All that was left was to write th' news stories. I didn't know where we were goin to git all th' reporters and corrystspondents we'd need, but LuBell had already thought of that. She hauled an old TellyType in from th' barn, hooked it up to a cereal port on one of th' PC's, and wrote a short program in Turbo Pascal which, she explained, would generate random headlines. Then we set back to let th' news roll in.

Boy, th' place really looked, sounded and smelt like a workin' newspaper with that TellyType chunkin' away. All we had to do was pick an excitin' headline and fill in th' details. (Just like all the big-time publycations do, accordin' to LuBell.)

Well, since then *The Talking Rock Tattler* has been sellin like hotcakes. We'd like to share our good fortune with your readers, so we are makin Miz LuBell's headline-maker program available free of charge to *Micro C* readers.

The program requires a word data-base to run. An ASCII file which contains the entire

386 386 386 386 386 386 386 386 386 386 386 386

CLONE SYSTEM

386 RUNNING AT 20 MEG Hz WITH 0 WAIT
MDA,CGA,EGA,VGA ON MOTHER BOARD
2 SERIAL AND 2 PARALLEL PORTS
ONE MEG RAM ON BOARD
WA2 HARD FLOPPY CONTROLLER
CASE, KEYBOARD AND POWER SUPPLY
EGA MONITOR
40 MEG HARD DRIVE AND ONE 1.2 FLOPPY
ONE YEAR WARRANTY ON SYSTEM

*****\$2995.00

286 286 286 286 286 286 286 286 286 286 286 286

AT TURBO SYSTEM

AT COMPATIBLE MOTHER BOARD WITH BIOS
8 MEG AND 12 MEG SWITCHABLE SPEED
512K RAM INSTALLED UP TO 1024 ON BOARD
WA2 HARD DISK/FLOPPY DISK CONTROLLER
MONOCHROME GRAPHICS VIDEO WITH PRINTER
40 MEG HARD DRIVE AND ONE 1.2 MEG FDD
200 WATT POWER SUPPLY AT CASE
AMBER MONITOR KEYBOARD
ONE YEAR WARRANTY ON SYSTEM

*****\$1695.00

XT CLONE SYSTEMS

PLEASE CALL FOR CURRENT PRICE

HARD DRIVES FOR XT AND AT

ST-225 KIT FOR XT (20 MEG) \$ 259.00
ST-238 KIT FOR XT (RLL 30 MEG) \$ 279.00
ST-251 FOR AT (40 MEG) \$ 359.00

MONITORS

Color Monitor RGB (CGA) \$ 255.00
Color Monitor RGB (EGA) \$ 355.00
Monochrome TTL (Green) \$ 95.00
Monochrome TTL (Amber) \$ 105.00
EGA Color Video Card \$ 129.00

CITIZEN PRINTERS

MODEL 120D 120 CPS 9" \$ 179.00
MODEL 180D 180 CPS 9" \$ 199.00
MODEL MSP-15E 160 CPS 15" \$ 359.00
MODEL MSP-40 240 CPS 9" \$ 319.00
MODEL MSP-45 240 CPS 15" \$ 439.00
MODEL MSP-50 300 CPS 9" \$ 419.00
MODEL MSP-55 300 CPS 15" \$ 499.00

CASCADE ELECTRONICS, INC.

ROUTE 1 BOX 8
RANDOLPH, MN 55065
507-645-7997

Please ADD Shipping on all Orders
COD Add \$3.00 Credit Cards ADD 5%
Limited to Stock on Hand Subject to change

200 word vocabulary of all the major tabloids is available from us for the small fee of \$199.95 plus shipping and handling and parts and labor plus applicabul Federal, State and local taxes and refreshmints for th' shippin' clerk.

Now, \$199.95 may seem like a lot for a couple hunnerd words, but we had to read a whole year's worth of grocery store magazines to gather them words, and we need th' money to pay for psycheatric treatments.

Got to go now, th' TellyType just spit out th' shockin' news: "TOKYO TODDLER MAKES SUSHI OUT OF SIT-TER!" a story which really piques this reporter's curiosity.

Sincerely,
Your Colleague & Fellow Publishing Mogul,
P. Leeward Sailors

p.s. For all you folks with enquirin' minds who want to see next month's hot news headlines in advance, here's a sample run from the *Tattler Tellytype*. (Don't laugh, I just saved you fifty cents.) These have been edited some, so they'd make sense, but we don't usually bother (pressure of deadlines, you know).

PSYCHIC CARROT REVEALS
SECRET OF ETERNAL YOUTH

BERMUDA TRIANGLE
MYSTERIOUSLY VANISHES

N.J. GIRL BECOMES GREAT-
GRANDMA AT SIX

PROOF: MUTANT SPACE ALIENS
CAUSE ACNE!

CHIMPS SECRETLY CONTROL
WORLD POWERS

PROOF! OPRAH WINFREY,
PRINCESS DI, SIAMESE TWINS

BRAIN TRANSPLANT CAUSED
PRESIDENTIAL FORGETFULNESS

CANNIBAL PROWLs SOVIET
SPACE STATION

MAIMED FOR LIFE, DOUBLE-
AMPUTEE FROG SUES
RESTAURANT

◆ ◆ ◆

instructions generated are reduced down to two MOV instructions.

The decision of which language to use for the development of a project is entirely dependent on a combination of design goals and some economic factors.

- Will the system be ported to more than one processor family?
- Are speed of execution and compact size of utmost importance?
- Can the task be handled by a high-level language?
- Does your staff currently have programmers skilled in both assembly and high-level languages?
- Do you want to take advantage of the dozens of commercially available libraries for screen handling, database management, etc.?

Please print a more balanced view of issues such as these in the future.

Tim Berens
3210 Ackerman Blvd.
Dayton, OH 45429

Another C Defense

I am writing in response to Eric Isaacson's article, "C vs. Assembly Language" in your March-April issue. Although I agree with him that most of what we hear about C is nothing more than marketing hype, I must disagree strongly with many of the statements he makes.

He begins with a short history of the Apple Lisa and states "[Apple] hired hundreds of programmers who dutifully wrote in Pascal, the darling high-level language of the day." I don't know where he got this figure, but let's assume for the moment that it's accurate. Such a large number of programmers is almost guaranteed to create havoc no matter what language they use.

Fred Brooks has written an entire book, *The Mythical Man-Month*, on this very subject. If the idea of hundreds of programmers writing in Pascal frightens you, then the sight of hundreds of people coding in assembler would be a vision from hell.

Mr. Isaacson goes on to describe two assemblers he has written, one while he was at Intel and another he wrote on his own. The Intel assembler was written in PL/M by a team of three people and took eleven months to complete (33

man-months). The other assembler was written by himself and took six months to complete.

Several questions need to be answered before we can make a meaningful comparison. Had these programmers ever used PL/M before? How much of the eleven months was spent learning the compiler, writing memos, attending staff meetings, and doing all the other things employees of a large corporation do? One month? Two? How much time was spent developing the requirements definition? Another month? How much design time did Mr. Isaacson save on his second project by reusing ideas from the first project? This 6-to-33 ratio he quotes doesn't tell the whole story.

When it comes to debugging, Mr. Isaacson seems to be a little behind the times. I know of at least three C interpreters (Safe C, Run-C, C-terp) that run under MS-DOS and Xenix, not to mention symbolic debuggers like CodeView. If you are a software developer in today's market, you owe it to yourself to stay informed on the latest tools and methods. You can bet your competitors do!

It is evident that Mr. Isaacson doesn't really understand why a C compiler generates code the way it does. The design of the language simply prohibits certain kinds of code from being produced. We need to remember two facts about C: All functions are potentially recursive and functions may be called with a variable number of parameters.

Recursion demands that all parameters and variables be kept on the stack so they won't be destroyed by a subsequent activation of the same function. And having varying numbers of parameters means the "RET n" instruction can't be used since the compiler has no way of knowing what the values of "n" should be. This is also why the calling function is responsible for removing the parameters from the stack.

One point deserves particular attention. Mr. Isaacson tells us his assembler contains an error-handling mechanism similar to Ada's exception handling and claims, "[it] could not have been done in any of the high-level languages available at the time." This is simply not true.

Any conceivable control structure can be implemented through the proper

combination of sequential execution, branching, and looping. This was formally proven by Bohm and Jacopini over 20 years ago. In fact, all the Ada compilers I am familiar with (Alsys, Meridian, R&R) are themselves written in Ada. In each case, the compiler writers have found a way to implement exception handlers in a high-level language (I think we all agree that Ada is a high-level language).

Finally, Mr. Isaacson concludes that, "[assembly language programs] match high-level programs in reliability, readability, and speed of development." In my view, he has not presented any evidence to support these claims.

Marc A. Poulin
XonTech, Inc.
6862 Hayvenhurst Ave.
Van Nuys, CA 91406

Seagate Repair

Well boys and girls, there's good news and bad news from the Seagate folks. First the good news: When, not if, your Seagate budget priced hard drive (ST-225, 238, or 251) craps out and the clone shop you bought it from has gone chapter 13, not to worry! Our buddies in Delray Beach, Florida, have implemented their long awaited "orphan policy."

Here's the way it works: Call 1-305-278-5899, explain the situation, and get an "RMA" number. Then ship the dead drive, along with 35 skinny frogs flat rate exchange fee to:

Seagate Technologies
Congress Park South
430 Commerce Dr.
Delray Beach, FL 33445

Seagate will ship back a rebuilt "more better" drive.

Now the bad news. Don't be in a hurry! When I got my RMA number I was assured that the turn around time was less than ten days. I sprung for Fed Ex shipping to get the broken puppy down there ASAP. Today when I called (ten days after Seagate had received my drive) I was told that it would be another week before "paper work made it upstairs."

When I asked what was the cause of the delay I was told, "There are a lot of drives being repaired." All things con-

Letters

sidered, it's still a good deal. Thirty-five bucks is "more better" than collecting broken puppies.

Mike Rutkoski
13523 Westwind Dr.
Silver Spring, MD 20904

Editor's note: We gave Seagate a call and, yes indeed, they are offering the flat rate exchange on the drives mentioned above. However, this only applies to drives still within the one year warranty. Use the date stamped on the drive itself, not the date of purchase.

For out-of-warranty drives, Seagate offers the following flat rates for rebuilds:

ST-225 \$170
ST-238 \$180
ST-251 \$290

Foreign Subscriptions

I subscribe to, and enjoy very much, *Micro C*. However, the cost of a foreign subscription really hurts, mainly due to the air mail postage. I subscribe to *PC*

Magazine and it costs me only about \$2.50 an issue. The reason *PC Mag* is so cheap (apart from its mass production) is because it's sent surface mail.

I know it may get over here a bit (a lot?) later than air mail but it sure saves a lot of money. I don't really get *Micro C* for its topical content, so does it matter if it gets to England in one week or one month?

I certainly wouldn't mind, especially if it brought the subscription price down to near the US amount. Is there any special reason you don't offer this option?

Phil Coull
40 Wear Bay Crescent
Folkestone
Kent, CT196BA
England

Editor's note: I have some good news, Phil. (And some bad news.) First, we're offering surface now. In fact, we've sent the last two issues surface. The first one went surface by accident, the second one went

surface after the post office sent us the new rates. They raised the foreign airmail rates from around \$4 per copy to as high as \$5.50.

So, we're increasing the subscription rates for foreign airmail, and we're keeping the old rates for surface. We considered reducing the surface rate a bit, but when we checked the new surface rates and added in postage for renewal notices and returned magazines (we pay to get them back if they can't deliver them), it didn't work out.

In Praise Of FORTRAN

I was amused to find out that the final score in the contest between Scientist and FORTRAN indicated that the scientist was the loser. Or so Mr. Eckel claims in *Micro C* Issue #40.

FORTRAN is *not*, as he indicated, around due to tradition and misconstrued convenience. The large libraries written in FORTRAN are certainly an incentive for using it in the scientific setting. But other important factors include speed of compilation, execution speed, simplicity, and sometimes, availability — it's the only decent compiler in some environments.

FORTRAN certainly has its limitations. So does virtually every other programming language (yes, including assembly). As a result, I spend about equal amounts of time with FORTRAN, Modula-2, and C. I use FORTRAN for number mangling, C to treat streams of characters, and Modula for data basing. The point is, use the best, and best suited tools available.

A colleague recently voiced a wish for Turbo FORTRAN. The idea of a simple, fast FORTRAN in an interactive environment is a good one. I think a letter writing campaign is in order here. Let Mr. Kahn know that there's a need for that unique Borland touch in the FORTRAN market.

This should not be construed as a criticism of C++; In fact, I think it is an exciting and positive development. I seriously doubt that it will make much of a dent in the laboratory environment, though, until it's available on standard lab computers like Perkin-Elmers and DEC Rainbows (yeah, I saw a whole lab full of Rainbows the other day).

James W. Albert
1500 W. William Cannon Dr. #183
Austin, TX 78745

◆ ◆ ◆

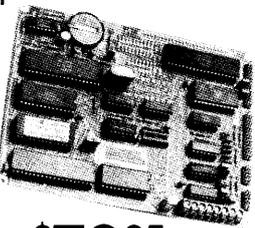
**THE NEW 65/9028 VT
ANSI VIDEO TERMINAL BOARD!**
★ FROM LINGER ENTERPRISES ★

A second generation, low cost, high performance, mini sized, single board for making your own RS232 Video Terminal. This highly versatile board can be used as a stand alone video terminal, or without a keyboard, as a video console. VT100, VT52 Compatible.

FEATURES:

- ★ Uses the new CRT9128 Video Controller driven by a 6502A CPU
- ★ On-Screen Non-Volatile Configuration
- ★ 10 Terminal Modes: ANSI, H19, ADM-5, WYSE 50, TVI-920, KT-7, HAZ-1500, ADDS 60, QUME-101, and Datapoint 8200
- ★ Supports IBM PC/XT, and Parallel ASCII Keyboards
- ★ Supports standard 15.75 kHz (Horiz.)
- ★ Composite or Split Video (50/60 Hz)
- ★ 25 X 80 Format with Non-Scrolling User Row
- ★ Jump or Smooth Scroll
- ★ RS-232 at 16 Baud Rates from 50 to 19,200
- ★ On Board Printer Port
- ★ Wide and Thin Line Graphics
- ★ Normal and Reverse Screen Attributes
- ★ Cumulative Character Attributes: De-Inten, Reverse, Underline and Blank
- ★ 10 Programmable Function Keys and Answerback message
- ★ 5 X 8 Character Matrix or 7 X 9 for IBM Monitors
- ★ Mini Size: 6.5 X 5 Inches
- ★ Low Power: 5VDC @ .7A, ± 12VDC @ 20mA.

MICRO SIZE!



\$79⁹⁵

FULL KIT
w/100 Page Manual
ADD \$40 FOR A&T

OPTIONAL EPROM FOR
PC/XT STYLE SERIAL
KEYBOARD: \$15

SOURCE DISKETTE:
PC/XT FORMAT
5 1/4 IN. \$15

Digital Research Computers
P.O. BOX 381450 • DUNCANVILLE, TX 75138 • (214) 225-2309

Call or write for a free catalog on Z-80 or 6809 Single Board Computers, SS-50 Boards, and other S-100 products.

TERMS: Add \$3.00 postage. We pay balance. Orders under \$15 add 75¢ handling. No C.O.D. We accept Visa and MasterCard. Texas Res. add 6-1/4% Tax. Foreign orders (except Canada) add 20% P & H. Orders over \$50 add 85¢ for insurance.

Reader Service Number 32

Housing shouldn't be a problem if you'll call early for reservations. Check the SOG registration info in this issue for a list of our recommendations. (Bend has some 300 new motel rooms since last summer.)

You can tent or trailer camp at Tumalo State Park (it has solar heated showers). They don't accept reservations so your best bet is to arrive around noon and get a spot as someone leaves. By 3 p.m. it's usually full. The park is in the community of Tumalo, just northwest of Bend. Or you can sleep in your car, trailer, or motor home in the COCC campus parking lot.

If you're flying in (commercial), you can come as far as Redmond, Oregon, and catch a shuttle into Bend (or rent a car at the airport). Or you can save money by flying into Portland and renting a car there. The drive from Portland is about 3 hours. (The scenery is great.)

Flying your own plane? You can join the SOG fly-in at the Bend Airport. Call Bill Davidson at PC-Tech, (612) 345-4555, for information on the fly-in. The Bend FBO will have a car for participants.

Are RAM Prices On The Up And Up?

If you've purchased a system in the recent past, you've probably been surprised by the price for the RAM. During the third week in April, for instance, the price of a 150ns 256K dynamic RAM rose from \$6.50 per piece to \$9.50. (Twelve months ago they were \$2.) That means that memory is approaching \$360 for a meg, up a bit from \$72 per meg a year ago.

What's causing this? I don't know for sure, but I'll let you in on the facts I've discovered and the rumors I've heard (only the best rumors, of course).

Fact: Last spring the U.S. put import restrictions on such luxuries as 256K dynamic RAMs, laptop computers, and power tools (limits on quantities and a 100% duty). As far as I can recall, the restrictions on RAM were pushed through Congress by Micron Technology, the only U.S. manufacturer of the little darlings. (Intel is buying theirs from Korea, TI is getting them from Hyundai.)

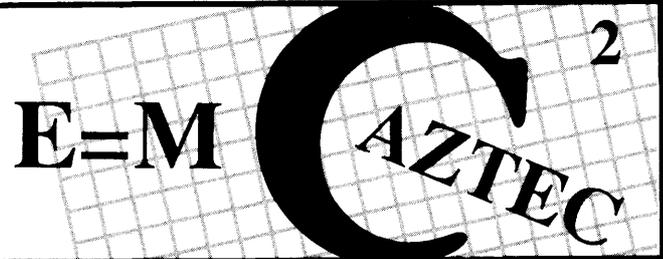
Fact: The Japanese have been changing 256K production lines into 1 meg production lines. The 1 meg parts, which have been designed into the new 386 systems, are selling for \$35 to \$40 each.

Fact: A majority of the chips currently produced are already spoken for. IBM, Compaq, Apple... have contracted for 80% of the current production. (Contract prices reflect the going rate at the time the contract was written so they are generally much lower. Parts houses with long term contracts are, no doubt, making out like bandits.)

Fact: Computer sales are very strong. Apple sales are up 50%, Compaq up the same percentage. The only ones doing lower volume than last year are IBM and Kaypro (as far as I know). When sales are stronger than expected, companies have to purchase the extra chips on the open market. When sales are weaker than expected, they sell their extra parts, usually for less than they paid.

Fact: Most of the new software needs lots of memory. MS-DOS 3.3 requires 256K just to load the operating system. OS/2 requires 400 to 600K. So, not only are companies selling more computers, but they're putting more memory in them. Plus, there's a strong market for extended memory boards and EGA or VGA video cards.

Fact: Even for those willing to pay current prices, supplies



Genius Begins With A Great Idea...

**FREE
2 DAY
DELIVERY**

**Aztec C86 4.1
New PC/MS-DOS
CP/M-86 • ROM**

Aztec ROM Systems
6502/65C02 • 8080/Z80
8086/80x86 • 680x0

Superior performance, a powerful new array of features and utilities, and pricing that is unmatched make the new Aztec C86 the first choice of serious software developers.

An IBM or Macintosh is not only a less expensive way to develop ROM code, it's better. Targets include the 6502/65C02, 8080/Z80, 8086/80x86, and 680x0.

Aztec C has an excellent reputation for producing compact high performance code. Our systems for under \$1,000 outperform systems priced at over \$10,000.

Aztec C86-p.....\$199

- optimized C with near, far, huge, small, and large memory - Inline assembler - Inline 8087/80287 - ANSI support - Fast Float (32 bit) - optimization options • Manx Aztec 8086/80x86 macro assembler
- Aztec overlay linker (large/small model) • source level debugger • object librarian • 3.x file sharing & locking • comprehensive libraries of UNIX, DOS, Screen, Graphics, and special run time routines.

Initial Host Plus Target...\$ 750
Additional Targets.....\$ 500
ROM Support Package....\$ 500

Vax, Sun, PDP-11 ROM HOSTS

Call for information on Vax, PDP-11, Sun and other host environments.

Cross Development

Most Aztec C systems are available as cross development systems. Hosts include: PC/MS-DOS, Macintosh, CP/M, Vax, PDP-11, Sun, and others. Call for information and pricing.

Aztec C86-d.....\$299

- includes all of Aztec C86-p • Unix utilities make, diff, grep • vi editor • 6+ memory models • Profiler.

Aztec C86-c.....\$499

- includes all of Aztec C86-d • Source for library routines • ROM Support • CP/M-86 support • One year of updates.

Third Party Software

A large array of support software is available for Aztec C86. Essential Graphics • C Essentials • C Utility Library • Greenleaf Com. • Greenleaf General • Halo • Panel • PC-lint • PforCe • Pre-C • Windows for C • Windows for Data • C terp • db_Vista • Phact • Plink86Plus • C-tree.

CP/M • 8080/Z80 ROM

C compiler, 8080/Z80 assembler, linker, librarian, UNIX libraries, and specialized utilities.

Aztec C II-c CP/M & ROM....\$349

Aztec C II-d CP/M.....\$199

How To Become A User

To become an Aztec C user call 800-221-0440. From NJ or international locations call 201-542-2121. Telex: 4995812 or FAX: 201-542-8386. C.O.D., VISA, Master Card, American Express, wire (domestic and international), and terms are available. One and two day delivery available for all domestic and most international destinations.

Aztec Systems bought directly from Manx have a 30 day satisfaction guarantee. Most systems are upgradable by paying the difference in price plus \$10. Site licenses, OEM, educational, and multiple copy discounts are available.

C' Prime\$75

C' Prime

**PC/MS-DOS • Macintosh
Apple II • TRS-80 • CP/M**

These C development systems are unbeatable for the price. They are earlier versions of Aztec C that originally sold for as much as \$500. Each system includes C compiler, assembler, linker, librarian, UNIX routines, and more.. Special discounts are available for use as course material.



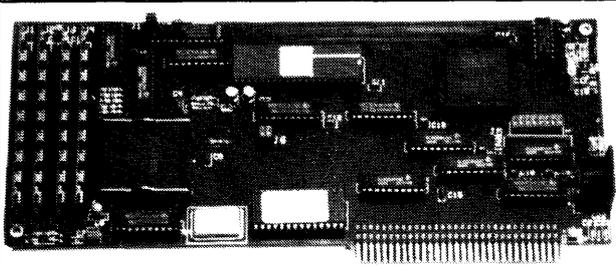
To order or for information call today.
1-800-221-0440

Manx Software Systems
One Industrial Way
Eatontown, NJ 07724

In NJ or international call (201) 542-2121
TELEX 4995812

Reader Service Number 17

A Reliable PC/XT Compatible For The Corner Stone of Your Products



Announcing The SLY40-XT

The SLY40-XT is a small (4-1/4" by 9-1/4"), four layer card featuring all of the PC/XT mother board functions. The board simply plugs into a passive back plane or SLICER'S 10 slot bus board.

- High Integration — Composed of just 17 Low Power CMOS ICS
- NEC's 8 MHZ V40
- One Megabyte of Zero Wait State RAM
- 8087 Co-Processor Socket
- Standard Keyboard Connector
- Slicer's Own Bios, Source Code Included
- Ideal For Tough Industrial, OEM and Portable Applications
- American Made and Fully Supported by Slicer

* TECHNICAL CONSULTING * CUSTOM ENGINEERING

Having a wealth of knowledge in hardware and software development, **SLICER** has been a leading manufacturer of industrial micro computers and numerical controls since 1977. If you require either **TECHNICAL CONSULTING OR CUSTOM ENGINEERING**, give us a call; it would be a pleasure serving you.

MasterCard, Visa, Check, Money Order, or C.O.D.
Allow four weeks for delivery.
Prices subject to change without notice.
NOTE NEW ADDRESS & PHONE NO.

Slicer Computers Inc.
3450 Snelling Ave. So.
Minneapolis, MN 55406
612/724-2710
Telex 501357
SLICER UD

PC and XT Are Trademarks of International Business Machines

Reader Service Number 19

aren't unlimited.

Fact: Suppliers are down to desoldering parts from old boards. You can tell they're recycled by looking at the legs. They usually get shortened slightly during removal. Also, you'll probably find several different product dates within a single tube.

Fact: It takes 6 to 9 months for U.S. manufacturers to get back into RAM production. Assuming that Intel and TI were interested in building their own again, it would be summer before their parts started showing up.

Rumor: Japanese have gotten together to withhold parts from the U.S. market to teach us a lesson. That's not too reasonable, considering how much trouble OPEC has had keeping their members from overproducing. Plus, the Japanese are getting better money per chip for their 1 meg devices.

Rumor: IBM is playing the Hunt Brothers game with the marketplace by buying up all the uncontracted RAM they can find. (I understand that this rumor came from the *Wall Street Journal*.) The increase in prices and decreased availability could hurt small competitors who have to depend on the spot market.

For instance, Kaypro Corp. says it's having trouble filling orders because of the RAM shortage. That's curious if their order rate is significantly down from last year. Rumor also has it that Kaypro had a surprise layoff on April 11. About 100 (out of 350) employees were surprised.

You Aren't Reading This

Like a lot of you, I write hard. Paper doesn't work for me because it can't stand up to blood. I take an axe to my weaker ideas (last issue, the editorial shrank from 21K to 12K in half an hour). After the axe, I take a chisel to what's left. Thus the mangled mess.

This issue, for example, I'm axing this paragraph and the one above it so you won't have to suffer through this kind of drivel.

Tech Calls

I had just bluffed my way through a technical problem when the caller blurted out:

"I wish I were you."

Boy! That will unhinge even the coolest pretender.

"Spending your life trying out and writing about new products."

Hey, this is the tenth tech call of the morning. My stomach, after feeling slighted by breakfast, is already mumbling about lunch.

I was planning to spend the morning writing an introduction for a new catalog, finishing the poster for the Computer Faire, meeting with an unemployed high schooler who wants to become an employed high schooler, photographing clone guts for a cover (we're fancy now, remember), and... (my list is here somewhere).

"Being able to call anyone in the industry to get the scoop."

Who writes this guy's material?

"And you're doing it in Bend, Oregon."

How quaint.

"Anyway, good to talk to you. This'll help me finish up this project so the little lady and I can take off for three weeks in..."

Three weeks? In three weeks he'll have forgotten every-

thing I've told him and he'll have to call back. (And if he mentions the vacation...)

Actually, not all mornings are like this. Some aren't nearly this interesting.

Writing Code

I used to write a lot of code. I wrote little databases, utilities, games, and other reinventions. Then night after night I'd nurture them. (Nobody could add features like I could add features.) With the loving care, they'd grow, sprouting new displays, new functions, new sounds...

Eventually, each got too large, or began to accumulate too many small crawly things, or a root would become entangled with something else. Then, one by one, I'd shove them onto floppy and let them mold in my bottom drawer.

I wrote the code as therapy. Now I don't have time. Even with the new debuggers, it takes days to write really therapeutic code. Now all I write is this. (Editorial debugger anyone?)

The Computer Journal For The Self Reliant?

I've been looking for a better description of *Micro C*. About the best I can say about "The Micro Technical Journal" is it hasn't offended anyone.

I've nosed around for something better (slightly offensive) by asking some of our regular writers what they thought.

Bruce Eckel feels our readers are cheap. Of course he doesn't really mean that because that's more than slightly offensive. (And he's a reader, too.) What he really means is that we know a bargain when we see it and we'll risk life savings and limb to get it.

For us, happiness is making our way through the dumpsters of life and coming away with a real treasure.

A group at SOG VI figured they could hand a *Micro C* reader any computer ever manufactured and by the end of the day he'd be doing something useful with it. (Heck, that's better than DEC could do with the Rainbow.)

Compview's Vedit+

Twas the day before the Faire and all through the office not a creature was stirring... except me. (I was working on a solid case of memory block trying to get everything together for the booth.) When what should my hypertense ear behold, but a quiet little voice, singing a four-letter word.

"Help."

It was Carol, trying to Ventura a giant text file.

"It isn't working," she said, pointing to the screen full of garbage.

I loaded up Vedit and started scrolling through the text file. I found incredible numbers of spaces at the ends of lines (sometimes over 100), anywhere from one to six carriage returns between paragraphs (Ventura requires two in ASCII files), double, triple, and quadruple spaces between words, and whole paragraphs indented by whatever happened to be lying around.

I started manually deleting spaces, removing carriage returns, locating tags... An hour later I realized that the Faire would be history before I finished this 72K file. (And there were more, many more, files coming.)

Fortunately Vedit has an incredibly powerful command mode. It's so powerful and complex that I hadn't used more than 1% of it. Best of all, its command mode accepts whole files of instructions, C-like in their power, brevity, and opacity.

C++

FULL COMPILER – not a preprocessor

- Complete software-development system
- Selectable AT&T C++, ANSI C, K&R C
- Based on proven Oregon Software technology
- Generates extremely fast, compact code
- Comprehensive error checking
- Backed by responsive support engineers
- Strongly typed language

DATA ABSTRACTION FACILITY (classes)

- Operator overloading
- Information hiding/sharing
- Constructors/destructors

YES! I want top performance
at an affordable price

To order, or for more information, call 1-800-874-8501
6915 SW MACADAM, SUITE 200, PORTLAND, OR 97219

OREGON SOFTWARE

Professional Products for Software Development

Reader Service Number 85

New from MSC

\$99.99

NanoLISP

A Common LISP Interpreter for MS-DOS

NanoLISP contains a large useful subset of the Common LISP standard, including most Common LISP operations, and adheres *precisely* to the specifications of the standard.

- Helpful Features: extremely thorough error-checking, explicit error messages, excellent debugging facilities
- Advanced Features: lexical and dynamic scoping, closures, lambda-list keywords, structures, bit-arrays, generic sequence functions, transcendental functions, output formatting
- UnCommon Features: graphics, low-level DOS access, customization
- Sample AI application programs
- Fully-indexed 150-page reference manual
- Unlimited free technical support

Free shipping on prepaid orders.

MSC Microcomputer
Systems
Consultants

P.O. Box 747, Santa Barbara, CA 93102

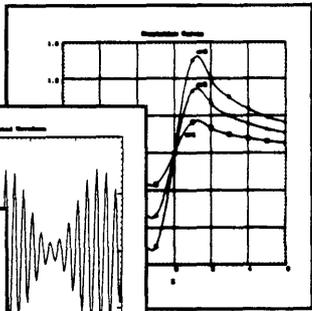
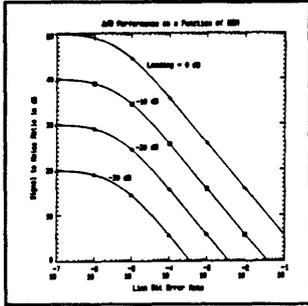
(805) 963-3412

Reader Service Number 36

SCIENTIFIC GRAPHICS

Presentation Quality Graphics
For Printers and Plotters

Screen Graphs for
Fast Previews



Curve-Smoothing
Interpolations

Legends Placed Anywhere

Built-In Editor Auto/Manual Scaling Log/Lin/SemiLog

An Indispensable Tool For Technical Professionals

GraphStar

Special Introductory Price **\$79**

System Requirements: IBM-PC, XT, AT or Compatible running DOS 2.0 or higher. Screen graphs require CGA. Printer Graphs require Epson EX, FX, JX, RX, HS; Star Gemini, Radix, SD, SG, SR; IBM Graphics; or compatibility with one of the above. Plotter graphs require HP-GL compatibility.



P.O. Box 956, Dept. C, Valley Forge, PA 19482
For Technical Information: (215) 269-0198
Reader Service Number 60

I needed to strip all spaces between the last printable character in a line and the <CR><LF>. I opened the manual.

Okay... |w is white space, any amount of white space, so if I replace white space followed by <CR><LF> with just the <CR><LF> I'd be set.

Now, back to the top of the file and I can search for three carriage returns in a row and replace them with two. Then back up two lines and search for three again.

After the returns I looked for an @ symbol at the beginning of a line (means it's a Ventura tag) just to make sure it was preceded by two sets of <CR><LF>. If there's only one set, then I'd add another. I used IF/THEN for the test. (Hey, they did a Z80 assembly to 8088 assembly translator using a single command file.)

Once I'd dug through the manual enough to get rolling, I was having fun. By midnight I had a whole set of utilities.

I threw half a dozen functions together (to do everything I've mentioned so far) into a file called CLEANUP and turned it loose. In under 60 seconds it had turned a raggedy 72K file into 42K of the cleanest, neatest, most beautiful pages.

Ventura was happy. Carol was happy (so happy that now she lets me write a new one for her at least once a week). And, I was happy (she could handle all the file cleanups). Now if Vedit could just locate the rest of my booth materials... (where's that manual?).

David Thompson
Editor & Searcher

ALL SALES SUBJECT TO THE TERMS OF OUR 90 DAY LIMITED WARRANTY. FREE COPY UPON REQUEST.

CANON 80 COLUMN PRINTER - \$29.95

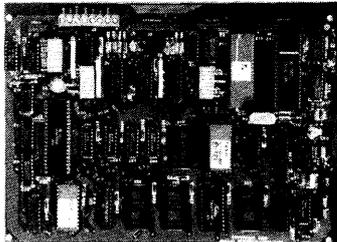
ORIGINALLY MANUFACTURED FOR THE PC JR. BUT WITH OPTIONAL CONNECTOR WILL WORK WITH PC, XT, OR AT. REQUIRES SERIAL I/O. THIS THERMAL PRINTER IS QUIET AND USES EASY TO GET 8 1/2 IN. ROLLS OF PAPER, 50 C.P.S., UPPER AND LOWER CASE, PLUS GRAPHICS. ORIGINAL LIST PRICE \$199.00. ADD \$3.00 FOR PC/XT CONNECTOR. ADD \$5.00 UPS.

THE NEW ZRT-80 CRT TERMINAL BOARD!

A LOW COST Z-80 BASED SINGLE BOARD THAT ONLY NEEDS AN ASCII KEYBOARD, POWER SUPPLY, AND VIDEO MONITOR TO MAKE A COMPLETE CRT TERMINAL. USE AS A COMPUTER CONSOLE, OR WITH A MODEM FOR USE WITH ANY OF THE PHONE-LINE COMPUTER SERVICES.

FEATURES:

- * Uses a Z80A and 6845 CRT Controller for powerful video capabilities.
- * RS232 at 16 BAUD Rates from 75 to 19,200.
- * 24 x 80 standard format (60 Hz).
- * Optional formats from 24 x 80 (50 Hz) to 64 lines x 96 characters (60 Hz).
- * Higher density formats require up to 3 additional 2K x 8 1/16 RAMS.
- * Uses N.S. INS 8250 BAUD Rate Gen. and USART combo IC.
- * 3 Terminal Emulation Modes which are Dip Switch selectable. These include the LSI-ADM3A, the Heath H-19, and the Beehive.
- * Composite or Split Video.
- * Any polarity of video or sync.
- * Inverse Video Capability.
- * Small Size: 6.5 x 9 inches.
- * Upper & lower case with descenders.
- * 7 x 9 Character Matrix.
- * Requires Par. ASCII keyboard.



\$89⁹⁵ A&T
ADD #ZRT-80 \$50
(COMPLETE KIT, 2K VIDEO RAM)

OUR BEST SELLER!

FOR 8 IN. SOURCE DISK OR PC-XT FORMAT 5 1/4 IN. ADD \$10

Digital Research Computers

P.O. BOX 381450 • DUNCANVILLE, TX 75138 • (214) 225-2309

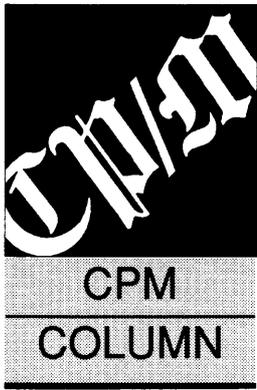
Call or write for a free catalog on Z-80 or 6809 Single Board Computers, SS-50 Boards, and other S-100 products.

TERMS: Add \$3.00 postage. We pay balance. Orders under \$15 add 75¢ handling. No C.O.D. We accept Visa and MasterCard. Texas Res. add 6-1/4% Tax. Foreign orders (except Canada) add 20% P & H. Orders over \$50 add 85¢ for insurance.

Reader Service Number 32



The Micro C staff from top left, Bruce Eckel, Melinda Zoehrer, Larry Fogg, and David Thompson. Bottom row from left, Gary Entsminger, Tammy Westfall, Cary Gatton, Carol Steffy and Laura Logan.



New Z180 Processor

And, Variable Half-Intensity Mod For Kaypros

New From Zilog

In a recent news release Zilog announced another addition to the Z80 family — the Z180. This 8-bit CMOS processor runs at 8 MHz and supports the Z80 instruction set. It also incorporates seven new instructions, including multiply.

Main features of the chip include:

- One MByte of memory address space.
- 64K of I/O
- Two DMA channels.
- Wait state generation.
- Programmable DRAM refresh controller.
- Two full-duplex asynchronous serial channels.
- Clock generator.
- Interrupt controller.

The Z180 comes packaged in either a 64 pin DIP or a 68 pin PLCC and will set you back \$11.45 each in lots of 100.

Zilog

210 Hacienda Ave.
Campbell, CA 95008
attn: Jim Magill

Wordstar 4.0 & Kaypro Intensity Mod

My my, what a nice surprise! I received a letter from Micropro offering to upgrade my Wordstar 3.3 to 4.0 for about \$95. They promised to have fixed all those annoyances that have resulted in the patch tables for WS 3.3 on the pbbs. Should I buy?

Decisions, decisions! After all, I have a rather well patched up version of 3.3 for my two Kaypros and make do nicely with that. Most of the patches concern my "outlaw" printer, a Canon PW-1156A, aka Centronics H136A. It's almost an FX-80 clone — almost. It has NLQ but I haven't been able to switch it on from inside the program without giving up something else. Maybe the new version has a driver for it. I or-

dered it. And it came. All six diskettes. And a large book. And I'm in love. Spend the money.

I have a lot more control over my printer now (except that the NLQ still doesn't work from inside the program). All the goodies are there, like proportional spacing, variable character width and mailmerge built in.

Micropro uses reverse video to mark blocks in the text, just as before. But now they use half intensity video for the text so that they can use normal video for highlighted commands and bold text. They recognize that this may cause problems and have provided a patch to reverse the situation so that text is normal and emphasized text is dim. On my monitor, neither is acceptable. To see the dim, the bright must be so bright as to bloom and is just a bit out of focus, while the dim is still dim.

Since the video brightness is a function of the hardware, a hardware fix is required. This turns out to be a simple thing and is probably worthwhile, even if you don't get the upgrade.

Video to the CRT is buffered through U1, a 7406 open collector inverter with R5, 150 ohms, as the pull up. To get half intensity, another inverter in U1 acts as a clamp (through R6, 300 ohms) to give about 2/3 of full intensity drive to the CRT.

300 ohms is somewhat low. On my old Kaypro, the difference was just tolerable. But on the new one it wasn't. The solution is to change the value of R6. Larger values make the half intensity mode brighter and do not affect the normal video. You can either replace R6 with about 360 ohms or with a 1K pot and have a contrast control. I chose the latter and did the installation differently on each computer.

On one I removed U1 and bent pin 4 out. I then soldered a tiny 1K pot between U1's pin 4 and pin 6. I reinstalled

the chip and now can set the level of half intensity from full to none at all. On the other system I installed the pot on the back of the cabinet where my reset switch used to reside.

You might wonder why I didn't simply remove R6 and solder the pot there. I thought about it, and while I've been certified to NASA 5200 and Mil-S-45783, I have an aversion to reworking boards unless it's really necessary. I've lost a few that way. This was an experiment and for all I knew, I'd have to try several values of pot to get what I wanted. The mod worked so well that I just left it. If the chip ever dies, I'll probably pull R6 and add the pot.

Philip W. Elrod

3245 Spring Dr. NW
Doraville, GA 30360

Imperfect Date

I am wondering if any of your readers has a solution to the "date" limitation of Perfect Filer v.1.2.

The limitation is that the maximum number it will accept for the year is 88. So the date feature of the program will be of no use after 1988.

I have called and written Kaypro, Thorne EMI, and Perfect Software in Eugene, Oregon, getting either "I don't know" or no response at all. I would appreciate it very much if you or one of your readers could help.

Fred Horton

180 Marsh Ave.
Metuchen, NJ 08840

◆ ◆ ◆



Bringing Up The Micro C 256K Mod

On '83 Kaypros

By John S. Allen
7 University Park
Waltham, MA 02154
(617) 891-9307

This is a fine-tuning on the popular 256K upgrade for 83 Kaypros. The hardware portion of this mod is not trivial and we strongly suggest that you not try it unless you build digital circuits on a regular basis. Also, this adds a 190K RAM disk, not additional program memory. See issues #30 ('83 Kaypros) and #34 ('84 Kaypros) for details on the modification.

The 256K mod for '83 Kaypros described in *Micro Cornucopia* issue #30 (June/July 1986) is inexpensive, and hardware installation is easy. But software installation can be tricky, so read the following in conjunction with the documentation (Micro C Users' Disk #K47 or Micro C bulletin board/(503) 382-7643).

I now have the mod running on my Kaypro II as a 191K RAMdisk, with a Micro Cornucopia Pro-8 version 3.3 monitor ROM and ZCPR1 from Micro C Users' Disk #K22. Here's what I learned.

CP/M Version

Early Kaypro II's were delivered with a CP/M version that had a two-line sign-on message. There's a big problem in using this CP/M with the 256K mod: you must use MOVCPM.COM to generate a 63K CP/M, but the two-line CP/M's MOVCPM.COM writes garbage characters into the numeric keypad table in the BIOS.

If you later try to change these characters so you can use the keypad, the computer will crash. Apparently, the early CP/M version uses the keypad table bytes for internal BIOS functions if the CP/M is less than 64K.

The CP/M which supports the 256K mod has a single-line signon message. "Kaypro II 64K CP/M vers 2.2." Its MOVCPM.COM resets the numeric keypad to the characters shown on the keytops. You will lose keypad settings which you had programmed earlier, but you can restore them — see below.

MOVCPM.COM works only with an un-

modified 64K CP/M of the correct version — no ZCPR or custom BIOS, thank you! If you run MOVCPM.COM on a different or enhanced CP/M, the message "SYNCRONIZATION ERROR" will grace your screen. As you may have noticed, the message is spelled wrong, adding insult to injury. But it need not defeat you. Pile on the enhancements after you have created your 63K CP/M.

Configuring Without CONFIG

Once you've created your 63K CP/M, you may run the RAMDRIVE.COM program in the 256K package to bring up the RAMdisk. You must use the RAMINIT.COM program as well, to format the directory whenever you power up the computer. You need not use RAMINIT.COM after a reset. Files in the RAMdrive will survive a reset, as long as you do not run RAMINIT.COM.

Do not use the Kaypro II CONFIG.COM program with a 63K CP/M. It will trash the system tracks. You may run the CONFIG83.COM program which comes with the Micro C Pro-8 monitor ROM, but you must avoid CONFIG83.COM options which alter the BIOS, such as changing the keypad settings.

You can change the cursor. The block cursor, installed by pressing the Return key, gets high marks for visibility in a field of text. But to avoid nasty mistakes, erase CONFIG83.COM from your system disks once you've set it up the way you like.

If you want your RAMdisk to come up every time you boot CP/M, install the CBIOSR.HEX file (in the 256K software package) using DDT.COM, as described in the 256K package's instructions. You may use CONFIG83.COM, with the same limitations listed above.

To configure your keypad and cursor keys (I told you we'd get to this), use DDT to modify the CPM63.COM file which you saved when generating your 63K CP/M. Type DDT CPM63.COM, then use the DDT s (set) command to change the bytes from 1FB5H to 1FC6H. These correspond to the four cursor

keys, then the numeric keypad in the order:

```
<up> <dn> <L> <R> 0 1 2 3 4 5 6 7 8 9
- , <CR> .
```

Adding ZCPR1

For your 63K CP/M, even on a Kaypro II, you need a 63K, Kaypro 4 version of ZCPR1, such as the one on Micro C Users' Disk #K22. The programs from disk #K22 which work with '83 Kaypros are 4INSTALL.SUB and ZCPR4S.HEX.

After completing all of your modifications and enhancements, use SYSGEN.COM to write your CPM63.COM file onto your disk's system tracks.

Questions

Some application programs reconfigure the cursor; for example, NULU12.COM and IMP245.COM make it into a "0". I'm not sure whether this oddity has to do with ZCPR1, the Pro-8

monitor, or the RAMdisk software, since I installed them all at nearly the same time. Does anyone know a way around this problem?

EX 1.4 won't run in the modified CP/M. It seems to be incompatible with ZCPR1. I'm stuck with kludgy, slow SUBMIT.COM.

I have occasionally received "NO DIRECTORY SPACE" messages from the RAMdisk after running the CBIOSR (automatic RAMdisk initialization) CP/M a few times. This happens even after a cold boot — a mystery, since nothing should rewrite CP/M system tracks. Writing new system tracks onto my system disk cures the problem, at least for a while. I suspect a hardware or timing problem. Any clues?

Dr. Michael Liddle, the author of the 256K mod, is using an earlier Pro-8 ROM version and ZCPR3 RAMdisk software, which he wrote. Note that he does not provide the entire ZCPR3 system or the utilities you will need to run it. You will have to configure these yourself.

Dr. Liddle's documentation mentions a special SYSGEN8.COM needed in ZCPR3, which he doesn't provide. I imagine that this must be part of the standard ZCPR3 utilities package. But he also mentions a RAMINITD program, needed to clear the RAMdisk in a ZCPR3 system. It's not in his software package from Micro C. Double check on the availability of this program before you try to bring up ZCPR3 with the RAMdisk.

Summing Up

The RAMdisk is easy to install in hardware and works fine once you're past a few software difficulties. It speeds up the computer tremendously. You do lose a few features, because there is no CONFIG program which fully supports the altered BIOS. Anyone who can solve these minor software problems will be doing a real service to users of the mod.

◆ ◆ ◆

Neural Nets!
Seen Twice in:

WHAT'S NEW

BYTE

NetWurkz for the IBM PC is a neural-network tutorial. It models associative memory to find the nearest match to your input word. The emulator is limited to about 1000 neurons. The program comes with a PL/D compiler (NetWurkz is embedded within the PL/D language as data statements). The program also comes with source code.

To run NetWurkz, you need an IBM PC, XT, AT, or compatible with at least 256K bytes of RAM.
Price: \$79.95.
Contact: DAIR Computer Systems, 3440 Kenneth Dr., Palo Alto, CA 94303, (415) 494-7081.

Nanobytes
Feeling left out of those dinner-party conversations because everyone's discussing neural networks and you're still trying to figure out local-area networks? DAIR Computer Systems (Palo Alto, CA) has something that might help: a program called NetWurkz, which the company says is a neural net tutorial and simulation package intended to serve as an introduction to neural modeling and associative memory concepts. The \$79.95 program runs on an IBM PC with 192K bytes of memory and MS-DOS 2.0 or higher. . . . We saw

FEBRUARY 1988 • BYTE 11

46 BYTE • OCTOBER 1987

Learn More for Yourself
Call (415) 494-7081

DAIR Computer Systems
3440 Kenneth Drive
Palo Alto, CA 94303

VISA



Reader Service Number 90

ShareWare™

BROWN BAG Software™

PC Outline!

- Organize Reports and/or Papers
- Brainstorming
- Managing a Project

SOME FEATURES:

- Structured Indentation
- Powerful Editing
- Outline Rearrangement Functions
- Choice of Multiple Numbering Schemes
- Margin Control
- Centering, Left & Right Justification
- Optionally Memory Resident
- Requires Less Than 90K
- 9 User Definable Windows
- Multiple-Line Text
- Import File from dBase™, DataBase™, Wordstar™, MaxThink™, and ThinkTank™.

\$89⁹⁵
Price \$99 (Shipping and Handling)

SOME USES:

- Things-To-Do List
- Name & Address Lists

Give us a call now at:

1-800-523-0764
IN CALIFORNIA 1-800-323-5335 or (408) 559-4545

VISA, MasterCard accepted or mail your check to:
File #179, Box 60000, San Francisco, CA 94160-0719

SHAREWARE DISK AVAILABLE FOR \$10.00

© Copyright 1988, Telemarketing Resources, Inc.

Reader Service Number 87



TECHTIPS

Technical Tips

Parallel Problems

I have two trivial but useful items for folks doing computer support. First: A \$2 IC (74LS374) will fix most IBM type parallel port boards fried by contact with a serial printer. I've fixed Compaqs and IBM Monochrome Adapters, more times than I'd care to admit, when a user decides they're a computer genius because they discovered that funny little plug is for a printer. *Editors's note: Even though the funny little plugs look the same for serial and parallel ports, the 5 V parallel port will definitely choke on 12 V of serial.*

Most adapter cards are single layer boards. So it's just a matter of breaking out the old IC (it's the only 74LS374 on the board), blowing the solder pad holes open (heat a hole and then blow air through or use a solder sucker), and then soldering the new one in.

Second: A convenient dummy parallel printer can be made by grounding the BUSY and PE (aka "out of paper") pins through about 1K Ohms (unscientifically determined resistance that's worked on a lot of different machines).

This can be useful for debugging when you want to exercise a print function but don't want to waste the time, ribbon, or paper. The other use is with accounting packages that insist on running reams of detail before doing the close-out you need (in MBA school I learned how to get to the bottom line fast).

Stick the dummy printer in and fool 'em into thinking they've just been treated to the fastest printer in history. I use a male 25-pin subminiature "D" plug (solder type) and put 1/8 Watt, 1K Ohm resistors between pins 11 and 20 and between pins 12 and 21. Sticking a cover on the plug makes it easier to handle.

Win Wiencke
Semi-Custom Software
6807 Brennon Lane
Chevy Chase, MD 20815

Editor's note: MicroSphere has seen a rash of bad parallel ports lately. It appears that plugging a

parallel printer into a parallel port while the printer's running causes the problem, even when both the printer and computer are plugged into the same power strip (no major ground loops). When the parallel port is on the graphics card, the result is often no video (as well as no parallel output). When the port is on a multi-I/O card, only the port dies.

They are now turning off both the computer and the printer before connecting same.

I've not heard of any problems connecting together hot RS-232 ports. Those little 1488s and 1489s that everyone's using for RS-232 buffers are really bulletproof. (Although Bruce did nail one when he accidentally connected the hot side of a 110VAC line directly to an RS-232 pin.)

360K Drives On An AT

We get the occasional call from folks trying to run standard double-side, double-density drives on an AT system. The problem is that the AT can't tell a 1.2 meg drive from a 360K.

XTs ignore line 34 of the floppy drive interface, ATs don't. Line 34 carries an active-low READY signal generated by the floppy drive. READY stays high until the drive has spun up to speed and stabilized, usually a few hundred msec. For some reason, if READY is disconnected, the AT assumes the drive is a 360K.

So, just cover the card edge contact for line 34 with a thin sliver of Scotch or electrical tape.

Meanwhile, if you're stuck with 1.2 meg drives and want to create a disk that someone with a 360K can read, you must:

(1) Find a new, unformatted, DSDD disk (or erase an already formatted disk with a magnet, very thoroughly).

(2) Format the disk as a 360K disk, but you must do it with the 1.2 meg drive. The command is:

```
FORMAT A:/4 (the /4 forces a 360K format)
```

(3) Now place the files on the disk using COPY.

You can continue to use this same disk for transferring files from the AT to a system with 360K drives as long as the 360K drive doesn't write anything onto the disk. (Remember, erasing a file writes data into the directory.) For

Micro Ads

A Micro Ad is the inexpensive way to reach over 22,000 technical folks like yourself. To place a Micro Ad, just print out your message (make it short and sweet) and mail it to Micro C. We'll typeset your ad (no charge) and run it in the next available issue. You can also send camera ready copy. Rates: \$99 for 1 time, \$267 for three times, \$474 for 6 times (a best buy at only \$79 per insertion). Full payment must accompany ad. Each ad space is 2 1/4 inches by 1 3/4 inches.

THE WINDOW BOSS

The Window Boss is a powerful windowing library for the "C" language. Popup windows, pull-down menus and status lines are a breeze to implement. The Window Boss is available for The Borland, Microsoft, Data-light and Lattice Compilers. Mention this ad and receive a 20% Discount.

Star Guidance Consulting, Inc.
273 Windy Drive • Waterbury, CT 06705
203/574-2449

Reader Service Number 96

CONSULTANTS AND SYSTEMS INTEGRATORS!

Surround & integrate your DOS software with

The Weiner Shell

The high-level TSR language.

- Does telecommunications, windows, menus and over 130 functions in the background of any application.

- Supports dBASE files.

From Gryphon Microproducts
(301) 384-6868 (301) 384-7238 (BBS)

Reader Service Number 97

Periscope Debugging Systems for IBM Personal Computers

FREE INFORMATION
800/722-7006

The **PERISCOPE** Company, Inc. 1197 PEACHTREE ST. PLAZA LEVEL ATLANTA, GA 30361 404/875-8080

Reader Service Number 94

SourceTools™

FULL SOURCE CODE MANAGEMENT

- Document revision history ■ Audit trails
- Branching and key word insertion ■ Utilities

POWERFUL MAKE FACILITY

Available for VAX, PDP-11 & MS-DOS.
To order or for more information, call 1-800-874-8501.

OREGON SOFTWARE
6915 SW Macadam Ave., Portland, OR 97219.

Reader Service Number 85

FUTURE86, THE LANGUAGE

Variable level, extensible, rommable language gives complete control. Including 80186/88. Simple syntax. Easy programming. Compact code. Easily optimized for speed or size. Interactive debugging facilities. Library source available. Investigative FUTURE86 . . . world class! Used in thousands of applications. Prices start at \$349.

Development Associates
1520 S. Lyon, Santa Ana, CA 92705
(714) 835-9512

Cross Assemblers Universal Linker Librarian

hosted on: **PC/MS DOS, micro VAX,
VAX VMS, VAX UNIX/ULTRIX**

Targeting over 30 microprocessors



Energic inc.
(215) 362-0968 19 Jenkins Ave
Lansdale, PA. 19446

Reader Service Number 82

GRAPHICS WINDOWS FOR IBM AND COMPATIBLES with SKYLIGHTS™

Screen Management Package

- Interactive Screen Editor
- Runtime C Library
- Grabber and Demo Maker Utilities

Information and Demo available on request
The Report Store, 910 Massachusetts St., Suite
602-MC, Lawrence, Kansas 66044 (913)842-7348

Reader Service Number 98

TurboGeometry Library (Source & Manual)

Turbo Pascal 4.0 C, Mac, & Microsoft C.
Over 150 2 & 3 dimensional routines including: Intersections, Transformations, Equations, Hidden Lines, Perspective, Curves, Areas, Volumes, Clipping, Planes, Matrices, Vectors, Distance, Poly Decomp. IBM PC & Comp., Mac.

\$99.95 + S&H. Visa MC AE
DISK SOFTWARE, INC.
2116 E. Arapaho, No. 487
Richardson, TX 75081 214/423-7288

Reader Service Number 80

PC C SHELL

A faithful implementation of the Berkeley Unix C Shell for MS-DOS.

- Full History Syntax
- Looping Statements (while, foreach)
- Shell Variables and Arrays
- Cursor Key Command Line Editor
- Runs as default command processor
- WORKS ON ANY MS-DOS COMPUTER!!

\$50.00
Omega MicroConsulting
722 Rundell Street • Iowa City, IA 52240
(319) 338-6053

Reader Service Number 77

Instruction Manuals for your product

Written in an informal style by
techies for real users.

Free quote.

Rare Earth Services, Inc.
3115 Willow Road NW
Roanoke, VA 24017
703-343-4565

Reader Service Number 73

DEVELOPERS, NOW AVAILABLE

Sof-Tel Inc., in their goal to provide innovative Software Solutions, now has ready for you two new IBM PC software packages. Do you need Modem dialing and interrupt Comm port support? Get AUTO-DIAL! User-definable script files allow use with many modems. Send \$49.95 plus \$3 P&H. Also available. STAGE2-PC, the PC version of the mainframe macro processor STAGE2. Disk plus printed manual only \$49.95 plus \$3 P&H. (Florida residents add 6% sales tax.)

Soft-Tel Inc.
P.O. Box 5116, Lighthouse Pt. FL 33074
(305) 942-6671

Reader Service Number 92

8051 Z8 / Super8 C COMPILER

* Call today for a FREE technical bulletin *

MICRO COMPUTER CONTROL
P.O. Box 275 - Hopewell, NJ 08525 USA
Telex 9102404881 MICRO UQ

(609) 466-1751

Reader Service Number 100

Unbelievable!

SOURCER™

Creates commented source code and listings from memory, COM or EXE files.

- **CLARIFY UNDOCUMENTED CODE**
- **EASILY MODIFY PROGRAMS**

SOURCER™ creates detailed commented listings and source code directly suitable for assembly. Built in data analyzer and simulator resolves multiple data segments and provides detailed comments on interrupts and subfunctions, I/O ports and much more. Determines all necessary assembler directives. Complete support for 8088 through 80286, V20/V30, 8087 and 80287 instruction sets. No other product comes close to the output quality of SOURCER.

BIOS SOURCE

PS/2 ▪ AT ▪ XT ▪ PC ▪ Clones

- **CHANGE & ADD FEATURES**
- **CLARIFIES BIOS INTERFACES**
- **SPECIFIC TO YOUR MACHINE**

The bios pre-processor to SOURCER provides the first means to obtain accurate legal source listings for any bios! Identifies entry points with full explanations. Resolves PS/2's multiple jumps for improved clarity. Provides highly descriptive data labels such as "video_mode" and "keybd_q_head," and much more. Fully automatic.

SOURCER	\$ 99.95*
SOURCER	
w/BIOS Pre-Processor	\$139.95*

(*OUTSIDE USA, ADD \$15 SHIPPING; CA RES. ADD SALES TAX)

All our products come with a 30 day money back satisfaction guarantee. Not copy protected. To order or receive additional information just call!

800-538-8157 x 811 (outside Calif.) 800-672-3470 x 811 (inside Calif.)

V COMMUNICATIONS
3031 Tisch Way, Suite 200, Dept. MC
San Jose, CA 95128
(408) 296-4224

PS/2, AT, XT and PC are trademarks of IBM Corp.

Reader Service Number 62

safety's sake, write-protect the disk whenever it's in a 360K drive.

Of course, the 1.2 meg drive will read anything created by a 360K drive. If the 1.2 meg writes *anything* onto the disk, however, what is written can only be read, dependably, by another 1.2 meg drive.

The 1.2 meg drive reads and writes a track that's half as wide as a 360K track. (In 1.2 meg mode, the drive writes twice as many tracks and writes nearly twice as many sectors per track (much higher bit density). In 360K mode, the 1.2 meg drive double-steps as it goes from track to track and it lays down bits at the lower 360K density.)

When you write over a 360K track with a 1.2 meg head, you leave half of the original 360K track. Another 1.2 meg drive will only see the latest data but a 360K head will see the entire, wide track, half written by the 1.2 meg head, half written by the earlier 360K head.

Quickies

Many dual-speed systems use the cursor as the speed indicator, underline for slow, larger block for fast. However, some programs can reverse the cursor. If you're running into formatting or other speed-related problems, you might suspect your cursor's been mislead.

Most people bring up a new hard drive by connecting it to their XT, getting into DEBUG, and entering G=C800:5. That's fine, but it doesn't give the drive a chance to warm up. Many hard drive experts strongly recommend that you let the drive run at least 2 hours before formatting. That way, temperature can stabilize before format begins. Then plan to format twice followed by a reformat every 6 months or so.

Sticky Floppies

Another problem that crops up from time to time at the old home place has to do with disk errors. Micro C receives a lot of disks from all over the world. And often they're a real challenge to read. They may be unlabeled Pied Piper Quad or even Gnat System 10 disks (I'm serious).

Sometimes, even if we know the disk format, the familiar "Not ready error reading disk ..." pops up. All may not be lost. If one of your disks (especially

one subjected to the rigors of a trip through the Postal Service) seems to have given up the ghost, try the following.

Grab the media by the inner hub (being careful not to touch the actual data surface) and turn it by hand. Exposure to the mails may have caused the media to stick to the jacket. (On hot, muggy days, postmen, too, stick to their jackets.) If you can free the disk by hand, fine. If not, run each of the four edges of the disk back and forth over the edge of a desk, pressing gently. It'll be obvious if you've done any good — the disk will turn more freely (and you'll be able to read it).

Micro C Staff

exec()

I'm writing regarding Ron Miller's article in *Micro Cornucopia* Issue #38 (Nov/Dec 87). He seems to be confusing the system(), spawn(), and exec() functions.

His exec() function is a great space saver. But it's really a spawn() function. exec()s overlay the current process while spawn()s run a new process and then return.

Some investigation of the Turbo C library code (using, by the way, Microsoft CodeView) shows that the spawn() and exec() functions do contain a lot of overhead. One warning: with the exec() function as written, it's a good idea to flush all buffers before starting a new process, in case that process wants to access the same files.

system() is a spawn of the command shell, allowing redirection, piping, and the execution of batch files. system() requires access to the COMSPEC environment variable to find the command shell. If Ron's TSR code frees its assigned environment block before going resident (as many TSRs do), then it's quite possible that the original environment block has been destroyed. So system() fails where his exec() function does not.

Milton E. Palmer III
Glenn L. Martin Wind Tunnel
University of Maryland
College Park, MD 20742

◆ ◆ ◆

SCREEN MANAGER
 MENU, WINDOW and DATA ENTRY Support for the Professional Programmer! Interfaces to most languages. BASIC, C, FORTRAN, COBOL, PASCAL, ASSEMBLER. 100 Page Manual. 30 day money back guarantee. No Royalties.
 from **\$79** VISA/MC
 The West Chester Group
 P.O. Box 1304
 West Chester, PA 19380
 (215)644-4206
CALL FOR FREE DEMO

Reader Service Number 78

68000 SOFTWARE
 • K-OS ONE operating system uses MS-DOS disks with source code **\$50**
 • K-OS ONE manual **\$10**
 • HT68K SBC w/K-OS ONE . **\$395**
 • Screen Editor Toolkit **\$50**
 • HT-FORTH **\$100**
 • BASIC **\$149**
 Free newsletter & spec sheets
HAWTHORNE TECHNOLOGY
 1411 S.E. 31st Ave., Portland, OR 97214
 (503) 232-7332

Reader Service Number 34

35 mm SLIDES FROM YOUR IBM/PC
 Computer Slide Express*
 Finally, a one day service to convert your graphic files into 4000 line color slides from your disk for \$9 or less!
24 HR SERVICE Since 1971
 Visual Horizons*
 180 Metro Park
 Rochester, N.Y. 14623
 (716) 424-5300

Reader Service Number 79

16 Megabytes EMS and/or Extended Memory
 • Works on 8 or 16 bit bus
 • 16 bit transfer on AT bus
 • Single board design
 • Includes RAM disk and extensive diagnostics
 • Quantity/OEM discounts
 XT and AT Compatible
 Designed, Manufactured, Sold and Serviced by **Rech**
 904 North 8th St. Lake City, MN 55041 (612) 345-4555

Reader Service Number 54

THE COMMAND System
 • Instant Access to Any Directory on Your Hard Disk
 • Automatically Generated Short Names For Every Directory
 • All Our Programs Use Short Names
 • Copy, Erase, Rename, Move, Compare, Search with:
 • Consistent Command Structure
 • Multiple Operations On a Line
 • Built-in Help
 • Super Programs to Show Files & Directories
Call 301/969-8068 ONLY with your order now. \$75
CompuMagic, Inc.
 P.O. Box 437 • Severn, Maryland 21144

Reader Service Number 95

CIRCUIT BOARDS FROM YOUR SCHEMATIC
 • ACCURATE CAD LAYOUT
 • PHOTOPLOTTED ARTWORK
 • PENPLOTTE SCHEMATICS
 • MULTILAYER AND SM
 AQUARIUS GROUP, INC.
 1450 VIEWTOP DRIVE
 CLEARWATER, FLORIDA 34624
 (813) 531-5464

Reader Service Number 102

Want to Throw Out your U.P.S. Log Book? Now You Can!
 Here's what EASY-SHIP can do for you:
 Automatic U.P.S. Shipping to all of U.S & Canada.
 Fast, Easy Multiple-Shipments with All Options.
 U.P.S. Approved Shipping Labels & C.O.D. tags.
 Access to your ASCII Customer Data File.
 U.P.S. Approved Reports and Manifest Summary.
 Approved Nationally by United Parcel Service.
 NO MORE MANUAL LOGGING! And more!
 For all IBM PC, AT, OS/2 Systems. Only \$365 + \$3 S/H.
Stat Supply Company
 20214 Brondesbury, Katy, TX 77450
 (800) 666-4567 or (713) 492-1931

8051 SIMULATOR
 Superb full function debugger simulator supports all 8051 modes of interrupt, just like the real thing! Full disassembler and many unique features, ONLY \$150.00
CROSS ASSEMBLERS
 For Z80, 8080/8085, 8048, 8051 and 8096 still \$75.00 each!
LEAR COM COMPANY
 2440 KIPLING ST., SUITE 201
 LAKEWOOD, CO 80215 303/232-2226

Reader Service Number 84

CASINO CARD GAME PAC \$29
POPUP VT100 EMULATOR \$49
POPUP TIME & BILLING \$59
DES SECURE ENCRYPTION \$59
BBS & REMOT ACCESS \$69
QUICK DATABASE & MAIL \$99
STOCK TRADING SYSTEM! Demo \$49
 IBM PC Compatibles GUARANTEED SATISFACTION or your Money Back! Don't Delay
 Get Our Free Catalogue Today
Hawkeye Grafic Inc. Box 1400 OLDSMAR FLORIDA 33557 Dial 813-855-5846

CBASIC, CB80, CB86 USERS
 Convert your Cbasic and Cbasic compiler programs into C with MB86! MB86 supports Access Manager, Display Manager, and allows your new C programs to use existing CB86 data files. MB86 breaks CB86 file size and record number limits and enables network support. MB86 enhances the Cbasic language with new features as well as fixing many old CB86 problems. Versions of MB86 are available for Microsoft C, Turbo C, Aztec C, and Microport Unix. Call or write for more information. We also carry a complete line of enhancements for CB86 and Pascal MT +.
Minnow Bear Computers
 P.O. Box 2233 Station A
 Champaign, IL 61820-8233
 (217) 344-1113

Reader Service Number 83

RAM DISK
S-100
2 Meg, Port I/O
New, Warranted
\$775
 S. Lugert
 439 Peck Slip or call:
 NY, NY 10272 718-622-0654

Reader Service Number 52

OPT-TECH SORT/MERGE
 Extremely fast Sort/Merge/Select utility. Run as an MS-DOS command or CALL as a subroutine. Supports most languages and filetypes including Btrieve and dBase. Unlimited filesizes, multiple keys and much more! MS-DOS \$149. XENIX \$249.
(702) 588-3737
Opt-Tech Data Processing
 P.O. Box 678 - Zephyr Cove, NV 89448

Reader Service Number 64

CROSS ASSEMBLERS
PseudoCode releases version 2 of its cross assemblers. Assemblers for the 8048, 8051, 8096, 8085, Z80, 6502, 1802, 6800, 6805, 6809, 68000, and 32000 microprocessor families are available. Macros, Conditional Assembly, Include Files plus extensive expression handling. Virtually no limit to program size. For IBM PC's and true compatibles with MS-DOS 2.0 or greater, and 256K memory. Complete with printed manual for \$35.00. Each additional is \$20.00. (Michigan residents add 4% tax).
 Visa/MC. Order from distributor:
Micro Kit
 6910 Patterson, Caledonia, MI 49316
 616/791-9333

Reader Service Number 75

Tentative List Of Speakers

Friday, July 15

Louis Baker

How to get your Technical Book Published

Joe Bartel, Hawthorne Technology

Generic MicroProcessors Hitachi HD641016

Bill Weinman

Why C is not the Devil's Programming Language

Earl Brabandt, Intel Corp

PS/2 Design

Mike Sequiera

Chaos 101

Jim Warren, Founder West Coast Computer Faire

(To be announced)

Bruce Eckel, Micro C Writer

C++ Grows Up

Bill Davidson, PC Tech

All About SCSI

Earl Hinrichs, PC Tech

3D Graphics on the 34010

Willy Steiger, Logitech

Inside Logitech's Modula 2

Greg Lobdell, Microsoft

The Design of Programming Languages

Allyn Franklin, Drive Masters

Drive Diagnostics

Micro C

Staff forum

Saturday, July 16

Mike Donovan

Pseudo Concurrency Using Probabilistic Technique

Don Doerr, National Advancement Corp

Diagnostic Software for XT & AT

Tom Ochs, Scientific Software, Inc.

Numerical Applications

Scott Ladd

C Compilers and C Tools

Andy Bakkers

(To be announced)

Gregg Tolleson, Intel Corp

Micro Channel Architecture

Greg Wolfson, Intel Corp

Starlan, Ethernet, & Cheapernet

Paul Voda, Complete Logic Systems

Trilogy: Designing A New Language

Dean Klein, PC Tech

Advanced Graphics Processors

Bill Davidson, PC Tech

Lorraine Systems

Michael Vore

Packet Radio

Oregon Software

SOFTWARE FORUM

Saturday Night Banquet Keynote Speaker

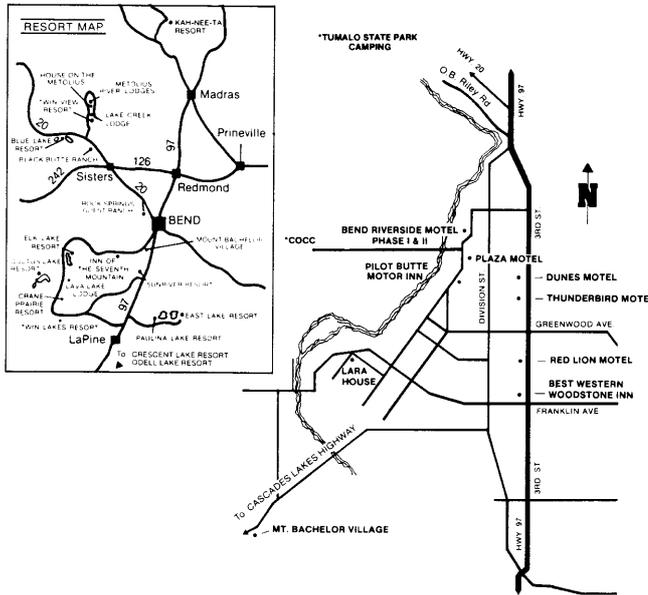
Jim Warren

When are we Going to do Something Useful with Microcomputers?

If you're not registered yet — use registration form in Issue #41
or call us at 503-382-8048 with visa or master card.

SOG Potpourri

City Map with Motel and Hotel Locations



Here are some Micro C suggestions for motel choices. Other extremes in price and luxury are available also.

BEND RIVERSIDE MOTEL RATES: \$28.00-\$44.00	389-BEND
DUNES MOTEL RATES/2: \$31.00-\$39.00	382-6811
PLAZA MOTEL RATES/2: \$21.00-\$24.00	382-1621
THUNDERBIRD MOTEL RATES/2: \$35.00-\$60.00	389-7011
PILOT BUTTE MOTOR INN RATES/2: \$20.00-\$24.00	382-1411
RED LION MOTEL RATES/2: \$44.00-\$56.00	382-8384
BEST WESTERN WOODSTONE INN RATES/2: \$38.00-\$48.00	382-1515
LARA HOUSE (B & B) RATES/2: \$45.00-\$60.00	388-4064

Quick Review of Days and Activities

Thursday, July 14th the Fun Begins

Rafting

Short Trip 2 1/2 hrs. with 3 miles of Class I-III rapids. Ticket includes BBQ dinner. (Check in at Pinckney Center, COCC, by 11 am, Thurs. July 14)

Long Trip All day with Class I-IV rapids. Ticket includes lunch and BBQ dinner. (Check in at Pinckney Center, COCC, Wed afternoon or by 7 am Thurs.)

Intro to Desktop Seminar

Seminar covers Scanning, Ventura'ing, and PageMaking. Lunch and BBQ are included, space limited. See page 39 for details.

Old Fashioned BBQ

Located in Shevlin Park, great food!

Friday and Saturday, July 15-16

Speakers, Seminars & Workshops

Build your own System

With plenty of help from Microshpere. 8088 based XT /80286 based AT /80386 Contact Cindy at 503-388-1194 to sign up.

Saturday Night Banquet

Keynote Speaker, Jim Warren, Founder West Coast Computer Faire. Price Adults, \$12 Child, \$ 7

Is There A Gap In Your Info?

Fill in your Back Issues of Micro C today!

ISSUE #1 (8/81)

Power Supply
RAM Protection
Video Wiggle
1/2 PFM.PRN
16 pages

ISSUE #2 (10/81)

Parallel Print Driver
Drive Motor Control
Shugart Jumpers
Program Storage Above PFM
1/2 PFM.PRN
16 pages

ISSUE #3 (12/81)

4 MHz Mods
Configuring Modem 7
Safer Formatter
Reverse Video Cursor
FORTHwords Begins
16 pages

ISSUE #4 (2/82)

Keyboard Translation
More 4 MHz Mods
Modems, Lync, and S10s
Undoing CP/M ERASE
Keyboard Encoder
20 pages

ISSUE #5 (4/82)

Word Processing
Two Great Spells
Two Text Editors
Double Density Review
Scribble, A Formatter
20 pages

ISSUE #6 (6/82)

BBI EPROM Programmer
Customize Your Chars
Double Density Update
Terminal In FORTH
24 pages

ISSUE #7 (8/82)

6 Reviews Of C
Adding 6K Of RAM
Viewing 50 Hz
On Your Own Begins
24 pages

ISSUE #8 (10/82)

SOLD OUT

ISSUE #9 (12/82)

BBII EPROM Program
Relocating Your CP/M
Serial Print Driver
Big Board I Fixes
Bringing Up WordStar
Cheap RAM Disk
32 pages

ISSUE #10 (2/83)

SOLD OUT

ISSUE #11 (4/83)

SOLD OUT

ISSUE #12 (6/83)

256K for BBI
Bringing Up BBII
dBase II
Look at WordStar
Double Sided Drives for BBI
Packet Radio
5 MHz for Kaypro
40 pages

ISSUE #13 (8/83)

CP/M Disk Directory
More 256K for BBI
Mini Front Panel
Cheap Fast Modem
Nevada COBOL Review
BBI Printer Interface
Kaypro Reverse Video Mod
44 pages

ISSUE #14 (10/83)

BBII Installation
The Perfect Terminal
Interface To Electronic
Typewriter
BBI Video Size
Video Jitter Fix
Slicer Column Begins
Kaypro Color Graphics Review
48 pages

ISSUE #15 (12/83)

Screen Dump Listing
Fixing Serial Ports
Playing Adventure
SBASIC Column Begins
Upgrading Kaypro II To 4
Upgrading Kaypro 4 To 8
48 pages

ISSUE #16 (2/84)

Xerox 820 Column Restarts
BBI Double Density
BBII 5"/8" Interface Fix
Kaypro ZCPR Patch
Adding Joystick To Color
Graphics
Recovering Text From Memory
52 pages

ISSUE #17 (4/84)

Voice Synthesizer
820 RAM Disk
Kaypro Morse Code Interface
68000-Based System Review
Inside CP/M 86
56 pages

ISSUE #18 (6/84)

Kaypro EPROM Programmer
I/O Byte: A Primer
Kaypro Joystick
Serial To Parallel Interface
Business COBOL
60 pages

ISSUE #19 (8/84)

Adding Winchester To BBII
6 MHz On The BBI
Bulletin Boards
Track Buffering On Slicer
4 MHz For The 820-I
64 pages

ISSUE #20 (10/84)

HSC 68000 Co-Processor
DynaDisk For The BBII
Serial Printer On BBI Sans S10
Cheap & Dirty Talker For Kaypro
Extended 8" Single Density
72 pages

ISSUE #21 (12/84)

Analog To Digital Interface
Installing Turbo Pascal
Low Intensity BBI Video
Turbo Pascal, The Early Days
80 pages

ISSUE #22 (2/85)

Xerox 820-II To A Kaypro-8
Sound Generator For the
STD Bus
Reviews Of 256K
RAM Expansion
In The Public Domain Begins
88 pages

ISSUE #23 (4/85)

Automatic Disk Relogging
Interrupt Drive Serial Printer
Low Cost EPROM Eraser
Smart Video Controller
Review: MicroSphere RAM Disk
Future Tense Begins
86 pages

ISSUE #24 (6/85)

C'ing Into Turbo Pascal
8" Drives On The Kaypro
48 Lines On A BBI
68000 Versus 80x86
Soldering: The First Steps
88 pages

ISSUE #25 (8/85)

Why I Wrote A Debugger
The 32-Bit Super Chips
Programming The 32032
Modula II
RS-232C: The Interface
104 pages

ISSUE #26 (10/85)

Inside ZCPR3
Two Megabytes On DSI-32
SOG IV
The Future Of Computing
Graphics In Turbo Pascal
104 pages

ISSUE #27 (12/85)

SOLD OUT

ISSUE #28 (2/86)

Pascal Runoff Winners
Rescuing Lost Text From
Memory
Introduction To Modula-2
First Look At Amiga
Inside The PC
104 pages

ISSUE #29 (4/86)

Speeding Up Your XT
Importing Systems From Taiwan
Prototyping In C
C Interpreters Reviewed
Benchmarking The PCs
104 pages

ISSUE #30 (6/86)

PROLOG On The PC
Expert Systems
Logic Programming
Building Your Own Logic
Analyzer
256K RAM For Your 83 Kaypro
PC-DOS For Non-Clones
104 pages

ISSUE #31 (8/86)

RAM Resident PC Speedup
Practical Programming In
Modula-2
Unblinking The PC's Blinkin'
Cursor
Game Theory In PROLOG and
C
104 pages

ISSUE #32 (10/86)

Public Domain 32000:
Hardware And Software
Writing A Printer Driver for
MS-DOS
Recover A Directory By
Reading & Writing Disk Sectors
96 pages

ISSUE #33 (12/86)

Controlling Stepper Motors
From Your PC
Introduction To Fractals
The Secrets Of MS-DOS, From
Boots To Device Drivers
Poking About In The System
With Turbo Pascal
96 pages

ISSUE #34 (2/87)

SOLD OUT

ISSUE #35 (4/87)

SOLD OUT

ISSUE #36 (6/87)

Build A Midi Interface
For Your PC
Designing A Database, Part 2
Interrupts On The PC
Hacker's View of MS-DOS
Vs 3.X
Digital To Analog Conversion,
A Designer's View
96 pages

ISSUE #37 (9/87)

Desktop Publishing On A PC
Build Your Own Hi-Res Graphics
Scanner For \$6, Part 1
Designing A Database, Part 3
Controlling AC Power
From Your PC
Expanded Memory On The
PC/XT/AT
Uninterruptable Power
Supply For RAM Disks
96 pages

ISSUE #38 (11/87)

Parallel Processing
Laser Printers, Typesetters
And Page Definition
Languages
Magic In The Real World
Build A Graphics Scanner
For \$6, Part 2
Writing A Resident Program
Extractor In C
96 pages

ISSUE #39 (1/88)

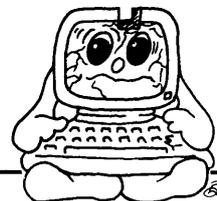
PC Graphics
Drawing The Mandelbrot And
Julia Sets
Desktop Graphics
Designing A PC Work-
station Board
Around The TMS-3410
96 pages

ISSUE #40 (3/88)

The Great C Issue
11 C Compilers
Writing A Simple Parser In C
C++, An Object Oriented C
Source Level Debugger For
Turbo C
96 pages

ISSUE #41 (5/88)

Artificial Intelligence
3-D Graphics
Neural Networks
Logic Of Programming
Languages
Applying Information Theory
96 pages

**To Order:**

Phone: 1-800-888-8087

Mail: PO Box 223

Bend, Oregon 97709

United States,

Issues #1-34 \$3.00 each ppd.

Issues #35-current \$3.95 each ppd.

Canada, & Mexico

All issues \$5.00 each ppd.

Foreign (air mail)

All Issues \$7.00 each ppd.

Advertisers Index

Issue 42

Reader Page
Service Number

72	Acquired Intelligence	71
4	Austin Codeworks	55
5	Blaise Computing	05
1	Borland Int'l	Back Cover
87	Brown Bag Software	49,65,85
**	Capital Software	21
15	Cascade Electronics	76
**	CC Software	65
71	Complete Logic Systems	11
7	CompuView	45
90	Dair Computer Systems	85
8	Datadesk Int'l	01
32	Digital Research Comp.	49,78,82
6	Dreamtech	7
09	Ecosoft Inc	62
38	E.M. Enterprise	66
10	Emerald Microware	29
93	Erac Company	59

11	Halted Specialties	31
22	Integrand	23
88	Koala Computers	47
12	Logitech Inc	13
17	Manx Software Systems	79
42	McTek Systems	53
35	Merlin Publishing	68
**	Micro Cornucopia	39,90,91
36	Microcomputer Systems Consultants	81
**	MicroMethods	68
37	Microprocessors Unlmtd	69
24	microSOLUTIONS	38
2	Microsphere	2
59	National Advancement Corp	16

85	Oregon Software	81
03	PC Tech	Inside Front Cover
20	Peacock Systems	54
60	Scientific Software Solutions	82
27	Serengeti Software	69
**	Sharp's Inc	36
19	Slicer Control/Computer	80
40	Star-K Software Systems	52
101	United Products	61
62	V Communications	88
14	WindowDOS Associates	63
99	Windsor Technologies	17
39	Xenosoft	36
70	Zortech, Inc.	Inside Back Cover

**Contact Advertiser Directly.



Tell the world! Wear the Micro C T-shirt.

Choose a blue shirt with the classic Micro C Mutt
or a silver-grey shirt with the Micro C logo.

\$8.95 each ppd.

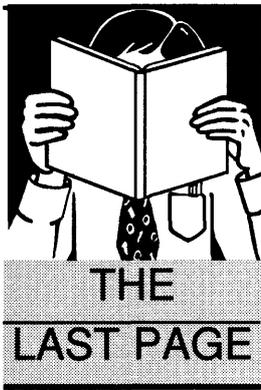
\$10.95 each ppd. - Canadian & Foreign orders

Available in S, M, L, or XL

MICRO CORNUCOPIA

P.O. Box 223
Bend, Oregon 97709

1-800-888-8087



Short Takes

C Power Users Guide

By Gary Entsminger
1912 Haussler Dr.
Davis, CA 95616

A C book, the Turbo Prolog Users Group, and a hint at puzzle results. Gary obviously has trouble sticking to one subject.

It seems C threatens to finally unseat Pascal as the hacker's language of choice. Former assembly language programmers who jumped to Turbo Pascal to check out the fuss no doubt think C is a step BACK in the right direction.

Consequently, C also seems to be the book publishers language of choice. I see more books to let us in on the tricks of C than most anything else.

Most, unfortunately, aren't that interesting.

Recently, though, I stumbled over one (*C Power User's Guide*, by Herbert Schildt) that takes on some dear subjects —

- pop-up and pulldown menus
- pop-up windows
- terminate and stay resident pop-ups
- language interpreters
- the mouse interface
- graphics routines
- video games
- poor man's networks

The graphics routines include functions for saving and restoring graphics screens.

The language interpreter routines include an expression parser and a small BASIC interpreter.

And in one chapter, after detailing the basics of serial port operation, Schildt goes on to develop a file transfer program which performs software handshaking.

Schildt's description (in C pseudocode) is clear, concise:

```
send()
{
    while (there are bytes to send) {
        send (a byte);
        wait();
    }
}
```

```
receive()
{
    do {
        receive_byte();
        send(acknowledgment);
    } while (there are still bytes to
        read);
}
```

"In this way, the transmitter never overruns the receiver, no matter how much the two computers differ in speed of operation. The only drawback to this type of handshaking is that it effectively halves the transmission rate since two bytes must be transmitted for each byte of information..."

The discussions are good (gentle but illuminating); the code detailed and intended to compile in the Turbo C and Microsoft C systems. And the subjects are at the heart of serious programming.

I've seen half a dozen other books by Schildt (*Advanced C*, *Advanced Turbo Prolog*, etc.), but this one has the most bells and whistles for serious hacking and programming.

**\$22.95 (pap). Published by —
Osborne McGraw Hill
2600 Tenth St.
Berkeley, CA 94710**

Turbo Prolog User's Group

If you don't know already, Turbo Prolog has an official user's group and an official newsletter.

Mike Floyd (technical editor of *Turbo Technix*), Dan Kerman (Borland R&D), and Kelly Rich (Borland technical support) are the organizers. The newsletter is chock full of programming ideas and tidbits, and the group has a BBS, user's disks, and provides a good platform for exchanging Turbo information.

For more information, contact —
**TPro Users
3109 Scotts Valley Drive, #138
Scotts Valley, CA 95066**



C++

FROM ZORTECH INC.

The change to a pure language

Now, C programmers can move over to C++ with Zortech C++ – the world's first 'true' C++ compiler for MS-DOS machines.

Zortech C++ is a 'true' compiler and fully conforms to Bjarne Stroustrup's specification as outlined in his book 'The C++ Programming Language'.

Previous implementations of C++ were actually 'translators' – only able to translate C++ source code into C. Of course, this was unacceptable due to the long translating and compiling times.

Now, C++ comes of age with the introduction of the world's first true C++ compiler – from Zortech!

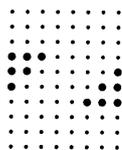
■ **Object Oriented Programming**
C++ is to C what Modula 2 is to Pascal. C++ brings 'classes' to C, so you can create separate modules that contain their own data and data-related operations. These 'classes' then become new types that can in turn be used to create further modules – this allows you to practically create your own language.

■ **ANSI C Superset**
You don't have to throw away your existing C programs – C++ is a superset of ANSI C. Now, you can take your Microsoft C or Turbo C compatible programs and easily migrate to C++ to take full advantage of the new C++ features.

■ **'Codeview' Compatible**
Zortech C++ is compatible with 'Codeview' – Microsoft's industry standard source code debugger.

■ **Improved Program Structure**
As stated in 'The C++ Programming Language', by using C++ "It would not be unreasonable for a single person to cope with 25,000 lines of code!"

■ **Other benefits**
Here's just a few: Operator overloading, overloading function names, default arguments to functions and better type checking.



ZORTECH
BOSTON LONDON FRANKFURT GENEVA

YES!
Rush me
C++ as shown
below:

Zortech C++ C++ Book
\$99.95 \$29.95
VISA/MC/COD/CHECK ACCEPTED

Name.....
Address.....
Phone.....

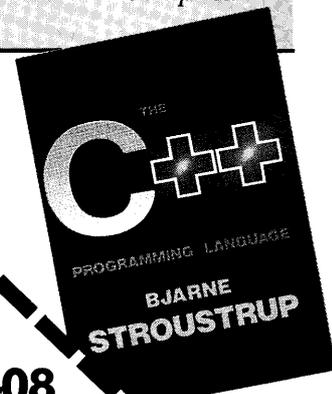
VISA or MC..... Exp. Date.....

To: ZORTECH INC. 361 Massachusetts Ave., Arlington, MA 02174.
Tel: 617-646-6703. Fax: 617-648-0603.

CALL THE ORDER HOTLINE 1-800-848-8408

Reader Service Number 70

ESSENTIAL READING!
This 325 page book 'The C++ Programming Language' by Bjarne Stroustrup contains the original definition of C++. All the examples shown in this book have been successfully compiled and executed with the Zortech C++ Compiler.

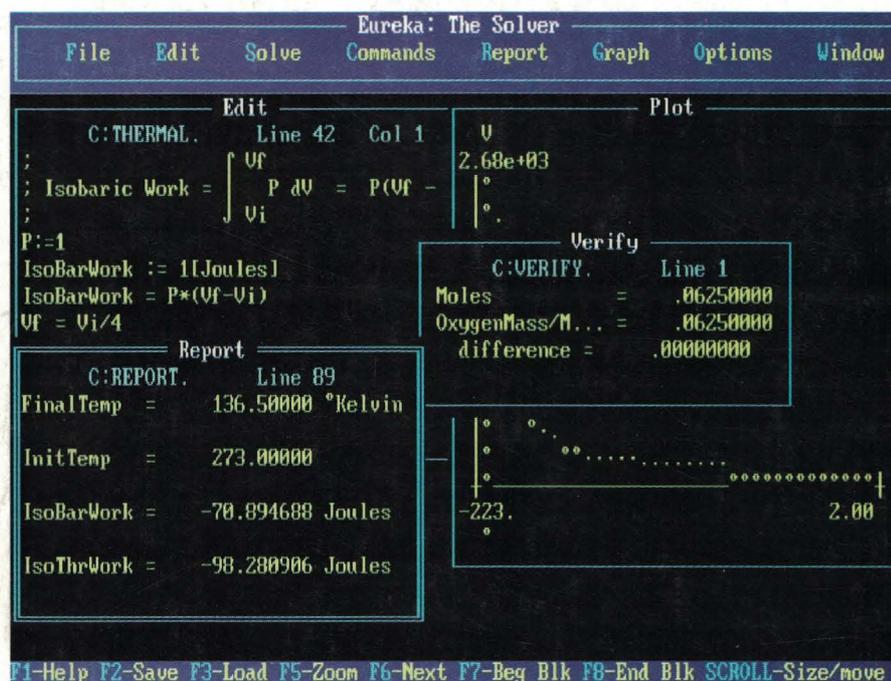


Eureka: Instant Answers!

Now solving scientific and engineering equations goes from agonizing to easy!

Eureka™: The Solver and your PC can handle any problem you can hand them—instantly. Just type in any linear or non-linear equation, select "Solve," and look at your answer. But that's not all, because Eureka can do a lot more. Like evaluate your solution and plot a graph. Or generate a complete report, sending the output to your screen, disk file or printer.

You can check the equation itself or ask Eureka to continue searching iteratively for a satisfactory solution. Or, change the active variables or constants and search for an entirely different solution. Using Eureka's windows, you can solve up to 20 equations at once. It's all done with pull-down menus, full help screens and absolutely no agony. And all for just \$167.



Eureka instantly solved this Physics equation by immediately calculating how much work is required to compress isobarically 2 grams of oxygen initially at STP to 1/2 its original volume. In Science, Engineering, Finance and any application involving equations, Eureka gives you the right answer, right now!

Easy as a Calculator . . . Powerful as a Mainframe!

- Pull-down menus
- Full-screen text editor
- Context sensitive help
- On-screen calculator
- Automatic 8087 math coprocessor support
- Inequality constraints
- Powerful built-in and user-defined math and financial functions
- Report generation complete with plots and lists
- Polynomial root finder

“ Merely difficult problems Eureka solved virtually instantaneously; the almost impossible took a few seconds.
Stephen Randy Davis, PC Magazine **”**



For the dealer nearest you
or to order call

(800) 543-7543

60-Day Money-back Guarantee*

System Requirements: For the IBM PS/2™ and the IBM® and Compaq® families of personal computers and all 100% compatibles. PC-DOS (MS-DOS) 2.0 and later. 384K.

*Customer satisfaction is our main concern; if within 60 days of purchase this product does not perform in accordance with our claims, call our customer service department, and we will arrange a refund.
All Borland products are trademarks or registered trademarks of Borland International, Inc. Other brand and product names are trademarks or registered trademarks of their respective holders.
Copyright ©1987 Borland International, Inc.

BORLAND