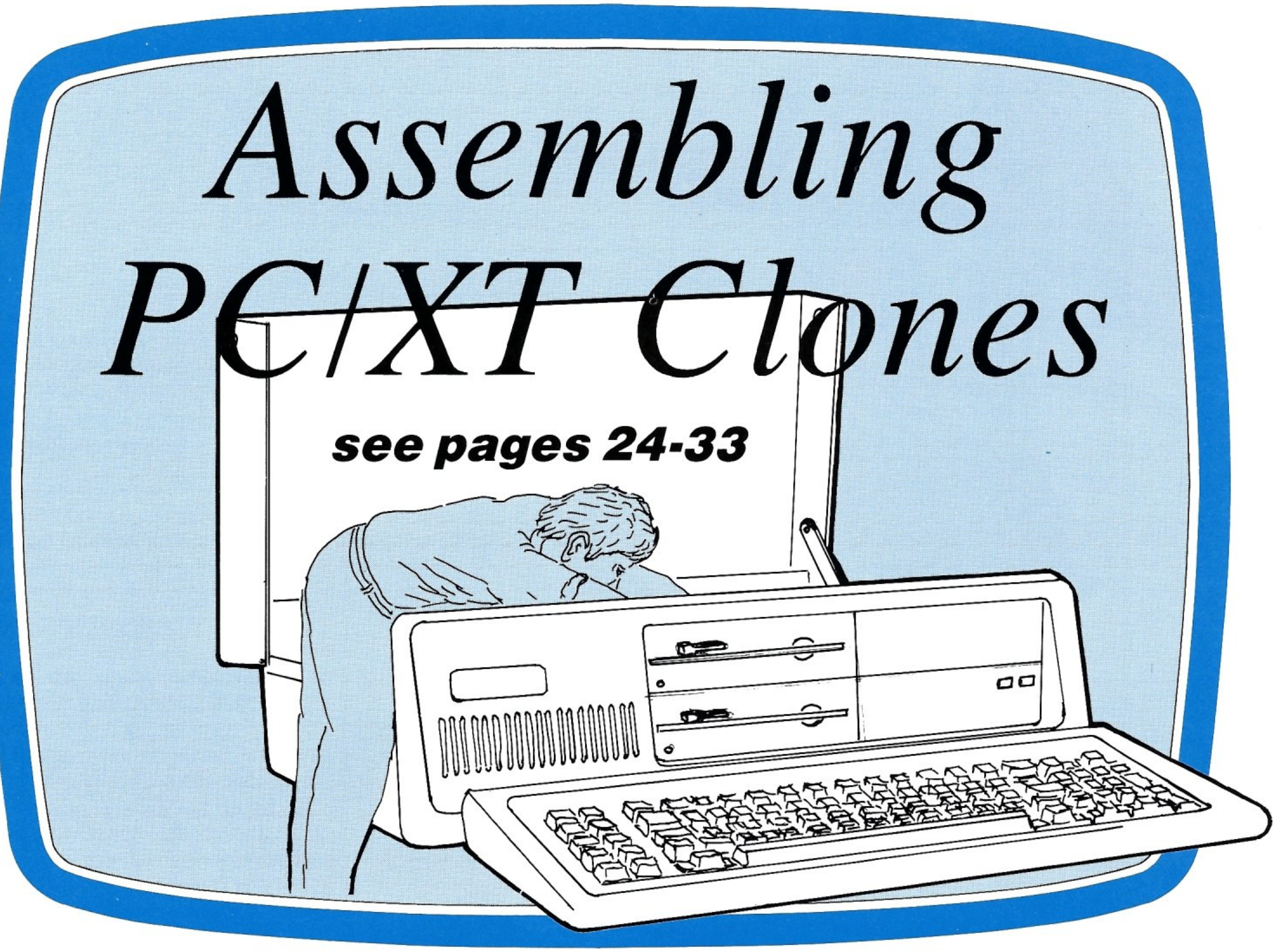
For the Advanced Computer User

Micro/Systems Journal™



Also in this Issue

Complete Table of Contents on Page 3

November/December 1986
Vol. 2 No. 6

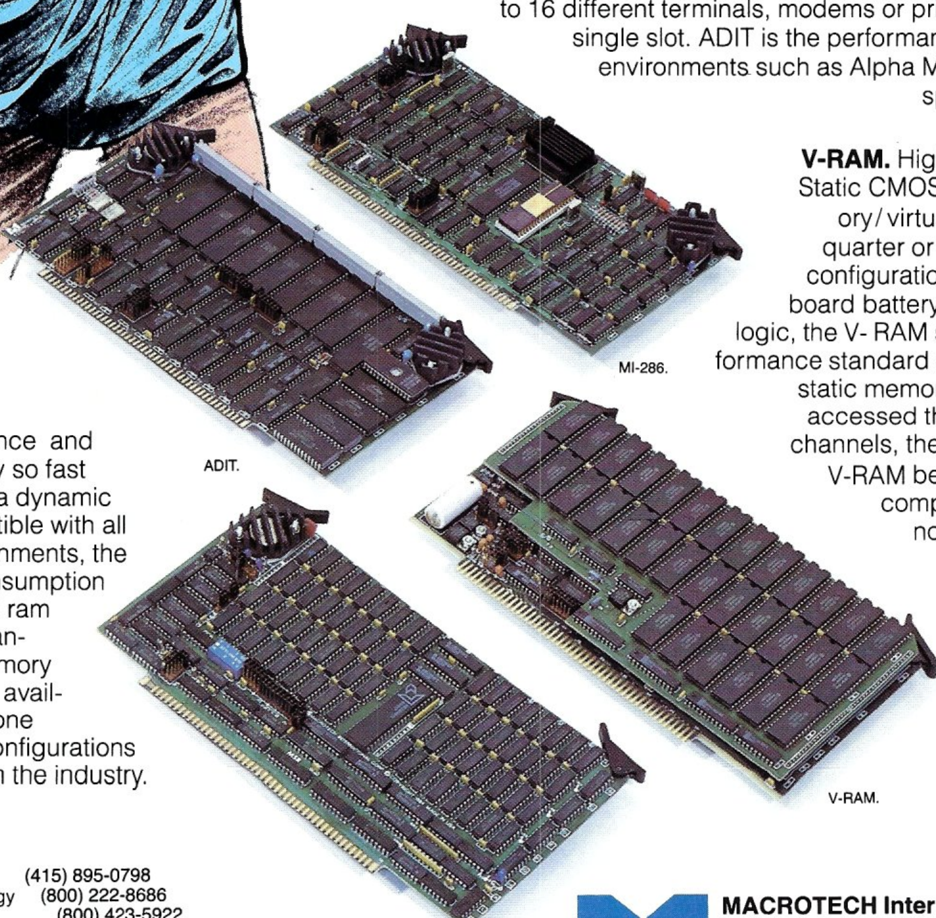
$3.95
U.S.A.

MACROTECH—*STILL THE S-100 PERFORMANCE PACESETTER*

MI-286. Our 80286/Z80H Dual CPU Board is at least twice as fast as Compupro's 8085/88 and it's a direct replacement. The MI-286 has already become the standard by which other 80286 based systems are measured. Ask us for a complimentary Benchmark Report.

ADIT. There's nothing else like it on the market. It's an Intelligent I/O Board with its own real time firmware that lets you control up to 16 different terminals, modems or printers all from a single slot. ADIT is the performance standard in environments such as Alpha Micro where I/O speed is critical.

V-RAM. High performance Static CMOS system memory/virtual disk in either quarter or half megabyte configurations. With its on-board battery and power-fail logic, the V- RAM sets a new performance standard at conventional static memory prices. When accessed through I/O port channels, the half megabyte V-RAM becomes M Drive compatible with true non-volatile solid-state disk capability.


MI-286.
ADIT.
V-RAM.
MSR.

MSR. High performance and reliability in a memory so fast you won't believe it's a dynamic ram product. Compatible with all popular S-100 environments, the MSR's low power consumption and 120 nanosecond ram devices set a new standard for dynamic memory products. The MSR is available in quarter, half, one and two megabyte configurations at the lowest prices in the industry.

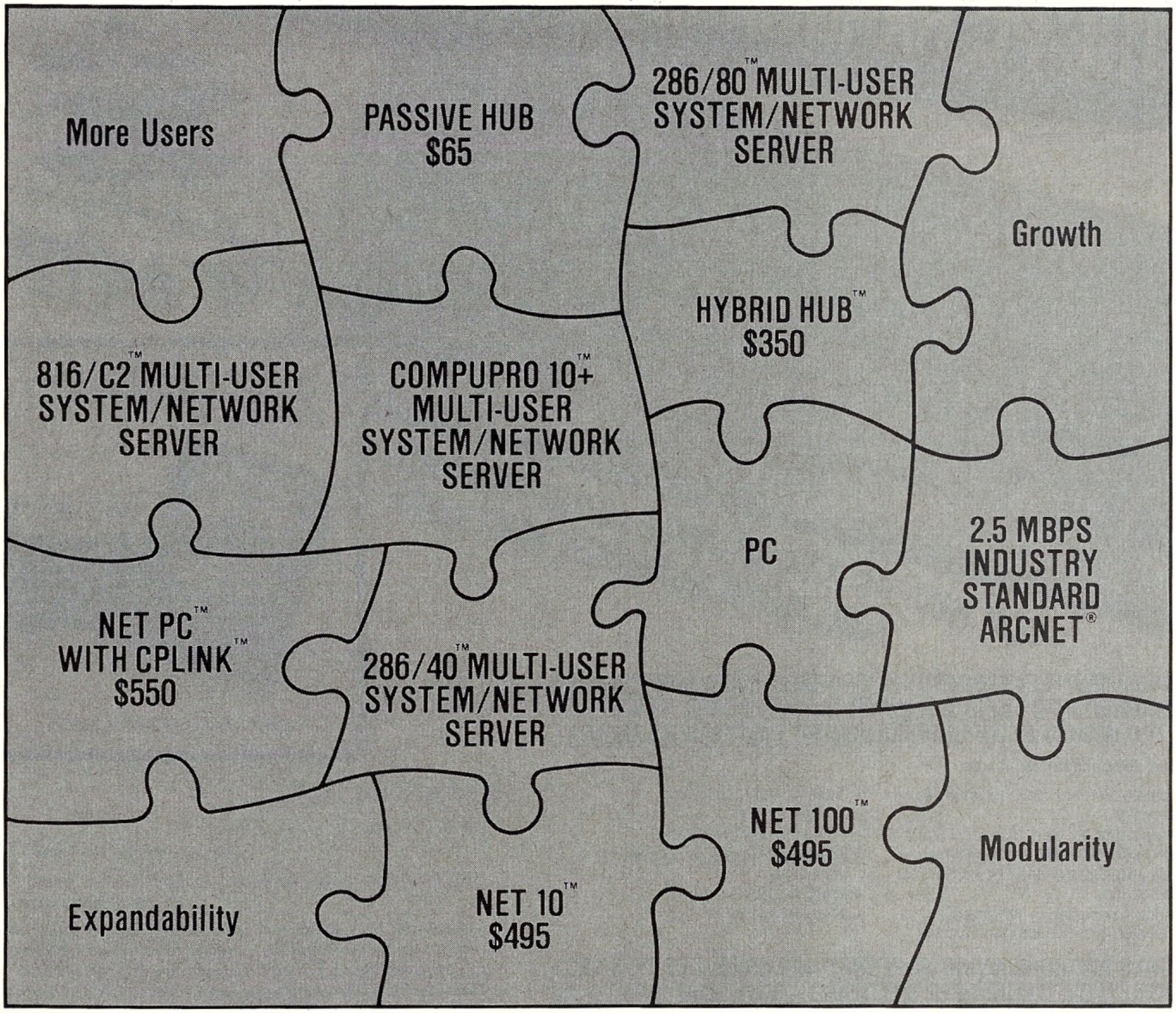
Dealers:

Gifford Computer Systems (415) 895-0798
Custom Computer Technology (800) 222-8686
S-100 (800) 423-5922
John D. Owens & Associates (212)448-6298
In England; Fulcrum (Europe) Ltd. (0621) 828763

Macrotech dealers also include most Compupro Systems Centers, Heathkit Electronic Centers and Alpha Micro Dealers.

MACROTECH International Corp.
21018 Osborne Street
Canoga Park, CA 91304
(800) 824-3181 • in Calif. (818) 700-1501
Telex: 9109970653

CompuPro Solves the Network Puzzle



Don't be puzzled by networking. CompuPro puts all the pieces together—our high performance multi-user systems simultaneously double as network servers for IBM® PCs. Move up to a true multi-user system and save your investment in existing PCs. To find out how CompuPro can solve your networking puzzle, call today for the name of your nearest dealer.

CompuPro™

Viasyn Corporation
26538 Danti Court, Hayward, CA 94545-3999 (415) 786-0909
TWX: 510-100-3288 Easylink Mailbox: 62877579

Suggested list prices, subject to change without notice.

Trademarks: CompuPro, 286/80, 286/40, 10 Plus, 816/C2, NET 100, NET 10, NET PC, HYBRID HUB: Viasyn Corporation. IBM: International Business Machines Corporation. ARCNET: Datapoint Corporation. CPLINK: ComputerNet A/S.

RISK FREE
30-DAY MONEY-BACK GUARANTEE

NOW FOR YOUR IBM PC, XT, AT, AT&T, PC Jr. AND COMPATIBLES *

"THE PC HARDWARE BARGAIN OF THE YEAR"

Chuck Humble — Computer Editor

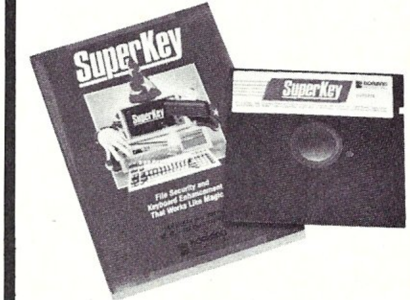
Both Hardware & Software

~~$129.95~~

Introductory Offer $99.95

OUR NEW PC 8700 AT-STYLE KEYBOARD AND BORLAND'S SUPERKEY® SOFTWARE

INCLUDES BORLAND'S *SUPERKEY* AWARDED "BEST OF 1985" BY PC MAGAZINE



RETAILS ALONE FOR $69.95

THE PC 8700 *AT-STYLE KEYBOARD* INCORPORATES EVERY IMPROVEMENT IBM DESIGNED INTO THEIR ADVANCED "AT" KEYBOARD. QUALITY SO GOOD, IT'S BACKED BY AN UNPRECEDENTED <u>TWO-YEAR WARRANTY</u>!

Advanced design features include:

- Separate Numeric Keypad simplifies data entry
- Status Lights for Cap, Num and Scroll Lock helps eliminate typing errors
- Extra-wide Control and Return Keys for ease of use
- Return Key placed in "standard typewriter" position for touch typists
- Selectric layout for Office Standardization
- Positive Tactile Feedback
- Keyswitches rated for 50 million keystrokes
- Made in USA
- 2-Year Warranty

BORLAND'S AWARD-WINNING *SUPERKEY* INCLUDED! GET THIS BEST-SELLING KEYBOARD ENHANCER SOFTWARE — VALUED AT $69.95 — AS PART OF THIS AMAZING OFFER AND CAPTURE THE POWER OF MACROS!

SuperKey's macro capability is remarkable. It literally turns a thousand keystrokes into one.

- Exclusive screen data Cut & Paste
- Screen Privacy and Protection
- Time & Data Stamping
- ProKey Compatibility
- Data Encryption
- Keyboard Lock
- Keyboard layout customization
- Integration with SideKick
- Pull-Down Menus
- On-Line Help and much, much more!

ORDER NOW — CALL TOLL FREE 800-826-5398; IN CA 800-592-9602

PLEASE RUSH ME __________ KEYBOARD/SUPERKEY BUNDLES AT $99.95 EACH. PLUS $10 SHIPPING AND HANDLING IN THE U.S.A.*

☐ ENCLOSED IS MY CHECK FOR $109.99 FOR EACH BUNDLE ($116.45 FOR CA RES. — SALES TAX INCL.)

☐ VISA ☐ MC ☐ NUMBER ______________________

EXP. DATE ________ SIGNATURE ______________________

NAME ______________________

ADDRESS ______________________

CITY ______________ STATE ______________ ZIP ________

TELEPHONE ______________ COMPUTER TYPE & MODEL ______________

* PC Jr., AT&T and Corona computers require adaptors available at a nominal fee. Call for pricing.

CRITICS' CHOICE

"Who Can Pass Up a Deal Dept?"
John C. Dvorak, Infoworld, 3-10-86

"The best part of the keyboard is the feel. It is ideal!"
PC Productivity Digest, 5-86

"The keyboard's construction is solid . . . also quick and responsive to the touch. It is ideal for anyone who wants to or has to replace their PC keyboard . . . At $129.95 DataDesk's keyboard is a bargain. The PC 8700 keyboard with SuperKey for that price is a steal."
Computer Shop Talk, 3-3-86

"DataDesk International has designed a sturdy and handsome keyboard that has a tactile response . . . and bundled with SuperKey is the PC hardware bargain of the year."
Charles Humble—Computer Editor, The Oregonian

"DataDesk's is genuinely a nice keyboard . . . It feels nice . . . I can type much faster and much more accurately on it than most others . . . and it's inexpensive. It costs only $129, compared to around $300 for the Keytronics 5151."
Harry Newton, Publisher, Teleconnect, 5-86

TELECONNECT
TEST DRIVE SCORECARD

The DataDesk PC8700 Keyboard

TECHNICAL PERFORMANCE	10
EASE OF USE	10
DOCUMENTATION	10
OVERALL RATING	10

TELECONNECT'S CHOICE

NECT
RIVE
CARD

The KeyTronics 5151 Keyboard

TECHNICAL PERFORMANCE	9
EASE OF USE	9
DOCUMENTATION	9
OVERALL RATING	9

datadesk
INTERNATIONAL
7650 Haskell Ave.
Van Nuys, CA 91406
818-780-1673

For the Advanced Computer User

Micro/Systems Journal™

November/December 1986
Vol. 2 No. 6

Micro/Systems Journal (ISSN #8750-9482) is published bimonthly by M&T Publishing, Inc., 501 Galveston Dr., Redwood City, CA 94063; (415) 366-3600. Second-class postage paid at Redwood City and at additional entry points.

Article Submission: Please write the editors at Micro/Systems Journal, Box 1192, Mountainside, NJ 07092 or call (201) 522-9347 to see if there is interest in a particular subject. Please do not send the article unless we ask for it.

If you are interested in reviewing hardware or software please write telling us your interests, your background, and include a sample of your writing.

Send a stamped self-addressed business size envelope (to NJ address) for a copy of our Author's Guide.

Address Correction requested: POSTMASTER: send address changes to Micro/Systems Journal, 501 Galveston Dr., Redwood City, CA 94063. ISSN 8750-9482

Change of Address: Please send old label and new address to: Micro/Systems Journal, 501 Galveston Dr., Redwood City, CA 94063.

Correspondence: Editorial Office: Box 1192, Mountainside, NJ 07092; (201) 522-9347. The editor may also be contacted via MCI Mail (SLIBES). Business Office: 501 Galveston Dr., Redwood City, CA 94063; (415) 366-3600.

Advertising Rates: Available on request. Call (415) 366-3600 or write to: Micro/Systems Journal, 501 Galveston Dr., Redwood City, CA 94063.

Subscription rates: U.S.: $20 for one year, $35 for two years. Canada & Mexico: $28 for one year, $52 for two years. Europe & So. America: $35 for one year, $64 for two years. Asia, Africa, Australia & Near East: $39 for one year, $70 for two years.

IN THIS ISSUE

FEATURE ARTICLES

PRODUCT REVIEWS

DEPARTMENTS

Editor's Page

by Lennie & Sol Libes

We have good news for you, and for us. We have finally found a publisher for Micro/Systems Journal.

About a year ago, we mentioned, in this column that we were looking for someone who could publish M/SJ while we would continue to edit the magazine.

Susan, Don and the two of us were overloaded and barely able to handle the current operation, much less expand it. During the almost two years that we published the magazine none of us have been able to take a vacation.

As many of you know Micro/Systems Journal is our second job. Three of us are teachers, and now that the summer is over are back in our classrooms. Sol is a Professor of Electrical Engineering at Union County College. Lennie is a Professor of Mathematics at County College of Morris. Susan is a Professor of Chemistry and Marine Science at the University of South Carolina. Don, the non-teaching member of the family, is a Computer Scientist at the National Bureau of Standards, developing software for the next generation of advanced robotic systems.

None of us wanted to give up our regular jobs and become full-time publishers. We all like our current jobs. We really do not care to be "in business", this is just not our thing. We just like to communicate with people about advanced topics in microcomputers. We like to learn more about the innards of these machines and how to control and apply them. And, in the process to teach others.

Publishing a magazine means that most of one's time is spent in publishing activities: servicing subscribers and advertisers, dealing with printers, typesetters, and the post office (this alone can drive one up the wall), managing subscriber lists, production problems, dealers, etc. The fact is that we had to spend much more time on publishing than on editing the magazine. Often, the editing part got the short end in an effort to get the issue out on time. Keep in mind that we were dealing with over 11,000 readers, typically a hundred advertisers per issue and a large number of newsstand dealers. And, we ran all of this out of our home, in our "spare" time.

A Report To The Reader

It was clear to us that the "magazine" had taken total control of our lives, and we had to find a solution.

It took a year for a us to find the "right" company. We did not want subscribers to have another Ziff-Davis experience. We had many offers for the magazine and selected M & T Publishing. M & T also publishes "Dr Dobbs Journal" and "Business Software" magazines, as well as books and software. They have the staff and experience to handle M/SJ. Further, they are able to provide the support staff and will enable M/SJ to expand and improve its editorial content. On the other hand, they are not so large that M/SJ will get slighted.

No longer will readers who call for information, or to remedy a problem, have to talk to an answering machine. Readers will be able to pay for subscriptions via credit card (this was always a big problem for foreign and Canadian subscribers). Readers who buy their copy on newsstands should find it available in many more outlets. These are some of the things that both readers and advertisers have been pressuring us to do. These things will be immediately available, and in the future, even more advantages will be coming your way with this change.

THE FUTURE

Our relationship with M & T will ensure the future of M/SJ's editorial independence. We will continue our editorial direction, providing information on the cutting edge of technology for advanced microcomputer users. Upcoming issues will be covering topics such as:

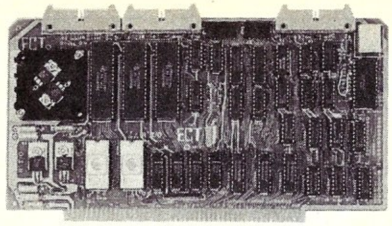
Reviews of 9600 Baud Modems
Interfacing To Microsoft Windows
multi-processing On The PC
Unix On The PC
80286 Protected Mode Programming
80386 Programming
MS-DOS V5.0
Network System Installation
Servicing PC/XT Machines
An S-100 Product Directory
High Resolution Display Systems

We are sure all of these topics will be of interest to people who do software and hardware design, system integration, installation and support.

We will continue our regular columns. There will be "News and Views" (also known as Sol's "gossip column"). And we will continue to publicize the latest in public domain software from the two leading authorities, Hank Kee and Steve Leon (who between them have created over 500 PC/Blue and SIG/M disks). And don't forget columns such as Don's "C Forum", Randy Davis' "Turbo Pascal Corner", Al Cameron's "Scientific User", Ian Darwin's "Unix File" and Bob Blum's "CP/M Bus", all authorities in their respective fields.

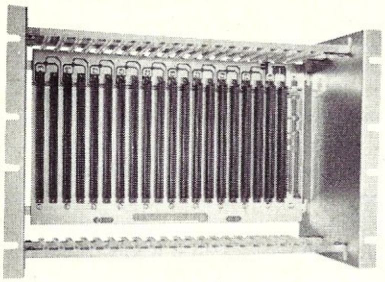
We are eager to hear from readers. Please let us know what you think of what we are doing and make suggestions on how we can improve. If you would like to write for M/SJ (we do pay for articles), ask for a copy of our author's guide. You can contact Sol via MCI mail (SLIBES), or write (M/SJ, Box 1192, Mountainside NJ 07092) or call (201-522-9347). µ

CUSTOM PRODUCTS
DESIGN • LAYOUT MANUFACTURING



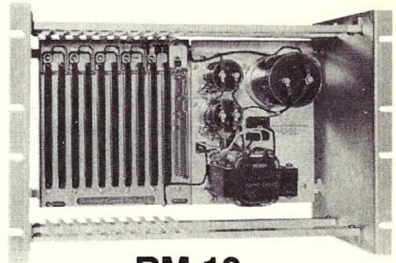
R ²I/O
ROM/RAM & I/O

S-100 PRODUCTS

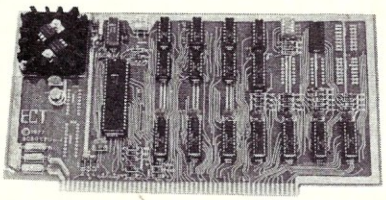


ECT-100-F
RACKMOUNT CARD CAGES

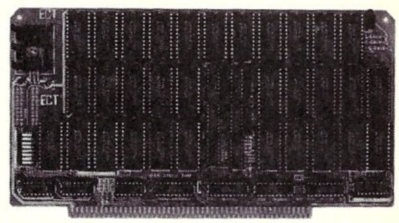
ECT™



RM-10
CARD CAGE & POWER SUPPLY



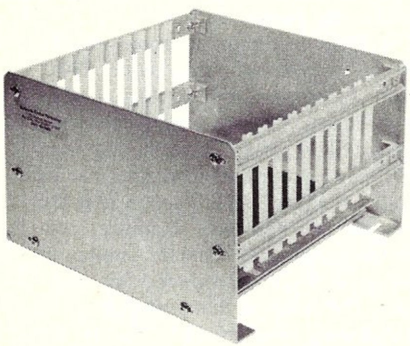
8080 CPU
CENTRAL PROCESSING UNITS



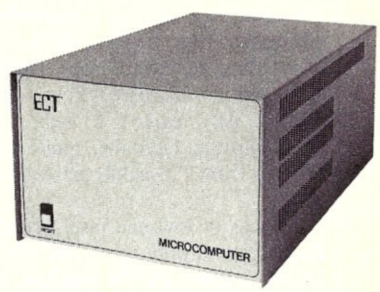
64K RAM
FULLY STATIC MEMORY

BUILDING BLOCKS FOR MICROCOMPUTER SYSTEMS, DEDICATED CONTROLLERS AND TEST EQUIPMENT

CARD CAGES, POWER SUPPLIES MAINFRAMES, CPU'S, MEMORY I/O, OEM VARIATIONS



CCMB-10-F MIN
6,10 OR 20 SLOT CARD CAGES

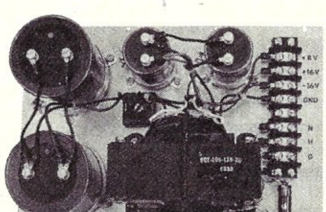


TT-10
TABLE TOP MAINFRAMES

ELECTRONIC CONTROL TECHNOLOGY, INC.

10 Cottage St., Berkeley Heights, NJ 07922 (201) 464-8086

SPECIALIZING IN QUALITY MICRO COMPUTER HARDWARE



PS-30 A
POWER SUPPLIES

MULTIBUS® PRODUCTS

MULTIBUS IS A TRADEMARK OF INTEL CORP.

News, Views & Gossip

by Sol Libes

RUMORS & GOSSIP

HiTech International Inc. (Sunnyvale CA) has cut the price of their AT compatible to $995. The system includes 640K of RAM, a 1.2 Mbyte floppy, I/O ports, and color graphics adapter. As far as I am aware, this is a new low price for an AT clone.

Microsoft is rumored rushing MS-DOS Version 5.0 out the door to head off introduction of 386-based systems from competitors. I expect it to be demoed at Comdex in November, and released early next year, when IBM is expected to introduce a new version of the AT (see story below). Version 5 should be optimized for the 286, use the protected mode, address up to 16 Mbytes of memory and provide multitasking. This should relegate 8088-based systems to the home and very low-end marketplaces.

Manufacturers of EGA-compatible display controller cards for the PC have discovered they can easily increase the resolution of these displays by increasing their clock speed, making some minor hardware changes and adding a software driver. Multi-sync type displays will automatically adapt to the increased speed. Thus, the resolution can be increased from 640 x 350 to 640 x 480 color pixels. And the monochrome Hercules mode can be increased from 720 x 348 pixels to 752 x 410 pixels.

Prices of low-end laser printers should finally start dropping as IBM is expected to shortly introduce a low-end laser unit based on a Ricoh engine. Also expected soon are color laser printers and laser printers that can emulate the HP and Apple laser printers, as well as Epson and Diablo printers. Also expected are miniaturized laser printers small enough to fit in a briefcase.

Ashton-Tate is promising a new version of dBase-III for early next year that will incorporate a high-level querying method. Called Query-By-Example (QBE), it enhances the users ability to search and retrieve data. Ansa Software's Paradox program already includes QBE. A user will no longer have to tell the database manager how to do the retrieval, merely what is to be retrieved.

This should be the last upgrade to dBase-III. Future generations are expected to run only on 286- and 386-based systems and be based on IBM's Structured Query Language (SQL). Ashton-Tate also announced that it is dropping copy protection from all its products.....that leaves Lotus as the only major software company using copy protection.

Sony has demoed a new 20″ color CRT display with 2048 x 2048 pixel resolution. Currently Sony's best display has a resolution of 1280 x 1024 pixels. No word yet on availability.

The Defense Advanced Research Projects Agency (DARPA) has awarded a contract to **BBN Advanced Computers, Inc.** (Cambridge MA) to develop an 8000-processor computer system.

A basic PC clone, with keyboard, and one floppy drive now typically sells for under $700. However, in Taiwan the typical price is about $350, indicating that there is still room for further price decreases. Some people are predicting that the price for a basic system may drop to as low as $300 in the near future.

Apple Computer is currently showing a prototype of its new Macintosh to selected customers. It includes a 68020 micrprocessor, memory management chip, a math coprocessor socket, 1Meg of RAM (expandable to 8Mbytes on board) and six expansion slots. Rumor is that the base unit will sell for $4,000.

AT&T is boasting that, as of the end of 1985, there were about 200,000 installations of Unix and that they expected this to double by the end of '86. One of the reasons is that AT&T has reduced the binary licensing fee to only $60 for a one or two user system. Three or more users raises the fee to only $150. There is no doubt that AT&T is trying to encourage the use of Unix for small multi-user applications.

Motorola is promising to ship samples of the 68030 micrprocessor next summer with production quantities to be available before the end of the year. They are boasting that the device will provide twice the performance of the 68020. The 128-pin chip will have a data cache, memory manager as well as other enhancements. Initial production will run at 16Mhz with later chips expected to run at up to 30Mhz.

IBM RUMORS

There are rumors that IBM has ceased manufacturing the PC and PC/XT and is clearing out stock (rumored to be worth $1 billion) preliminary to introducing new versions of the PC and XT. The new machines are intended for the school/home market and to compete with low-cost clones. IBM is currently testing several different versions of these new low-cost PC systems with key software developers. Prototypes all use an 8Mhz 8086, new gate-array surface-mount chips to reduce chip count, as much as 1 Mbyte of memory and 3½″ floppy drives. Some versions have 4 expansion slots and built-in EGA controllers. They will be assembled on a new highly automated production line in Austin,Texas and may contain proprietary patented features that will be difficult for the clone makers to copy.

IBM has also signed a contract with Intel to develop semi-custom ICs. It is expected that this will include a customized version of Intel microprocesors and other proprietary circuits.

There are also rumors that early next year IBM will introduce a new version of the AT and a 386-based system that is graphics-based. These machines are expected to run at fast clock speeds and include features that further enhance performance. The 386-based system is expected to have a new bus architecture that will emulate the old PC bus and also provide a full 32-bit wide data path.

The general reaction to IBM's token ring networking system is that it is too slow. Thus, IBM has disclosed that it is working on a 16Mbit/sec version, that is four times faster than the current version.

The foregoing rumors are apparently the result of intentional *leaks* from IBM. Although the systems may be announced as early as January, they are not expected to be shipped until the spring. As such, it appears that IBM is resorting to marketing tactics that they have successfully employed in the mainframe marketplace. Namely, pre-announcing products that are similar to products already introduced by competitors. The result is that customers hold off purchasing these products waiting for IBM to release their products.

The new XT-286 appears to be an interim product that IBM designed and got into production quickly, and at low cost, to compete against the AT clones. The unit has met with a ho-hum reaction in the marketplace and some dealers were offering discounts on the system within one week of introduction. Some incompatibilities in the product have already surfaced. Although the unit has five 16-bit expansion slots, the case (being a PC/XT case) is not high enough to accept AT plug-in cards. Also, the I/O bus timing is different and IBM's Token-Ring and Novells NetWare LAN systems and some expanded memory cards will not work in the unit. All of which serves to point out that even the letters "IBM" do not guarantee IBM-compatibility!

There are also reports that the IBM "Convertible" is turning in disappointing sales and may have as short a life as its predecessor, the IBM "Portable". In the meantime, IBM finally caved in and dropped the proprietary command set from their modem in favor of the Hayes command set. This may be a harbinger of things to come; if IBM tries to go the proprietary route again on new PC/XT/AT products, the market may force them to be compatible.

The RT is also rumored not measuring up to IBM's sales expectations. Competitors in the engineering/scientific workstation market, such as Sun Microsystem, Apollo and DEC, have introduced new systems that outclass the RT in speed and graphics capability and have MS-DOS compatibility. The result is that the RT has garnered little software and peripheral support, further compounding IBM's inability to compete in this market. IBM has made some improvements to the RT and cut prices (over 30%) and provided incentives to software developers to support the product, in an effort to boster sales.

IBM, in a effort to bolster sagging sales, has moved 3,000 employees out from its headquarters into the field to sell. This is in addition to more than 2,000 additional salespeople hired earlier this year. Thus, this year, IBM has increased its field sales staff from 5,000 to 10,000.

MS-DOS VERSION 4.0

Microsoft has released version 4.0 of MS-DOS to OEMS in Europe and Australia. As yet, there is no word if, and when, it will be released in the U.S. The likelihood is that if IBM does not endorse it, it will not be released here.

Version 4.0 is a single user multitasking system for 8088, 8086 and 80286-based systems offering improved response for networking environments. It includes a preemptive time slice scheduler, overlapping I/O, interprocess communications via pipes, shared memory and signals, and intraprocess semaphore and process control primitives. The foreground partition emulates Version 3.2. However, programs running in the background, must be specially written for this purpose. Background programs communicate with the user via a *pop up* facility.

4.0's limitations are still the same as earlier versions in terms of maximum addressable memory and disk space....namely 640K of RAM and 32 Mbytes of disk space. Users will have to wait for version 5.0 for these to change.

Microsoft has released driver for MS-DOS 3.1 and 3.2 which allows users to operate CD-ROMS with MS-DOS. MS-DOS treats the CD-ROM as a disk drive with up to 550Mbytes of storage using the High Sierra group data file hierarchical protocol.

386 UPDATE

Microsoft is expected to shortly release a version of MS-DOS tailored for the 80286 that will allow users to run simultaneous 640Kbyte sessions. This is viewed as an interim version until Microsoft is able to develop V6.0, a multiuser multiprocessing version of DOS designed expressly for the 386. This new verion will utilize the virtual 8086 mode of the 386. However, delivery of the system is not expected until late next year. With Microsoft running late in shipping the 386 DOS several other companies are expected to get into the 386 operating system race. And, there is a possibility that IBM may release their own 386 operating system in an effort to better compete in that market.

In the meantime Microsoft will begin shipping a 386 version of Xenix in January. And **Phoenix Technolgies** is promising what it calls a *hypervisor* that will run MS-DOS applications under Unix on 386-based systems.

Corvus and **Compaq** are already shipping 386-based systems. The Corvus system, with 70Mbyte drive is priced at $12,795. File server versions are $16,595 (70Mbyte drive) and $19,795 (126Mbyte drive). The Compaq systems are $6,499 (40Mbyte drive) and $8,799 (130Mbyte drive). Compaq is running an enhanced version of MS-DOS 3.1 that integrates the Lotus/Intel/Microsoft Extended Memory Specification and provides some multitasking features.

Several mail order companies have already announced 386-based systems. **Computer Classified Inc,** Miami FL, has a $2,195 System and a $1,795 replacement motherboard for the XT and compatibles. **Advanced Logic Research** (Irvine CA) has announced three systems ranging in price from $3,990 to $6,494. **PC's Limited** (Austin TX) and **PC Designs** (Tulsa OK) are promising systems and plug-in 386 accelerators cards before year's-end. PC Designs has stated that their system will sell for less than $3,000, while PC's Limited is promising an under $5,000 price. **American Computer & Peripherals** (Santa Ana CA) is promising a card that replaces the 80286 on the AT motherboard.

The early systems will run with 16Mhz clock rates and offer processing speeds about three times that of an 8Mhz AT system. They are expected to use reworked versions of the companies' AT motherboards and hence work with many AT and PC plug-in cards,**Computer Dynamics Inc.** (Austin TX) has already announced a 24Mhz system priced at $3,995.

The 386 marketplace is expected to grow slowly. Not until a new operating system and applications software that exploit the features of 386 becomes available are 386-based systems expected to develop significant marketshare. This should take at least a year, and more likely 2 or 3 years.

IBM is expected to introduce their 386 system early next year. However, positioning this system in their product line should be a problem in that it would seriously impact sales of their RT product line. The system is expected to have proprietary features that forestall cloning. Many industry pundits expect IBM to introduce a proprietary 386 operating system and networking hardware to link to IBM mainframes

NEW CHIP SET SQUEEZES AT FURTHER

Faraday Electronics and Chips & Technology Inc. are giving IBM fits. C&T, which is only 18 months old, has introduced LSI-IC's which reduce the AT motherboard chip count by more than 30% and te EGA card chip count by more than 50%. As a result clone makers have been able to introduce EGA cards and AT systems that provide more features and performance on smaller PC real estate and at less cost.

The new C&T chip replaces 9 more of the AT chips and promises to reduce the size and cost of the AT even further. It combines the functions of DMA, interrupt control, clock/calendar, memory mapping and other functions.

C&T has also disclosed that it will soon begin sampling an IC chip set for 80386-based systems. The company predicts this will enable a basic 386 motherboard to be built with only 30 chips.

Faraday has announced that it will soon sample a 2-chip set for the AT that replaces 50 AT chips. To prove its point Faraday is making available a 9″ by 4″ AT-compatible evaluation motherboard to OEMs. By contrast, the IBM-AT motherboard is 12 ″ by 13″.

IBM is expected to shortly introduce new lower cost versions of the PC and AT, and an EGA card based on reduced component chip sets. You can bet they are fuming over how these upstarts have become the backbone of the clone industry.

GRAPHICS NEWS

Texas Instruments appears to be the first to start shipping samples of the new generation of graphics controller chips. These chips are really dedicated microprocessors that relieve the system processor of display functions. They promise to make low-cost, high pixel density, high-performance graphics systems available by late '87. The current generation of display controller chips are *hardwired* and typically rely on the system processor for control.

The TI chip (TMS34010) was announced in January and demoed in a PC system at the SIGGraph conference in August, by **Number Nine Computer Corp,** of Cambridge MA. **Intel** announced their 82876 chip in May but as yet has not demoed the chip or shipped samples. **Microsoft** has already announced that it will include drivers for the TI and Intel chips with several of their PC software products. **National Semiconductor, Advanced Micro Devices** and **Motorola** are also promising similar chips, with formal announcements and release of specs expected early next year.

The first products using these chips should be introduced by late '87. Software should follow in '88, making graphics capability, previously available on high-end workstations, available on PC/XT/AT-based systems.

In the meantime, the performance of high-end workstation graphics is also improving. Using dedicated 32-bit processor systems, manufacturers are moving to 2048 x 2048 pixel color displays. **Megascan Technology Inc.** (Gibsonia, PA) has even introduced a 4096 x 3278 19″ color display. Also, at the recent SIGGraph conference, **Metheus Corp.** (Hillsboro OR) demoed a real-time system which wrote to the screen at a rate of 160 million pixels/second.

HOME COMPUTER MARKET PICKING UP

The home computer market from late 1984 through early 1986 all but dried up spelling the end for several companies. **Commodore** managed to survive but is in very poor shape. Warner Communications gave up on **Atari** selling it to Jack Tramiel for next to nothing. Even **IBM** gave up with the PCjr. **Apple** now has a very low profile in the home market. And, the Japanese did not even bother to bring their MSX machines to the U.S.

Now the word is that sales of personal computers to home users is picking up again. The chief beneficaries are Atari with its ST systems and Commodore with its models 64 and 128. The Commodore Amiga still appears to be having trouble getting off the ground. More and more home users are buying IBM-PC clones which are now relatively close in price to the Atari and Commodore machines. Although games are still very popular for home systems, home users seem to be spending more time with wordprocessing, database managers and spreadsheets. It appears that the new generation of home users are more computer literate.

Atari also appears to be gaining a share of the European engineering market as evidenced by support products that have been released there. The most interesting is the K-Max from Kuma Computers Ltd, Pangbourne England. K-Max attaches to the Atari's ROM port and turns it into a parallel processor (containing two Inmos Ltd T414 Transputers) capable of up to 15 MIPS. The price is $2,175, an unheard of price for a true parallel processor system. Also, in Europe, Atari is selling versions of the ST with 2 and 4 Mbytes of RAM and a software package which emulates the DEC VT-100 and Tektronix 4010 terminals.

INTEL 80486 RUMORS

Intel is expected to make a formal release of the specifications for the 80486, their next generation microprocessor, as early as January 1988. They have not as yet committed the 486 to silicon but the rumors are flying fast and furious. Industry prognosticators are predicting that the chip will be optimized for AI applications. They expect the 80486 to be designed for knowledge-based systems and support of the Lisp and Prolog languages. They expect it to compete with AI processors planned by TI, Hitachi and others.

Intel currently has several advanced microprocessor development projects underway. For example, the 80930 project is attempting to implement parallel processing on a single chip. And Intel has resurrected the 432 in a joint study with Siemens, to develop a fault-tolerant processor. Also, there is the VL82C389 message-passing coprocessor project.

The expectation is that the 80486 will appear somewhere around 1990. In the meantime, Intel is beginning to ship 16Mhz 80386 chips and expects to soon offer 20Mhz versions. They claim that this high speed device will allow the building of 386 systems that perform four times faster than an IBM-RT or DEC MicroVax and twice as fast as a Sun III. (μ)

Quality S-100 Products

Now Available! FULCRUM'S New...

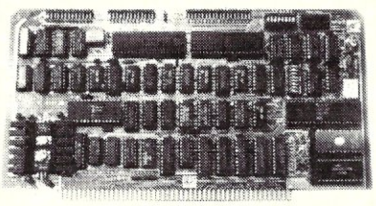
256K STATIC RAM

- 8/16 Bit Transfers
- 24 Bit Extended Addressing
- Battery Back-Up

$495

Designed For Speed & Flexibility.

MPUZ CPU

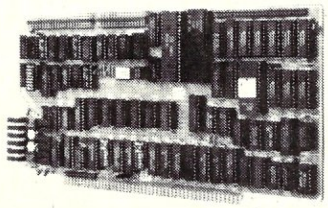


This NEW MPUZ CPU utilizes the Z-80 8MHz uP as a basis for its 8MHz CPU for S-100 systems, and has been carefully designed to meet the requirements of the IEEE - 696 standard. The quality and performance this CPU provides is rarely found in S-100 products, and you can see why... ***only $299***

- 4 or 8MHz clock rate
- Two RS-232 serial ports
- Centronics printer ports
- Real time clock with battery back-up
- Vectored interrups to any block location in memory
- Programmable timer
- ROM monitor
- Power on Jump
- On board wait states
- 2K of RAM space
- 24-bit extended addressing
- Latched Status
- Front panel compatibility
- MPM support

Best Value In Disk Controllers.

OMNIDISK



Now the FULCRUM OMNIDISK offers S-100 systems users a unique marriage of component compatability and technological innovation. These together produce features not found in any conventional disk controllers made today. See for yourself what tomorrow looks like... ***only $299***

- Simultaneous support of both 5¼" and 8" floppy disks and hard disks
- Complete 24 bit DMA
- Power on boot for 5¼" and 8" floppy and hard disks
- Power on boot PROM
- On board de-blocking to save RAM space over BIOS
- Interfaces with the WD 1001* hard disk controller
- Supports 13 devices simulatneously
- Full track buffer allows controller to recall entire track
- DMA'S at 10 MHz
- Supports MS DOS
- 10K on board buffer saves two K of TPA

So before you buy another S-100 component, call or write for our *FREE* catalog. And see how your system can benefit from the FULCRUM difference.

*CP/M *2.2 configured for OMNIDISK $60. *Trade mark of Digital Research. FREE U.P.S. ground shipping on prepaid orders. Shipping is added to VISA, M/C, and C.O.D. orders. CA residents, please add sales tax.

FULCRUM COMPUTER PRODUCTS

707/433-0202 459 Allan Court, Healdsburg, CA 95448.

Also in FULCRUM'S Family: OMNIRAM 64K memory board. Serial I/O 2-2 & Video I/O Interface boards. Relay board. I-8080, 8015 and 8035 main frames with 21 slot mother boards. CPA front panel and A/D board.

there is mail....

We welcome your letters with comments, compliments, criticism and suggestions. We read them all and publish the most noteworthy, even if they are critical of us. We do not have the staff to answer all letters personally. And all letters become the property of M/SJ and may be subject to editing. Further, we do not print letters that do not include a name and address.

Please send your letters to: Micro/Systems Journal, Box 1192, Mountainside NJ 07092.

A READER'S REACTION

Dear Sol,

I would like to make two comments on Leon Suchard's article "Building an AT Clone" (M/SJ, May/June '86):

1. An Intel 80287-3 is tested to 5 Mhz, not 8 as stated. For the record, it's 80287-3/6/8/10 for 5/6/8/10 Mhz respectively.

2. Regarding the V20 - you consider it a "worthwhile change" in running PC-DOS software. Perhaps it does give some speed improvements, if you can live with the uncertainties and exceptions, but have you considered that perhaps the most appropriate software to run on a V20 would be those illegal copies some unscrupulous people make of the other peoples' intellectual property?

Yours sincerely,
Pat O'Leary
Ardrew, Ireland

DOES SHAREWARE BELONG IN PD LIBRARIES?

Dear Lennie and Sol:

I continue to enjoy your journal and appreciate especially reading "Editor's Page." Since the departure of John C. Dvorak from *Infoworld*, "Editor's Page" (and "News, Views & Gossip," too) alone make my subscription to the journal worthwhile!

I have been sensing a deep sense of *frustration* by Steve Leon towards PC Public Domain programs. I have been wondering about it as I have had very satisfying experience with SIG/M Public Domain. I wonder why Steve feels a sense of even *indignation* towards some of PC public domain programs. A couple months ago, I got my own Zenith, an IBM PC/XT clone, and I began to acquire PC Public Domain programs. Now, I know. Some programs offered in PC Public Domain as *shareware* are absolutely worthless or very misleading. Some of these programs offer attractive graphic display, but they do *not* work unless I pay $35-50. Why should I pay $6.00 to get that program to start with? I realize now that Steve Leon has been sheltering us CP/M public domain users from all those hungry (and greedy) programmers by refusing to include *shareware* asking money for giving you a demonstration program.

The most sensible solution is to remove all shareware from Public Domain. We users of Public Domain programs can certainly give feedback (bug reports, etc.) to programmers. After a thorough exposure of the program to us, then programmers can take it out to a full-fledged market, just like Buttonware has done.

I am really appreciative of Steve Leon's editorial policy, and I hope that PC/Blue is better that what we have in California in terms of refusing to distribute *demo* programs as a part of PD program disk at $6.00.

Sincerely yours,
Paul Naitoh, Ph.D.
San Diego CA

ANOTHER WAY TO DEFINE & DECLARE C VARIABLES IN ONE FILE

Gentlemen:

Readers may also be interested in the following method to "Define and Declare C Variable in One File" (Micro/Systems, Vol. 2, No. 4) which is easily documented and has performed for me successfully for the past several years as a computer consultant to industry and business.

Regards,
Edwards Fields
South Chelmsford, MA

```
/* DEFINE and DECLARE C VARIABLES in ONE FILE REDUX */

C>type flags.h

/* flags.h */
#ifdef INITFLAGS
     int flag1;
     int flag2 = 1;
#else
     extern int flag1, flag2;
#endif
          /*-----------------------*/

C>type define.c

/* define.c */
#define INITFLAGS
#include "flags.h"

define()
{
}
          /*-----------------------*/

C>type declare.c

/* declare.c */
#include "flags.h"

declare()
{
}
          /*-----------------------*/

C>msc /EP define.c

     int flag1;
     int flag2 = 1;

define()
{
}

C>msc /EP declare.c

     extern int flag1, flag2;

declare()
{
}
```

S100 USERS ! ! !

THUNDER/PLUS IS HERE 8087 ONBOARD

We at **LOMAS DATA PRODUCTS** never rest on previous success. The **THUNDER 186** has been an overwhelming success over the past two and a half years but we just made it better. **THUNDER/PLUS** is now the most powerful sixteen bit single board computer on the S100 bus. We took all the features of the **THUNDER 186** and added more memory and math coprocessor support and just for extra measure we increased the clock speed from 8Mhz to 10Mhz. This has been added to these additional features:

- 2 RS232 serial ports at 19.2 Kbaud max
- 1 Centronics compatible parallel port
- On board floppy disk controller will control both 8″ and 5¼″ disk drives including the new high density PC-AT disk drives
- 512K bytes of onboard memory standard, 1 full Megabyte of memory optional
- Up to 64K bytes of onboard EPROM
- Fully complies with the IEEE 696-1983 specification for the S100 bus
- Optional 8087 math coprocessor at 10Mhz
- Comes with CONCURRENT-DOS, the multitasking operating system that is MS-DOS compatible
- Will directly boot PC-DOS**
- One year warrantee on parts and labor

PC-DOS may be booted and used to run programs that can be configured to run on terminals or some programs that restrict their operation to ROM BIOS calls for console I/O. Programs that directly access the video RAM will only operate satisfactorily when used in conjunction with our PC compatible color graphics board, **COLOR MAGIC.

THUNDER/PLUS is ideally suited for any high performance 16 bit application, including but not limited to: high performance CAD systems, Scientific processing, data base applications, and process control applications. The 8087 coprocessor allows use of **THUNDER/PLUS** in applications not able to be run on previous 16 bit single board computers.

PRICE:
512K version.................................... $1195.00
1 Megabyte version........................... $1395.00
8087 option.. CALL

IBM COMPATIBLE S100 BUS COLOR GRAPHICS

LOMAS DATA PRODUCTS presents COLOR MAGIC, the most complete compatibility solution for S100 bus computer products. COLOR MAGIC includes three major hardware subfunctions which allow it to emulate the IBM-PC: An entirely compatible video function, an IBM-PC keyboard interface and an IBM-PC compatible timer interface including IBM-PC sound compatibility. HOW COMPATIBLE IS IT? Currently we are running MICROSOFT's FLIGHT SIMULATOR recognized as one of the severest tests of compatibility. We can also directly boot PC-DOS for the IBM-PC with no alterations. Other programs which have been tested and function without problems are: LOTUS 1-2-3, DBASE III, WORDSTAR and VOLKSWRITER.
COLOR MAGIC (16K byte version) .. $595.00

■**THUNDER 186** single board computer provides a high performance 16 bit computer all on one board. It is an ideal companion to the COLOR MAGIC to provide a low cost, high performance IBM-PC compatible system. The 8Mhz 80186 offers 10Mhz 8086 performance. THUNDER 186 provides all the components necessary to form a complete system including: 256K bytes of no wait-state RAM, 2 serial ports, a parallel printer port, high performance floppy disk controller controls both 5¼″ and 8″ drives simultaneously, full IEEE 696 (S100) bus for system expansion. The COLOR MAGIC and THUNDER 186 combine to provide performance you won't find in other compatibles. The price includes the powerful Concurrent DOS operating system. **THUNDER 186 .... $1195.00**

■**NV-DISK** is a solid state memory with software to emulate a disk drive under MS-DOS, Concurrent DOS, and CP/M-86. NV-DISK is entirely COMPUPRO software compatible allowing COMPUPRO users to take advantage of the lower cost and battery protection support offered by NV-DISK. It offers the advantage of high speed access and no moving parts. It can be battery protected to allow data to stay even while powered down. The board is available in either 512K or 2 Megabyte configurations and multiple boards may be used to create disk drives with up to 16 Megabytes of storage.

NEW LOWER PRICING

NV-DISK 512K, **$595.00** 2 MBYTE, **$995.00**

■**MEGARAM** is a high density, high performance dynamic RAM board with up to two Megabytes of storage. Megaram offers no wait state performance in 8086 systems with up to 10 Mhz processors at a fraction of the cost of comparable performance static RAM.
MEGARAM .... ¼ MBYTE, **$475.00**
½ MBYTE, **$595.00** .... 1 MBYTE, **$795.00**
2 MBYTE, **$1095.00**

IBM-PC COMPATIBLE S100-BUS SYSTEM

LOMAS DATA PRODUCTS offers IBM-PC compatible systems with performance far exceeding that available from IBM. You can purchase systems offering performance of an eight Mhz 8086 or up the performance of an 8Mhz 80286. Each system is capable of supporting 8Mhz math coprocessors. Our 8Mhz 80286 system offers IBM-PC compatibility while offering up to 2 times the performance of the IBM-PC-AT. For applications where PC compatibility is desirable but higher performance is a benefit or requirement LDP offers the only viable solution.

WHO IS LOMAS DATA PRODUCTS . . .

LOMAS DATA PRODUCTS has been shipping 16 bit microprocessor S100 bus products for five years. We have earned a strong reputation for reliability and performance over these five years. We were running MS-DOS (SCP-DOS/PC-DOS) before IBM knew the operating system existed. We offer a wide range of 16 bit operating systems including MS-DOS, CP/M-86, CONCURRENT CP/M-86 and CONCURRENT DOS. All our products are backed by a one year guarantee. We offer no 8 bit products and concentrate entirely on high peformance 16 bit systems. *If you are looking for the highest performance possible on the S100 bus, you can be sure LOMAS DATA PRODUCTS offers it.*

The S100 Bus Experts.

LDP

Dealer inquiries invited.

MS-DOS, trademark of Microsoft
PC-DOS, trademark of IBM
Concurrent CP/M-86, concurrent DOS, trademark of Digital Research

LOMAS DATA PRODUCTS, INC.

182 CEDAR HILL STREET, MARLBORO, MASSACHUSETTS 01752 □ TELEPHONE: (617) 460-0333 □ TELEX: 4996272

TO AT OR XT, THAT IS THE QUESTION!

Dear Mr. Libes:

I am a subscriber to Micro/Systems Journal and was a subscriber to the old Microsystems when you were the publisher.

I recall that in a recent issue of your journal there was an article by you about an AT clone that you purchased. I have been considering purchasing such a machine and would value any additional comments that you have. At present, I own a Chameleon Plus which, as you may know, has no expansion slots. The main use that I have for such a machine is scientific computing using higher level languages and word processing. I have been using Turbo Pascal, Modula-2, FORTRAN, and LISP. LISP, in particular, is very memory hungry.

I have been using IBM XT's and IBM AT's at work and they seem equally easy to use. The price difference between an XT and AT clone is substantial. Aside from the extra speed of the CPU on the AT, just what advantage would I get? For the most part RAM above 640K is usable only as ram disk and although the CPU on the AT's is usable in two modes, only the mode which is compatible with the regular PC or PC XT is used. Will new software be available for the AT's before they become obsolete?

I would very much appreciate any comments.

Sincerely,
Eugene Friedman
Los Angeles CA

I see two advantages to the AT vs. the XT.

The newer AT's run at 6/8 and 6/10 Mhz and have much faster hard disk drives. The result is that processing speed is typically six to nine times that of a standard XT and disk access time is typically two to three times faster. In this issue is a review of the new PC's Limited 12 Mhz AT clone. This superfast system provides almost a twelve times processing speed improvement over a standard XT and comes with a 40Mb hard disk that is 3.5 times faster than the standard XT hard disk.

Thus, if you are running applications which tax an XT (e.g. developing large program, using large spreadsheets, large databases, CAD, page makeup, network file server, multi-user system, etc.), then the speed improvements are certainly worth the added cost.

The second advantage is that I fully expect Microsoft to shortly release a new version of MS-DOS specifically designed to use the 286 instruction set, registers and protected-mode features. This version should be designed for users of LANs, window software, graphics oriented packages, etc. The likelihood is that many software vendors will offer versions of their more complex packages that run only on 286-based systems.

Keep in mind that today a basic AT clone with 640K of RAM and 30Mb hard disk can be assembled for as little as $1,600. A basic XT clone with 20Mb can be put together for as little as $900. The difference is only about $700.

DEBUG HAS IT TOO!

Dear MS/J,

The review by Randy Davis of the Phoenix PFIX-Plus Debugger was very good (M/SJ Vol. 2/No. 4). It gave me a good feel for the program and kept me interested from beginning to end. Part of my interest comes from spending hours and hours using DEBUG. One thing *bugged* me, however. Mr. Davis called attention to the feature of PFIX that treated a call to a subroutine or a system call as one instruction, not halting until after control has been returned after the call. DOS 3.x's DEBUG has this same feature! It is misleading to omit that fact.

Graeme McRae
Monmouth Junction NJ

WANTS MORE C & UNIX ARTICLES

Mr. Sol Libes:

Good to have you back. We need you!

I'd like to voice some preferences, if you please.

1) Less assembler, more C!

2) Don't forget UNIX. It's out there and growing fast and there is a real dearth of hard-core published on it.

3) How about the VME bus and DEC Q-bus? Using DEC's new LSI-11/73 chip set you can build quite a machine.

In case you cannot guess, I am lusting for a 68020 powered VME with a few dozen Megs of RAM.

Robert M. White
Houston TX

A CLONE BUILDER WRITES

Dear Sol,

Thanks for putting out the magazine. The sentiments expressed by others so many times in the letters section are a reflection of my feelings after finding the original Microsystems and then seeing it turn into a collection of pie in the sky articles.

Anyway, I now read Micro/Systems Journal and Micro Cornucopia which together cover my interests.

After reading your article on building a clone, I dove in and was eager to move into the *modern era* after being a CP/M user for years. As so many others have noted, the hardware fails to live up to the expectations. However, in my case, the software justifies the step down from a 5 Mhz Z-80. From this experience I have two items, one an endorsement, and the other a technical question.

First of all, I live in Haiti where things are little harder to get, especially information. I got a copy of Computer Shopper and looked at all the ads (a real job in itself) and decided to order my clone parts from American Digital Discount Association in Torrence, CA. I called them on the phone, got the exact prices, sent in a cashiers check, and had the parts in short order. I too have the SUPER PC/XT and floppy controller, mono/graphics board (Hercules clone) and a multifunction board. The mono/graphics board I got had a strange intermittent (aren't all intermittents strange?) that caused it to come up with assorted graphics characters with blink attribute randomly placed characters around the screen. I sent ADDA a note to this effect and they sent me a new board immediately, asking me to return the defective board only after I verified that the new board was good. That, to me, is great service from a "Cheap Mailorder Place." I recommend them to others looking for clone parts. Again, they do not have the cheapest prices, but the service is great in my opinion. The new board works fine.

The second point is a technical question. Have you noticed that the disk write speed is MUCH slower on your clone than it is on the *real* PC? I thought this was because I am using a mixture of old rejected drives I had around, however, we now have another system up and running with new drives and it is the same. Is there any cure? Do you think that the floppy board could be at fault? The hard disk seems to be about the same speed, as does the floppy read, but the write takes forever. It takes over 2 minutes to format the system.

A related question which you may be reluctant to publish the answer to is "Can these clones run with a set of IBM XT ROMS plugged into them?" I wonder about the code that somehow gets around the missing second mother board dip switch. Also, I see "Taiwan Basic" ROM chips advertised. Any experience with these? Can one then run IBM Basic? The disk basic I have does not support music, or should I say sound, which so many programs use.

Do you know of a user's group for clones like the "Super?" How about a listing of error codes? Are they the same as in the real "IBM?"

Thanks for your time and effort to make Micro/Systems Journal a valuable tool for me.

Ron Angert
Port-au-Prince, Haiti

Turbo Pascal Corner

by Stephen R. Davis

One of the most common complaints with home-grown software is the speed of its screen output. Somehow, when it is possible to watch the bouncing cursor as it moves from line to line drawing a new screen, it just doesn't seem to have the *jazziness* of those commercial packages whose output is seemingly instantaneous. And then there's that annoying flicker on the color/graphics adapter when scrolling.

In this column, we will examine tricks to make text output from your Turbo programs much faster (or without flicker). I have generated a Turbo source as an example (Listing 1). Two programs are contained in Listing 1, sharing most of the same code with the proper sections commented. This program was written to be easily dropped into your existing program sources (feel free to do so). Replace calls to Write or WriteLn with calls to QWrite and QWriteLn.

Note, however, that these routines only support text output and not graphic output. More specifically, these routines will support display modes 2, 3 or 7. Fast and Faster should support the EGA when in these modes also. Also note that QWrite and QWriteLn only accept a single text string, you must still use Write to display numbers. Neither programs work on CP/M-80 Turbo. (Any of you CP/Mers out there with high speed output tricks for Osbornes and other memory mapped displays, please send them in).

Fast works by accessing display memory directly. Think of your display as a matrix or spreadsheet with 80 cells across and 25 cells down. For convenience, programmers number from 0 rather than 1. Each cell contains 8-bits of ASCII character and 8-bits of attribute. The attribute field controls such things as full-bright, inverse video, blinking and color. To put a sentence on the screen, just place each successive character (with an attribute attached) into each successive cell of the spreadsheet. Remember that blank is just another character and that NewLines do not get displayed.

When a NewLine is desired, just drop down to the next row of the spreadsheet and back to column 0 and continue from there. Once you reach the bottom of the screen, it is necessary to scroll the screen up one line. The spreadsheet analogy of this is a *move block* up one row; that is, row 1, col 0 gets placed into row 0, col 0, row 1, col 1 into row 0, col 1 and so on until eventually row 24, col 79 gets moved to row 23, col 79. At this point you can clear row 24 and add whatever new text you desire.

This column features tips and techniques for using Turbo Pascal productively on MS/PC-DOS and CP/M microcomputer systems. It discusses typical problems and their solutions. Reader suggestions, comments, and questions are encouraged. Address them to Turbo Pascal Corner, Route 5 Box 107K, Greenville, Tx 75401 or through MCI mail, 289-6124.

HiSpeed Screen Output

This is exactly how Fast works. As with any real world problem, there are a few details to keep in mind, such as the fact that the display area for the monochrome and graphics screens do not start in the same place (this is to allow both to be present in the same machine). You will have to edit the constant SEGMNT to match your display. Of course, you could always add the code to decide at run time by examining location $0:$449. If this byte is a 7, then it is a monochrome; if <7, it is either an EGA or a CGA and if >7, it is definitely an EGA.

Fast does not move the cursor around. If you desire the cursor to be left after the last character output, add 'GoToXY (x_pos, y_pos);' after your call to QWrite or QWriteLn. Further, you can control where the next text will appear by modifying 'x_pos' and 'y_pos' to any legal value.

Some CGA cards (including IBM's) generate white specks all over the screen when you access screen memory. This so-called *snow* is caused by memory contentions between the PC's CPU and the 6845 CRT controller chip. The 6845 is a special purpose microcomputer which takes the display *spreadsheet* and sends it to the monitor. When the 8088 gets in the way of the 6845's reading of display memory, the 6845 does not know what to send to the monitor, so it sends a random block. This appears as snow and can be very distracting. Turbo avoids the problem by turning off the screen during screen access. This cures the snow, but causes an even more annoying flicker.

This problem can be avoided by asking the 6845 for permission to access memory. During the vertical retrace cycle, the 6845 is not sending anything of importance to the screen and so can be interfered with without generating snow. The code to avoid snow is present in all 3 routines and commented with the special (* *) comment. Removing these comments will remove the snow, but slow down the display dramatically (so much so that Fast actually becomes slower than normal Turbo output, but without any flicker).

Hardware hackers take note. Even during the vertical and horizontal retraces, the 6845 continues to access screen memory. This continual reading slows down the 8088 screen accesses considerably in both displays, but especially in the CGA. If the 6845 could be *turned off* to allow the 8088 unhindered access to screen memory, Fast's speed could be doubled. I was unable to convince the 6845 chip to go to sleep. Anyone know how to do it?

Fast is reasonably quick and straightforward. Treating the screen as if it were an 80 by 25 matrix is a useful model, but every reference of a matrix requires a multiply and an add (nonassembler programmers don't think about it, but the instruction 'A[i][j] := 0;' must perform a multiply). The add is no problem, but the integer multiply is a very slow instruction in the 8088 (only the divide is slower). In cases such as ours, a multiply can be replaced by a look-up table, which is much faster. The subsequent indexing into an array is equally quick. The improvement is not be terribly significant, but the trick is worth noting as it can be used in other types of software also.

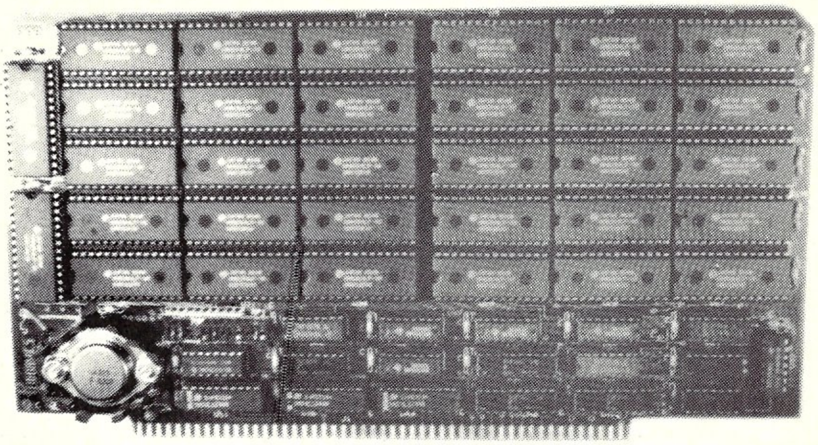
If you were to analyze where Fast spent most of its time, you might be surprised to discover that the scroll operation is by a large margin the largest user of CPU time. Actually, you can convince yourself of this quite easily. Clear the screen and type a file of more than 25 lines. The first 25 lines appear more rapidly than the remaining lines, which require a scroll operation.

One of the programmable registers of the 6845 controller chip is the starting offset address. It is possible to emulate a scroll function by reprogramming this starting

256K BYTES OF 100ns STATIC MEMORY

$395

NO HIDDEN CHARGES!
Performics prices include dynamic burn in and UPS 2nd day air delivery.
* 128K byte version (depopulated 256K) **$345**



S-100

STANDARD FEATURES

- 100ns memory chips
- Operates in excess of 12 MHz
- Certified system ready
  (industrial grade dynamic burn in)
- Supports 8 and 16 bit data transfers
- IEEE 696/S-100 compatible
- 24 bit addressing
- Address strappable to any 128K block within the 16 meg address range
- Extremely low power consumption
- Single +5 volt operation
- One year warranty

AVAILABLE SOON! High speed dynamic ram boards. 1 & 2 megabyte densities at Performics realistic prices.

PERFORMICS INC.

Delivering high performance products and excellent service at realistic prices.

(603) 881-8334 ORDER NOW!

TERMS: Check, Visa/Mastercard
Performics guarantees our boards will work in your system or return for full refund.

11 Morning Dove Road • Kingston, NH 03848
• Hudson, NH 03051

MasterCard VISA®

offset. I actually wrote a third version of Fast which used hardware scrolling but the results were not completely satisfactory. The biggest problem is that it leaves the screen in a state which no other software in the world knows how to deal with.

A better technique to avoid scrolling is to roll from the bottom to the top of the screen. You must clear out the current line (and maybe the next) to indicate where writing is taking place. The speed improvement is impressive, but it is not as attractive.

My next column we will be examining another aspect of Turbo Pascal. Send in your request and maybe we can take a look at your favorite problem. (μ)

Listing 1.

```
FAST & FASTER

{High and HigherSpeed screen output -
these routines provide very acceptable speed on all monochrome,
EGA, and many color/graphics adapters.  Snow generated by not
waiting for retrace is completely unacceptable on some CGA's.
Removing (* *) comments will put in a check to remove CGA
snow; however, result is actually slower than standard Turbo
output, although more pleasing since it doesn't flicker.
}

type
    outline = array [0..79] of integer; {this defines screen}
    display = array [0..24] of outline;
    strng = string [255];

const
    cga = $b800;          {offset of color graphics}
    ega = $b800;          {          ega}
    mono= $b000;          {          monochrome screen}
    attribute = $0700;    {this attribute is normal video}
    segmnt = mono;        {set this to match monitor type}

var
    screen : display absolute segmnt:0; {currently set for monochrome}
    rows : array [0..24] of integer;
    x_pos, y_pos : integer;     {cursor location}
    null_line : outline;        {used to clear a line of screen}
    value : integer;
    i, j : integer;

{****scroll N lines from bottom of screen towards top****}
Procedure Scroll (count : integer);
var
    index : integer;

begin
    for index := 0 to (24-count) do {move screen up count lines}
    begin
(*       repeat until ((Port[$3da] and 8) = 8); *)
         screen [index] := screen [index + count];
    end;
         {no clear count-1 lines at bottom}
    for index := (25 - count) to 24 do
    begin
(*       repeat until ((Port[$3da] and 8) = 8); *)
         screen [index] := null_line
    end;
end;

{****initialize quick output variables and clear screen****}
Procedure Init;
var
    index : integer;

begin
    x_pos := 0; y_pos := 0;
    for index := 0 to 79 do
         null_line [index] := attribute;
    Scroll (24);
       {for higher speed output only multiply at start}
    for index := 0 to 24 do
         rows [index] := index * 80;
end;

{****move cursor location down one line****}
Procedure NextLine;
begin
    x_pos := 0;
    y_pos := y_pos + 1;
    if (y_pos > 24) then
    begin
         Scroll (1);
         y_pos := 24
    end
end;

{****write a string to display screen****}
Procedure QWrite (outstrng : strng);
var
    count : byte absolute outstrng;
    temp : integer;
    offset : integer;
    fscreen : array [0..1999] of integer absolute screen;
```

ALL SALES ARE MADE SUBJECT TO THE TERMS OF OUR 90 DAY LIMITED WARRANTY. A COPY OF THIS WARRANTY IS AVAILABLE FREE, ON REQUEST.

DIGITAL RESEARCH COMPUTERS (214) 225-2309

TEXT TO SPEECH BOARD!

PC/XT COMPATIBLE. MAKE YOUR COMPUTER TALK!

A VERY POWERFUL AND AMAZING SPEECH CARD. USES THE NEW GENERAL INSTRUMENTS SPO256-AL2 SPEECH CHIP AND THE CTS256A-AL2 TEXT TO SPEECH CONVERTER.

THIS BOARD USES ONE SLOT ON THE MOTHER-BOARD AND REQUIRES A COM SERIAL PORT. BOARD MAY ALSO BE USED IN A STAND ALONE ENVIRONMENT WITH ALMOST ANY COMPUTER THAT HAS A RS232 SERIAL PORT. FEATURES ON BOARD AUDIO AMP OR MAY BE USED WITH EXTERNAL AMPS.

NEW!

DEMONSTRATION SOFTWARE AND A LIBRARY BUILDING PROGRAM ARE INCLUDED ON A 5¼ INCH PC/XT DISKETTE. FULL DOCUMENTATION AND SCHEMATICS ARE ALSO INCLUDED.

69.95 COMPLETE KIT   **89.95** ASSEMBLED & TESTED

HARD DISK CONTROLLER CARD

THE FANTASTIC KONAN KDC-230. WITH MANUAL. SUPER EASY TO CONFIGURE AND USE.

HALF SIZE PCB. $99.00 — ADD $8.00 FOR CABLES

MEX-PC MODEM SOFTWARE PACKAGE $49.95

A FANTASTIC COMMUNICATIONS PACKAGE WITH FEATURES TOO NUMEROUS TO LIST. SEE MAY/JUNE '86 MICRO SYSTEMS JOURNAL FOR FULL REVIEW OR CALL FOR BROCHURE. SUPPORTS COLOR, KERMIT PROTOCOL, Y MODEM BATCH, VT100 EMULATION, REMOTE OPERATION, ON LINE HELP, ETC. RUNS UNDER PC/MS DOS. 180 PAGE MANUAL & BINDER.

MEX-PACK LIST IS $99.95 — **SPECIAL $49.95**

HOT COMPUTER BOOKS!

GET THE MOST OUT OF THE INVESTMENT YOU MADE IN YOUR COMPUTER HARDWARE AND SOFTWARE WITH THE NEW WORDWARE SERIES OF COMPUTER REFERENCE AND TRAINING MANUALS. LEARN HOW TO MAKE ALL THAT EXPENSIVE SOFTWARE YOU BOUGHT REALLY COOK! *BUY ANY 3: TAKE AN EXTRA 10% DISCOUNT!*

ILLUSTRATED TURBO PASCAL	$13.50
ILLUSTRATED AUTOCAD	$18.50
ILLUSTRATED WORDSTAR	$18.50
ILLUSTRATED d BASE II	$13.95
ILLUSTRATED d BASE III	$18.50
ILLUSTRATED LOTUS 1-2-3	$15.50
ILLUSTRATED SYMPHONY	$17.50
ILLUSTRATED C PROGRAMMING	$17.95
ILLUSTRATED MS/PC DOS	$12.95
ILLUSTRATED MULTIMATE	$17.50
WORDSTAR CUSTOMIZING	$18.50
COMMUNICATIONS HANDBOOK	$12.50
ADVANCED TURBO PASCAL PROGRAMMING TECHNIQUES	$19.95

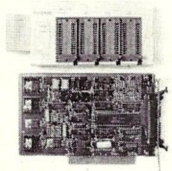
NEW!

S100 BUSS LIVES! WE STILL SELL S100 PRODUCTS AND CP/M SINGLE BOARDS. FREE CATALOG.

SCHEMA: Schematic Capture and Drawing Package

WORKS WITH PC'S AND COMPATIBLES. EASY TO USE WITH OFF THE SHELF PRINTERS, MOUSES, VIDEO CARDS, ETC. COMES COMPLETE WITH A HUGE COMPONENT DATABASE. SUPPORTS MONOCHROME, COLOR, OR EVEN EGA. FREE BROCHURE AND DEMO DISKETTE ON REQUEST. LIST $500 – **SPECIAL $396.15**

NEW! PC/XT EPROM PROGRAMMER $199

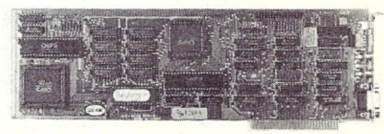


★ LATEST DESIGN ★ PROGRAMS UP TO 4 DEVICES AT ONE TIME ★ FEATURES EASY TO USE MENU DRIVEN SOFTWARE THAT RUNS UNDER PC OR MS-DOS. ★ USES AN INTELLIGENT PROGRAMMING ALGORITHM FOR SUPER FAST (8X) EPROM BURNING. ★ THIS PLUG-IN BOARD ATTACHES TO AN EXTERNAL MINI CHASSIS CONTAINING 4 TEXTOOL Z.I.F. SOCKETS. ★ NO PERSONALITY MODULES REQUIRED ★ AUTOMATIC VPP SELECTION: 12.5V, 21V, OR 25V. ★ EPROM DATA CAN ALSO BE LOADED FROM OR SAVED TO A DISKETTE. ★ PROGRAMMING SOFTWARE SUPPORTS: 2716, 2732, 2732A, 2764, 2764A, 27128, 27128A, 27256, 27256A, 27512, AND 27512A. ★ ASSEMBLED AND TESTED, BURNED. IN WITH MANUAL. $199 WITH SOFTWARE.

MICROTEK 2 M.B. EXPANDED MEMORY BOARD $195 (O. K)

MEETS LOTUS/INTEL/MICROSOFT SPEC. FOR EXPANDED MEMORY. FOR USE AS EXPANDED MEMORY (EMS), ELECTRONIC DISK, OR PRINT SPOOLER. WITH A SUPERB MANUAL AND INSTALLATION SOFTWARE. FOR PC/XT OR MOST COMPATIBLES. FOR 2 MEGABYTES OF 150 NS RAM (72 DEVICES) INSTALLED AND BURNED IN ADD $225.

PC/XT AT ENHANCED GRAPHICS ADAPTER



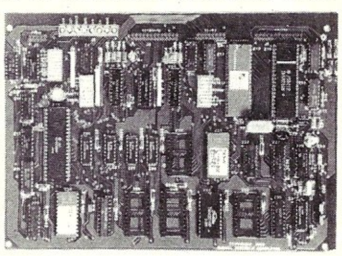
$249

- ★ 3 CARDS IN ONE!
- ★ 100% IBM COMPATIBLE.
- ★ COMPATIBLE WITH IBM EGA, COLOR GRAPHICS ADAPTER, OR MONOCHROME ADAPTER.
- ★ 256K VIDEO RAM! (4 TIMES MORE THAN IBM!)
- ★ DUAL FREQUENCY OUTPUT FOR EITHER EGA OR STANDARD RGB COLOR MONITORS.
- ★ PERFECT MATE FOR NEC MULTISYNC COLOR MONITOR!
- ★ LIGHT PEN INPUT.
- ★ FULL 16 COLORS.
- ★ PARALLEL PRINTER PORT.
- ★ A SUPERIOR BOARD AT ABOUT 1/3 THE COST OF IBM EGA!
- ★ ASSEMBLED, TESTED, AND BURNED IN. WITH MANUAL.
- ★ USES CHIPS & TECH. VLSI 4 CHIP SET.
- ★ FIELD PROVEN BIOS.

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 #ZRT-80   A&T ADD $50

(COMPLETE KIT, 2K VIDEO RAM)

BLANK PCB WITH 2716 CHAR. ROM. 2732 MON. ROM

$49.95

SOURCE DISKETTE - ADD $10

SET OF 2 CRYSTALS - ADD $7.50

FOR 8 IN. SOURCE DISK (CP/M COMPATIBLE) ADD $10

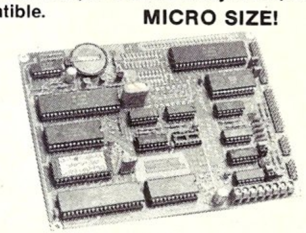
THE NEW 65/9028 VT ANSI VIDEO TERMINAL BOARD!

★ 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.

MICRO SIZE!

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 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.



$99.95

ADD $40 FOR A&T

(Full Kit w/100 Page Manual)

SOURCE DISKETTE: PC/XT FORMAT 5¼ IN. $15

NEW OPTION

PC/XT keyboard ROM. Allows use of IBM style keyboard. $15

Digital Research Computers

P.O. BOX 381450 • DUNCANVILLE, TX 75138 • (214) 225-2309

TERMS: Add $3.00 postage. Orders under $15 add 75¢ handling. No C.O.D. We accept Visa and MasterCharge. Tex. Res. add 5-1/8% Tax. Foreign orders (except Canada) add 20% P & H. Orders over $50 add 85¢ for insurance. Prices subject to change without notice.

*TM OF DIGITAL RESEARCH INC. (CALIF.) *IBM TM OF INT. BUS. MACHINES.   WE ARE NOT ASSOCIATED WITH DIGITAL RESEARCH INC. (CALIF.) THE SUPPLIERS OF CPM SOFTWARE

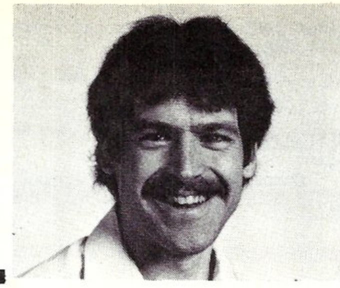
```
begin
    offset := rows [y pos];
    for temp := 1 to count do
    begin
        value := attribute + Integer(outstrng [temp]);
(*      repeat until ((Port[$3da] and 8) = 8);*)

  {this is slightly faster (since it avoids multiply) but less
   straightforward than high speed version:}
        fscreen [offset+x pos] := value;
  {this is straightforward high speed version}
  {     screen [y pos][x pos] := value;}
        x pos := x pos + 1;
        if (x pos > 79) then
            NextLine;
    end
end;

Procedure QWriteLn (outstrng : strng);
begin
    QWrite (outstrng);
    NextLine;
end;

{****main -- simple test program of quick write****}
begin
    Init;
    for j := 1 to 100 do
    begin
        for i := 1 to 24 do
        begin
            QWrite ('this is so called hispeed output');
            QWriteLn ('        this is more')
        end;
        QWriteLn ('******************************');
        for i:= 1 to 24 do
        begin
            Write ('this is normal Turbo output     ');
            WriteLn ('        this is more')
        end
    end;
end.
```

Listing 2.

```
FASTEST

{Highest Speed output --
these routines have all of restrictions of high and higher
speed routines with extra restrictions that: 1) it will not work
with any display adapter that does not use a 6845 CRT controller
chip (including all EGA cards) and 2) it leaves screen in state
you will have to clear with MODE command
}

type
    outline = array [0..79] of integer;
    display = array [0..24] of outline;
    strng = string [80];

const
    cga = $b800;            {offset of color graphics}
    mono= $b000;            {               monochrome screen}
    mono_index = $3b4;      {6845 index and data registers:}
    mono_data = $3b5;
    mono_mem = 2048;        {amount of memory on mono card in words}
    cg_index = $3d4;
    cg_data = $3d5;
    cg_status = $3da;
    cg_mem = 8192;            {16k in words}
    start_addr_h = 12;        {data register for offset start}
    start_addr_l = 13;
    attribute = $0700;        {this attribute is normal video}
    segmnt = mono;  {these equates are different for mono & cga}
    index_register = mono_index;
    data_register = mono_data;
    end_of_mem = mono_mem;

var
    screen : array [0..1] of integer absolute segmnt:0;
    x_pos, x_start : integer;         {cursor location}
    y_start : integer;                {current beginning line}
    index, value : integer;           {used to speed up output}
    i, j : integer;

{****position 6845 to start at a particular line****}

Procedure Position;
begin
    port[index_register] := start_addr_h; {out this to CRT chip}
    port[data_register] := hi (y_start);
    port[index_register] := start_addr_l;
    port[data_register] := lo (y_start)
end;
```

```
{****clear last line of display and move display up a line****}
Procedure Clear_Line;
var
    index : integer;

begin
    for index := 0 to 79 do
    begin
        screen[y_start] := attribute;
        y_start := (y_start + 1) and (end_of_mem - 1)
    end
end;


{****scroll N lines by repositioning 6845 controller start****}
Procedure Scroll (count : integer);
var
    index : integer;

begin
    for index := 1 to count do
        clear_line;
    Position         {then position controller to match}
end;

{****initialize quick output variables and clear screen****}
Procedure Init;
var
    index : integer;

begin
    x_pos := 0; x_start := 24*80; y_start := 0
end;

{****move cursor location down one line****}
Procedure NextLine;
begin
    x_pos := 0;
    x_start := x_start + 80;
    if (x_start >= end_of_mem) then
        x_start := x_start - end_of_mem;
    Scroll (1)
end;

{****write a string to display screen****}
Procedure QWrite (outstrng : strng);
var
    count : byte absolute outstrng;
    temp : integer;

begin
    for temp := 1 to count do
    begin
        index := x_start + x_pos;
        value := attribute + Integer(outstrng [temp]);

(*           add this test to remove cga snow -- cga only!!
        repeat                          {wait for the ok sign}
        until ((port [cg_status] and 8) = 8);
*)
        screen [index] := value;     {quick -- get data out!}
        x_pos := x_pos + 1;
        if (x_pos > 79) then
            NextLine;
    end
end;

Procedure QWriteLn (outstrng : strng);
begin
    QWrite (outstrng);
    NextLine;
end;

{***main -- simple test program of quick write****}
begin
    Init;
    for j := 1 to 20 do
    begin
        for i := 1 to 24 do
        begin
            QWrite ('this is so called highest speed output');
            QWriteLn ('       this is more');
        end;
        QWriteLn ('******************************')
    end;
end.
```

The C Forum



by Don Libes

Keeping Track Of malloc( )

malloc( ) and free( ) provide a simple general-purpose memory allocation package. malloc(size) returns a pointer to a block of at least size bytes. When the user is finished with the block, free(p) returns the malloc( )'d storage pointed to by p.

malloc( ) and friends (find the others by looking in your local C library manual) are quite easy to use. They do have some pitfalls, however, and I will be discussing one of them in this column, along with a solution.

The motivating problem with malloc( ) is this: Once a user has malloc( )'d a piece of memory, it is the responsibility of the user to return the memory (via free( )). If the user is finished with the memory, and destroys the pointer (by overwriting it, perhaps) instead of calling free( ), garbage is created.

The reason this is known as *garbage*, is that no one, not the user (through his or her data structures) nor the system (through malloc( )), is responsible for this data. It is inaccessible for the remainder of the life of the current program.

Although garbage collectors (routines that comb through memory looking for garbage) have been written in C environments, their use is very specialized. C, itself, has no support for garbage collection. The reason is because, in general, it is impossible to look at memory and tell whether you are looking at data, pointers or garbage. It all looks the same.

Because of this, one must be very careful when allocating memory. In order to keep track of dynamically allocated storage, we typically have two alternatives:

1) One must create a function that *knows* how to free our complex data structures, or

2) One must keep track of dynamic allocations in such a way that we don't forget to free( ) them when done.

Let us take an example structure that can be combined recursively to build up a tree.

```
struct node {
        struct node *left;
        struct node *right;
} *a, *b, *c, *d, e, *f;
```

Imagine now that we made the appropriate calls to malloc( ) and the correct assignments so that we built a tree that looked like this:

```
        a
       / \
      b   c
     / \ / \
    d  e f  0
   / \   . . .
  0   0
```

Now suppose we are finished with the tree, and we would like to free( ) all the nodes in it.

Since the tree has a very simple structure, it would be easy to create a function that simply free( )'d nodes in a postorder traversal. (We will discuss later why it has to be postorder.)

However, notice that node e was statically allocated. It would be a mistake to free( ) node e (since free( ) is only defined for malloc( )'d data). Unless we add some information to the node structure as to whether it was allocated dynamically or statically, it will be impossible to decide whether to free( ) the node or not.

Now this is a very simple tree, and it is easy to write a tree-walk function, but now suppose that you have many different types of trees, linked-list and other data structures. You must write a free-like function for each of these and you must change it when you change any of the tree data structures or construction functions.

The alternative is to allocate memory in a way in which track of which nodes depend upon other nodes is done automatically.

I have written an allocator that does exactly this. It can be used to allocate any data structure, and you do not have to modify your data structure. Like malloc( ), one call is used for allocations, and one call for deallocations.

These routines are as simple to use as malloc( ) and free( ), and further, are compiler and machine-independent since they call malloc( ) and free( ) to do the real allocation and deallocation.

key_malloc( )

The three user-callable functions in this package are:

```
char *data = key_malloc(char *key, unsigned int size);
void key_free(char *data);
void key_change(char *key, char *data);
```

key_malloc(key,size) behaves like malloc(size) except that a key must be supplied. This key is used to group allocations together.

In order to return key_malloc( )'d memory, the buffer pointer is passed to key_free( ). If this pointer is also a key, any memory key_malloc( )'d with that key is key_free( )'d.

For example:

```
a = (struct node *)key_malloc(0,SIZE);
a->left = (struct node *)key_malloc(a,SIZE);
a->right = (struct node *)key_malloc(a,SIZE);
a->left->left = (struct node *)key_malloc(a->left,SIZE);
a->left->right = (struct node *)key_malloc(a->left,SIZE);
a->right->left = (struct node *)key_malloc(a->right,SIZE);
a->right->right = NULL;
```

Here we are building up a tree with the root a. When we are finished with the tree, we return all storage with:

```
key_free(a);
```

Now we may begin to see the power and elegance of this approach.

Though I didn't mention it earlier, the definition of key_free( ) implies that it recursively descends through the tree, free( )ing everything it comes to that was key_malloc( )'d. That is because we built the tree in such a way that each node depends upon its parent node. Notice that the NULL nodes, the e node, and any statically allocated nodes will not be key_free( )'d because they were not key_malloc( )'d to begin with.

If we had created the tree with malloc( ), not only would we 1) have to understand the exact structure of the tree and 2) know whether the nodes were dynamically or statically allocated, but we must 3) free( ) the tree in a post-order traversal.

WHY POST-ORDER?

You must be careful not to free a node before free( )ing its children, because malloc does not guarantee that once data is free'd, it is not overwritten.

In particular, if we wanted to free a very small tree, a, with only a left and right node, the following code fragment (pre-order) is not portable:

```
/* WRONG */
free(a);
free(a->left);
free(a->right);
```

Similarly, an in-order traversal has similar problems. It must be done this way (post-order):

```
/* RIGHT */
free(a->left);
free(a->right);
free(a);
```

The reason the first example is non-portable (i.e. it may work on some systems when the moon is in the right phase) is that free(a) relinquishes the memory that a points to and yet, in the next line, free(a->left) refers to that memory.

If you read the documentation carefully, free( ) makes no guarantees that one can continue to reference free( )'d storage (at least until the next malloc( )) even though it is a reasonable expectation. In fact this expectation is naive. Indeed, many implementations of free( ) will modify memory returned to them immediately (typically using the first few bytes as a pointer to the next element in a linked-list of unallocated buffers).

Looking back at key_free( ), we are assured by its definition (and looking in the source below) that when it deletes a node, it will first delete any nodes that depend upon that node. Thus:

```
key_free(a);
```

recursively free's (in the correct order) any key_malloc( )'d data that depends upon a.

Any storage that was key_malloc( )'d may be key_free( )'d explicitly at any time. Thus, it is possible to do partial key_free( )'s, such as:

```
/* free right half of tree */
key_free(a->right);

/* build it back up again */
a->right = (struct node *)key_malloc(a,SIZE);
a->right->left = NULL;
a->right->right = NULL;

/* free entire tree */
key_free(a);
```

A function you may find helpful with key_malloc( ) and key_free( ) is key_change( ). key_change(newkey,buffer) allows a part of the tree to be saved from garbage collection by associating the named buffer with a new key.

For example, if we create the tree a, as above, we may want to save the right branch but destroy the rest of the tree. We can do that as follows:

```
key_change(0,a->right); /* or any other key */
key_free(a);
```

Since a->right is no longer keyed to a, it and its children will not be deallocated when the rest of a is. Of course, we must save the value of a->right somewhere, or else we will not be able to access it ourselves!

#new and #key_new

An aid to using malloc and key_malloc are the macros below:

```
#define new(type) (type *)malloc(sizeof(type))
#define key_new(key,type) (type *)key_malloc(key,sizeof(type))
```

These macros allow us to rewrite calls to malloc such as:

```
nodeptr = (struct node *)malloc(sizeof(struct node));
nodeptr = (struct node *)key_malloc(key,sizeof(struct node));
```

as

```
nodeptr = new(struct node);
nodeptr = new_key(key,struct node);
```

This makes code much more readable, while making it less likely that you will use the wrong cast or forget the cast entirely. You can see this technique used in Listing 1.

Before concluding this column I would like to mention an obvious extension to this package which while I have not implemented it, I probably will at some point in the future. If you really want to understand my code, you might try the following:

Create a function called, say, key_end( ). Upon being called, key_end( ) will print out all buffers that have not been free( )'d, which we can simplistically call garbage. A really nice implementation will print out not only the address but their sizes, their contents (interpreted into printable form if necessary) and the line number where the memory was allocated.

This last piece of data could be gathered by setting up a macro such as,

```
#define key_malloc(key,size) key_malloc1(key,size,__LINE__)
```

where key_malloc1( ) does the real work. (__LINE__ is a symbol automatically declared by the preprocessor for just this purpose.) This would be great for debugging!

CONCLUSION

I have presented several functions that allow the user the ability to easily keep track of dynamically allocated memory. key_malloc( ) and friends are oriented towards hierarchical structures (such as trees) where one is constantly building up structures and tearing them down again.

key_malloc( ) runs on top of malloc( ) and introduces both a small space and time overhead on top of what malloc( ) already consumes. However, the effect of these functions is to relieve the programmer of writing many utility functions and keeping track of every malloc( ).

There are many opportunities for refinement in the ideas presented here. Generalized graphs can be supported with reference counts. Circular data structures typically require true garbage collection. Both of these are fascinating topics. Their implementation and study can provide the programmer with tremendously powerful tools and knowledge.

I encourage readers to write to me about topics or problems that you want to know about. I want this column to be reader driven. Write to me care of M/SJ, Box 1192, Mountainside, NJ 07092. (µ)

Don Libes is a computer scientist working in the Washington, DC area. He works on artificial intelligence in robot control systems. He is also the son of Lennie and Sol Libes.

Listing 1.

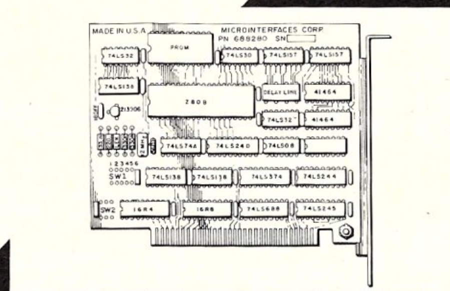
```
#include <stdio.h>

/*
The following km link structure is used to build up a singly-linked
list.  Each link in the list points to one malloc({th})'d item as well
as the key it depends upon.  Thus, it isa simple operation, to scan
during a key free({th}) to find all the links and their data that
should be free({th})'d.
*/
static struct km_link {
        char *data;      /* pointer to malloc'd memory for user */
        char *key;       /* pointer to what it depends on */
        struct km_link *next;    /* next km link */
} *first_km_link = 0;           /* initially empty list */

char *malloc({th});
#define new(type) (type *)malloc(sizeof(type))

char *
key_malloc(key,size)     /* returns data or 0 if none left */
char *key;
unsigned int size;
{
        struct km_link *link;

        if (!(link = new(struct km_link))) return(0);
        if (!(link->data = malloc(size))) return(0);
        link->key = key;

        /* insert link into linked list of km_links in no */
        /* particular order.  Currently in front of list. */
        link->next = first_km_link;
        first_km_link = link;

        return(link->data);
}

key_change(key,buffer)
char *key;
char *buffer;
{
        struct km_link *link;

        for (link = first_km_link;link;link=link->next) {
                if (link->data == buffer) {
                        link->key = key;
                        return;
                }
        }
        fprintf(stderr,"key_change: could not find buffer to change key\n");
}

key_free(p)
char *p;        /* be may be both data and a key */
{
        struct km_link *link;

        /* look for nodes dependent upon p */
        for (link=first_km_link;link;link=link->next) {
                if (p == link->key) {
                        key_free(link->data);
                }
        }

        /* if the key was also key_malloc'd, free it (and its link) */
        for (link=first_km_link;link;link=link->next) {
                if (p == link->data) {
                        unlink_km_link(link);   /* remove from linked list */
                        free(p);                /* free the (now) data */
                        free((char *)link);     /* free link */
                        break;
                }
        }
}

/* remove this link from the linked-list of km links */
static unlink_km_link(p)
struct km_link *p;
{
        struct km_link *link, *previous_link 0;

        for (link = first_km_link;link;link=link->next) {
                if (link == p) { /* found it */
                        if (!previous_link) {
                                /* first link in list */
                                first_km_link = p->next;
                        } else previous_link->next = p->next;
                break;
                }
                previous_link = link;
        }
}
```

SOLUTIONS!

POWER

The Power to Run Z-80 software on a PC!

Never before has it been easier, or more affordable to add the power of Z-80 compatability. Plug this half size card in your PC, XT, AT or compatible and within minutes you will be running the most powerful CP/M software available including the most popular assemblers, compilers, word processors, spreadsheets, etc. Included with each Z-80 card is our incredible RUN/CPM™ emulation software which will allow your PC to emulate 100's of popular CP/M computers, such as "Kaypro," "Osborne," "Morrow," Etc. Your CP/M programs will run up to 300% faster, with full color, and you can run them from your PC's hard disk or "RAM" disk, and run MS-DOS background programs such as Borland's "Sidekick" on top of your CP/M programs. RUN/CPM™'s amazing disk emulation feature will actually allow your PC's floppy drives to Directly READ, WRITE and FORMAT over 165 different CP/M disks, offering you the utmost in data disk compatability. The RUN/CPM software and Z-80 Co-processor are the perfect solution to bringing the power of the Z-80 to the PC. Call today and order the most affordable Z-80 Co-Processor ever!



Z-80 Co-processor $199.95

RUN/CPM™ V20/30 Emulator
RUN/CPM™ is also available for the popular N.E.C. V-20 microprocessor chip. This version supports the 8080 instruction set and runs on PC's and XT's. Order now at only $99.95 software only, or $124.95 including an 8 mhz V-20.

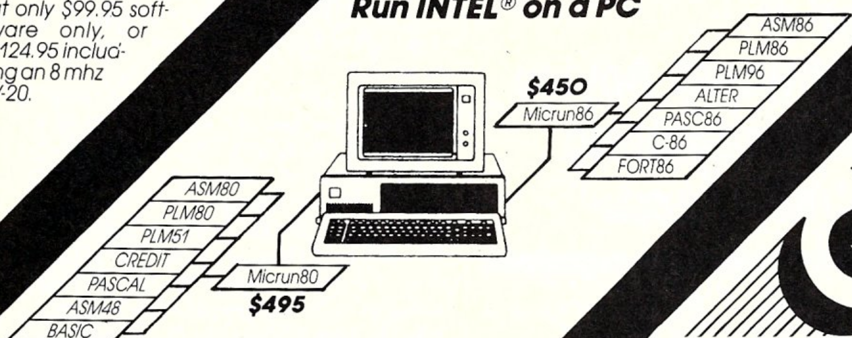
Fast! Hard Disk Back Up.

Quiksave™ is the solution to all of your hard disk back up needs. Quiksave's incredible speed will allow you to back up 10 megabytes of data in only 10 minutes! Because Quiksave backs up data on to standard floppy diskettes, there is no need for an expensive tape drive system. Best of all, Quiksave comes with 20 "Free" pre-formatted floppy disks,* so you can start backing up your valuable hard disk data immediately. Don't risk costly data loss, order Quiksave™ today and rest assured.

*Enough disks to backup 10 Megabytes of data on most systems.

COMPATABILITY

Run INTEL® on a PC



QUIKSAVE
ONLY **$99.95**

SPEED Backup 10 Megabytes in 10 Minutes

Transform an IBM PC, XT, AT, or compatible into an "Intel" Micro Development System, capable of running 8 bit ISIS-II, and 16 bit ISIS-III (UDI) software while remaining 100% MS-DOS compatible. MICRUN 80, our 8 bit ISIS-II Interface is priced at $495 and includes the above pictured Z-80 card plus all necessary software for running 8 bit Intel software on PC's and AT's. MICRUN 86, our 16 bit Interface is $450 and does not require hardware. MICRUN 80 and MICRUN 86 are the solutions to Intel compatability on a PC.

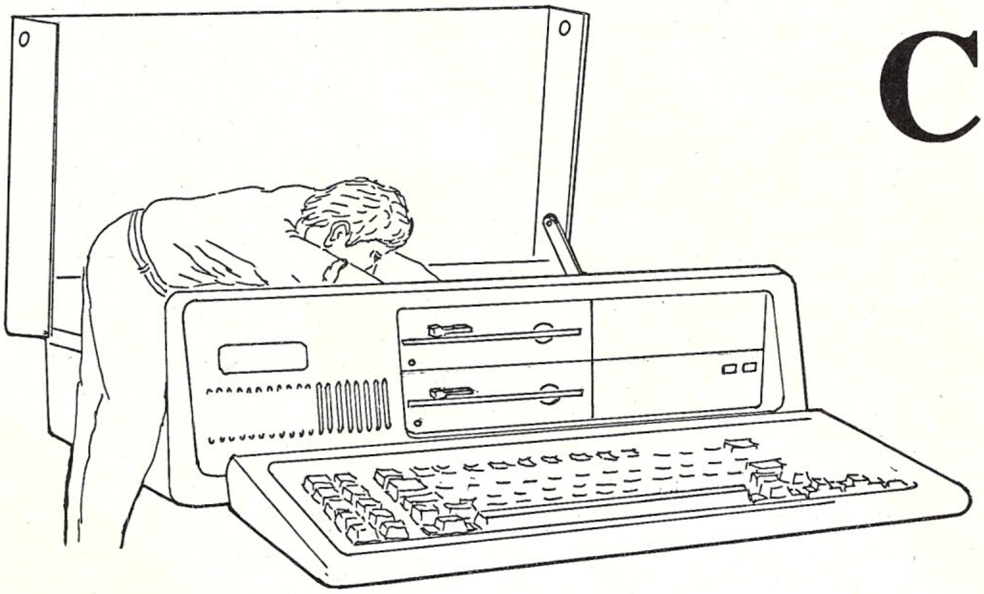
Call to Order, or for more information.
All products come with a 30 Day Money Back Guarantee.

1-800-637-7226

MICRO INTERFACES
6824 N.W. 169th Street
Miami, Florida 33015
(305) 823-8088
Telex 5106004680 MICRO INTER CO.

RUN/CPM is a registered trademark of Micro Interfaces Corporation; Sidekick is a registered trademark of Borland International; Kaypro is a registered trademark of the Kaypro Corporation. Intel is a registered trademark of the Intel Corporation; CP/M is a registered trademark of Digital Research; Z-80 is a registered trademark of Zilog Corp.; Quiksave is a registered trademark of Micro Interface Corp.

Roll Your Own PC Clone



by Sol Libes

Putting together a PC/XT clone today is pretty much like putting together a custom hi-fi system. The only tools required are screwdrivers and a pair of pliers. A large number of dealers offer the components at very competitive prices. A few PC component suppliers advertise in M/SJ and a more complete listing can be found at the end of this article. Most of these dealers sell mail order.

However, be warned that buying mail-order has its problems. Some dealers advertise products they do not have in stock, some are slow to ship, many cannot answer technical questions and some often make substitutions. The best thing is to buy only from mail-order dealers recommended by friends. I can say that we have yet to receive a single complaint regarding a PC component supplier who advertises in M/SJ.

Even better prices can often be gotten at computer flea markets. Further, you can see and usually check out the components you buy before paying for them. I have seen some flea market sellers assemble the system for the purchaser, and check it out, at the flea market, with no extra charge. In any case, insist on a warranty. It has been my experience that mail order and flea market sellers generally replace defective components sent back to them, by return mail.

The net result of assembling a system yourself, is that you can save from 25% to 50% of the cost of a similar, fully-

Part II - Additional Advice On Assembling PC/XT Clones*

assembled system sold under a brand name by a local dealer.

MOTHERBOARDS

The typical clone motherboard comes with either 256K or 640K of memory. The 256K version may include sockets for installing more memory. If not, a plug-in card will be needed to add the additional memory. I prefer the boards that come with 640K.200ns memory chips are standard on boards that run at the IBM clock speed. If you are planning to run at a faster speed, the chips should be rated at 150ns.

The first part of this series appeared in the May/June 1985 issue of M/SJ. Articles on assembling AT clones appeared in the January/February and July/August 1986 issues of M/SJ.

Most clone motherboards contain sockets for plugging in up to 8 cards. Older style cards (à la the original IBM PC) have only 5 cards slots. The 8-slot card is definitely preferable.

Some motherboards offer a "turbo" mode which allows the user to switch between IBM's standard 4.77Mhz clock speed and a 6.6Mhz or 8Mhz speed. The turbo mode is enabled through a switch or a sequence of keystrokes from the keyboard. This is a very worthwhile feature. However, some of the boards which have this feature include it at the expense of other features. After all, there is only so much code one can put in a 2764 PROM. In one case, including this feature meant dropping the ability to address the second serial I/O port. Check this point when ordering the motherboard.

Also, be sure to check that the ROM BIOS is for an "XT" system. In other words, that it will boot the operating system directly from a hard disk if there is no floppy disk in drive A. Some of the older BIOS ROMs do not have this feature. The Phoenix ROM BIOS is generally recognized as one of the best.

Better performance can be obtained by replacing the Intel 8088 chip with a NEC V20 (see M/SJ Nov/Dec 1985 and Jan/Feb 1986). This chip will typically improve processor throughput about 5-10% and even more on math-intensive processing. A math coprocessor chip (8087) will provide even better math processing if the software uses it. Note that the 8087-3 is the standard speed version and the 8087-2 is the higher speed version which should be used on turbo boards.

If you are looking for even better performance, then you may be interested in an AT motherboard with a PC form factor. In other words, an AT motherboard with the same dimensions and mounting holes as a PC/XT motherboard. If you decide to go this route, then you should also use an AT-type disk controller and power supply.

CASES

Three different styles of cases are currently sold. One uses a metal shell that mimics the IBM PC/XT case. The second mimics the IBM AT case. The third provides a hinged top lid that provides easy access to the innards of the system. I find the later preferable. However, note that the installation and removal of the disk drives is more easily done in the IBM PC/XT style case. If you do a lot of this you will probably prefer the IBM style case. I have no direct experience with the AT style case, but several people have reported problems fitting components into this one.

Also, check the fit and finish of the case. I have seen some in which the alignment of the card guides and disk-mounting brackets left something to be desired. Mounting hardware should be included, although frequently this is supplied with the other components. Suppliers may include a speaker with the case.

POWER SUPPLIES

Most clone component suppliers sell a 135-Watt power supply as a standard supply. I have tested many of these supplies and found many to be inadequate if you have a hard disk or if you live in an area where the AC line voltage frequently drops to low levels. I recommend the 150-Watt size or larger. (See the review of PC's Limited 12Mhz AT clone in this issue for a further discussion.)

A particular problem is the 12V line, which often has the same current rating as on a 63.5W supply. The problem arises from the fact that the PROMs used on clones draw power from the 12V line, whereas the ROMs on the IBM-PC do not. This loads down the 12V line, leaving insufficient power for the hard disk drive. Make sure the power supply is UL listed and try to find out the DC power specifications for all power supply outputs.

Note that these are switching-type power supplies and should not be powered up without a load on them.

DISK DRIVES

A single floppy drive and hard disk is the minimum for serious computer use. However, I like to have two floppies, particularly when I have to copy from one floppy to another or when I backup files from my hard disk.

I prefer half-height drives. They take up less room and allow for the mounting of more mass storage components, Also, they are more reliable and consume less power because full-height drives use a belt-drive system that tends to slip as the belt ages causing speed fluctuations and read-write problems. The half-height drives use direct-drive motors, and so do not have this problem. Even IBM has finally given up on full-height drives. The floppy drives that have proven reliable are the Teac, Toshiba and Mitsubishi.

Note, that when installing the floppy disks, only one drive should contain a terminating resistor package, usually the drive A. Since most of the drives are sold with resistor packages, you should remove the one in the B drive. Drive A is the drive at the end of the cable.

When it comes to hard disks, Miniscribe, Seagate, Microscience, Shugart, Matrox, Micropolis, Priam, Quantum, Rodime, NEC, Fujitsu, Hitachi, Siemens and Control Data provide the most in reliability and performance. I have had bad reports about CMI and Tandon drives. The lower cost drives generally have slower access times (typically 80msec) while a premium price is charged for the faster drives (typically 40msec). If you expect to be using software which makes a lot of disk accesses (e.g. compilers, assemblers, database managers) you will find the added cost of the faster drives worth it.

DOS allows you to directly access a hard disk obtaining a maximum of 32 Mbytes. If you purchase a drive that is larger than this, you will need software to partition the drive into two, or more, logical drives (C,D and so on). Be sure that this software is provided with any drive larger than 30 Mbyte.

Most floppy-disk controller cards will drive only 2 drives, but some will drive up to 4 drives. If you are planning to add an Irwin tape streamer for backup and still have 2 floppy drives, you will want the latter since the Irwin uses a standard floppy interface.

For controller cards, those from the Data Technology Corp. (DTC), Western Digital (WD) and Xebec get the nod. Check to be sure that a manual and cables are provided. Most suppliers furnish a setup program with the controller although it is possible to configure a hard disk without it.

KEYBOARDS

Although clone keyboards look identical to the IBM-PC keyboard, beware of the differences. First of all, most clone keyboards have a touch that is usually much softer or spongier. If you are used to the IBM *feel*, you may not like the touch of many clone boards. Therefore, it's best to try out the keyboard.

Another thing that irritates users of both IBM and clone systems is that key locations are variable. Very often the positions of the left shift and backslash keys are reversed. So check that the keys are where you want them.

The most popular keyboards are patterned after the Keytronics 5150 and 5151. Although the Keytronics units are considered the best, the clone knockoffs are also very popular.

One additional point. If you are planning to use a turbomotherboard at 8Mhz, check that the keyboard will operate at this speed as some will not. For example, the Keytronics requires a change in the ROM BIOS timing delay to work at 8Mhz. This entails changing the BIOS code and burning in a new PROM, which is not a trivial task.

DISPLAYS

IBM currently sells monochrome, Color Graphics (CGA), Extended Graphics (EGA), and professional display adapter (PGA)controller cards. Clones of the monochrome, CGA and EGA cards are widely available. In addition, Hercules makes a high-resolution (720 x 348 pixels), monochrome controller which has also been widely copied by clone suppliers. Most of the clone EGA cards will emulate the monochrome, CGA and Hercules cards.

The CGA card can drive the composite, TTL and RGB type monitors, while the EGA can drive TTL, RGB and EGA type monitors. An EGA display system can provide high-resolution graphics (640 x 320 pixels) with up to 16 colors, while the CGA is limited to 640 x 200 pixels with up to 4 colors. The EGA card and EGA display combination will cost about twice the price of a CGA/RGB display combination. If buying an EGA or clone card be sure that it includes the software to emulate the other cards. Also, the Hercules clone cards should include software to drive the card. I have seen some clone vendors omit this software.

High-resolution, monochrome displays are referred to as "TTL monochrome." The difference in cost between the TTL monochrome and the lower resolution "composite" display is usually less than $50. If you are using a "Hercules" high-resolution controller card, the TTL monochrome type display is required. The CGA and mono cards will work with the composite displays, but do not give the same quality as a Hercules-type display. I find both the green and amber screens acceptable with a slight preference to the amber.

When it comes to CGA color displays, an RGB display should be used. These come in *high* resolution (640 x 200 pixels), for use with a standard CGA display con-

troller card and *ultra-high* (720 x 400 pixels) resolution for use with the high-resolution display controller cards. The latter are preferred for applications such as CAD. The top-rated RGB displays are from Princeton, Quimax, Polo, Taxan and Amdek. The Taxan, model 640, rates very high because it can accommodate CGA, EGA and ATT 6300 color (though you will have to change some DIP switch settings and the cable).

If you are using an EGA (Extended Graphics Adapter) display controller, you will need a special color monitor which operates at a different scan rate. The NEC Multisync unit is currently the most popular.

Some clone EGA cards provide only the EGA mode, while other provide as many as 4 different modes (EGA, CGA, mono, and Hercules). The latter is preferred but costs more. Cards which provide different modes use either board jumpers, a switch on the rear, or keyboard codes to switch modes. The jumpers are the least desirable. Most EGA cards will work with a second display controller in the system, but some will not. If you are planning to use two displays, check for this capability. Some EGA cards come with as little as 64K of video memory, and sockets are provided for adding up to 256K of memory. To get the 640 x 400 pixel, 16 color graphics operation you will need the 256K of video RAM.

EXPANSION CARDS

If your motherboard already contains 640K of RAM, then all you will need to complete the system is a plug-in card that provides serial/parallel I/O and a clock/calendar circuit. Check that the clock/calendar circuit uses a lithium battery (these are usually good for 5 years) or a standard hearing aid battery that is easily replaceable. Some cards use non-standard batteries which could be a problem replacing. Also be sure that software for setting the time and date is included with the card.

The I/O board should allow you to change/or and disable the port addressing, so that they do not conflict with other boards using the same ports. Also, if you want a game port, you will have to buy a multifunction card with this feature as I have not seen it available separately.

DOCUMENTATION

The manuals provided with clone components have improved tremendously over the past two years. In the beginning, you were lucky to even get documentation. Today, virtually all components come with very complete manuals.

However, there are still quality control problems. All too frequently you will find typographical errors, schematics that are incorrect and difficult to read. When you check a component at a flea market, be sure to check the manual as well.

PC/MS-DOS & BASIC

I recommend buying a copy of IBM PC-DOS from an IBM dealer. It comes with a terrific manual. However, note that the Basic interpreter program will not execute on a clone system. This is because the IBM PC/XT contains much of its Basic code in a copyrighted ROM. So, you will have to buy a copy of GW-Basic, or even better buy one of the improved versions of Basic. For example, Microsoft sells *Quick Basic* which is compatible with IBM basic and is far better.

ASSEMBLY HINTS

Here is the way I go about checking components and assembling the system. I hold off installing the components in the cabinet. Rather, I assemble them all on a table top (with a non-conductive surface) and I do it one module at a time so I can test each one. I start by connecting the power supply and speaker to the motherboard and turning on power. Within a few seconds, I should hear a beep from the speaker if everything is OK.

I shut down power, connect the keyboard, install the display controller card, and connect the display. If I hear the beep again a few seconds after powering up, I make sure that the memory-check messages are displayed on the CRT.

Once again, I turn off power and install the floppy disk controller card, connect the floppy disk to the power supply and card (as drive A, at the end of the cable), put a DOS disk in the drive, and turn on the system. After the beep and the memory check messages, I expect the system to boot from the floppy.

If everything checks out up to this point, I install the hard disk controller card and drive, turn on power, and boot the system. Lastly, I format and configure the hard disk. If all goes well, I park the heads on the hard disk, shut down power,and install all the components in the case.

FINAL THOUGHTS

If you decide to buy your system in component form and assemble it yourself, be prepared for some delays and problems. You will have to make a lot of phone calls to check out details and delayed shipping (a supplier with an 800 number is preferable). It may take up to 2 to 3 months to receive all the components. If you need a system quickly, buy it assembled rather than buying separate components.

If a mail order dealer has a technical support phone number, this is a definite plus. If it is an 800 number, it will be a miracle.

Be sure to get a warranty. Most suppliers offer a 90 day warranty. Some offer a 1 year warranty on hard disk drives and memory. Sometimes even longer periods are offered. Some mail order companies, on assembled systems, will even offer a 30-day

IBM AT Compatible Features:

- 512K RAM expands to 1MB
- 33% faster than IBM AT
- 1.2MB floppy drive
- 8/6 MHZ CPU 80286-8
- USA BIOS fully compatible
- Clock/calendar w/battery
- Hard disk/floppy controller
- AT style Keyboard
- 195 watt power supply
- 48 hour burn in
- Full documentation
- Includes setup software
- One year limited warranty

$1125*

MADE IN U.S.A.

** Monitor not Included*

Hard Disks

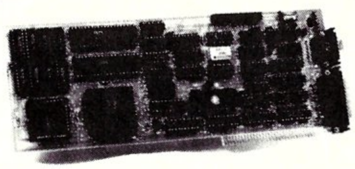
Let PC INNOVATIONS provide, install, and test the proper Hard Drive for your AT

(drive only)

SEAGATE 225 65MS	$310
SEAGATE 4026 39MS	$549
SEAGATE 4038 39MS	$595
SEAGATE 4051 39MS	$699
SEAGATE 4096 28MS	$1095
MINISCRIBE 70MB 28MS	$1195

10MHz Zero Wait State SI Rating 11.7 **$1495**

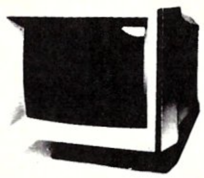
EVEREX EGA GRAPHIC CARD



- 16 colors 640 x 350 res.
- 256K video memory
- Parallel port

$249

NEC MULTISYNC

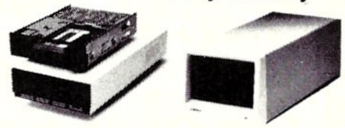


- EGA, PGA, CGA monitor
- 800 x 560

$569

EVEREX TAPE BACKUP SYSTEMS

The best tape backup money can buy

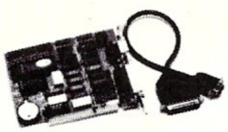


	INTERNAL	EXTERNAL
20MB	$589	$639
60MB	$759	$829
100MB	——	$929

Complete subsystem w/software

TRY PC INNOVATIONS PERIPHERAL CARDS FCC CLASS 'B' APPROVED

I/O for AT



$75

**XT Version available*
- Half slot
- Parallel port
- Serial port
- Optional 2nd serial

THE EDGE

$215

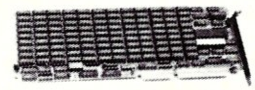
- Color display
- Hercules mono
- 132 column
- Printer port
- Software
- Lotus 123 graphiics

HERCULES COMPATIBLE

$99

- Monochrome graphics
- Requires no software
- Printer port
- Optional serial port
- Runs Lotus 123 graphics

AT MEMORY CARD



$165

- Up to 3MB
- No piggy back
- Split memory
- Uses 64K or 256K chips
- Software

MULTIFUNCTION CARD FOR AT

$175

- Up to 3MB
- Serial port
- Optional 2nd serial port
- Parallel port
- Game adaptor
- Software

HAYES 1200B MODEM

With SmartCom II

$349

2400 BAUD	CALL
EVERCOM 1200 BAUD	$128
U.S. Robotics:	
2400 Internal	$389
2400 External	$389

PC INNOVATIONS SPECIAL OFFERS

TEAC floppy disk 360K **$89** 1.2MB	$119
Toshiba floppy disk 360K**$95** 1.2MB	$125
Floppy disk controller	$40
Multifunction floppy disk controller	$95
XT 150 watt power supply	$75
AT 192 watt power supply	$109
Memory chips, Eproms, 8087,80287	CALL

EPSON FX 286	$539
EPSON FX 85	$379
EPSON LQ 800	$549
EPSON LQ 1000	$739
Toshiba P321 printer	$525
Toshiba P341 printer	$757
Leading Mfg. 13" RGB display	$259
MS400 AT 4 port serial card	$115
MS400 XT 4 port serial card	$110

VOLUME DISCOUNTS AVAILABLE

PC INNOVATIONS

UNIVERSITY AND CORPORATE BUYERS WELCOME

ESTABLISHED SINCE 1980

631 S. Main Street • Plymouth, MI 48170

(313) 451-0664

IBM, IBM AT, Hayes, Hercules are trademarks of their respective companies.

VISA & MasterCard Welcome
Prices subject to change

Our trained sales and technical staff is ready to serve you.

COMPETITIVE EDGE

631 S. Main St. — Plymouth, MI 48170 — (313) 451-0665

INTRODUCING

VELOCITY 286-10™/12

A T™ COMPATIBLE

10/12 MHz OPERATION

1024K on MOTHERBOARD

16 BIT HARD DISK TRANSFERS
SERIAL, PARALLEL, CLOCK

Enhanced Graphic Adapter Now Standard Runs with Color or Mono Monitor
1.2 MB Floppy, A T™ Compatible Keyboard, DOS 3.2

30 MEGABYTE FAST HARD DISK (38MS)
10 MHZ Velocity 286-10 As Above $2595.
12 MHZ Velocity 286-12 As Above $2895.
132 Column by 44 Lines No Cost Option
10 MHZ Velocity 286-10 with 70MB 28MS H.D. $3095
12 MHZ Velocity 286-12 with 70MB 28MS H.D. $3395.

SHIPPING NOW!!!

— OPTIONS —

80287 Math CoProcessor
Computone Xenix Tape Back Up $999
30 to 100 MB Hard Drives
20 to 100 MB Streaming Tape Back (2 minutes per 10 MB)
CPM®/Z80H Emulator Card Only **$395.00**
8MHZ 8088 Slaves (up to 31) for Multi-User
3 MB Memory Expansion Only **$795.00**
Monochrome & Color Monitors
Basic Xenix® 1024K, 8 Serial, 30MB System **$4095**

NO EXTRA CHARGE FOR VISA OR MASTER CHARGE

CPM is a registered trademark of Digital Research Inc. Hercules is a trademark of Hercules. A T is a trademark of International Business Machines. Velocity 8 and are a trademark of Competitive Edge. Screen is PC Paint Brush copyright and product of 2Soft, Inc. Xenix is a trademark of Micro Soft.

money back, no-questions-asked, guarantee (this is certainly worth some extra bucks).

Be prepared for some quality control problems and for delays while you exchange a component or two. If there is a user group in your area, try and find a *buddy* who has had some experience with clone assembly and who can help you if you have a problem.

In the end, you will have a system tailored to your needs and you will have learned a lot about the internal workings of the PC/XT.

Suppliers of PC/XT/AT Components

Atlaz Computer Supply
616 Burnside Ave
Inwood NY 11696
(516)239-1854

Digital Research Computers
Box 381450
Duncanville TX 75138
(214)225-2309

The Wholesale Outlet
1 Interstate Ave
Albany NY 12205
(518)459-7883/(800)344-4387

Yilon USA Inc
31352 Via Colinas, Suite 103
Westlake Village CA 91362
(818)991-4330

Competitive Computer Components
202 W. Center St, Suite 101
Duncanville TX 75116
(214)780-5796

B.G. Micro
Box 280298
Dallas TX
(214)271-5546

Continental Peripheral Unlimited, Inc.
2048 West Pepper St
Alhambra CA 91801
(818)282-121

Ultra-Comp
2600 N. Broadway
St. Louis MO 63102
(314)436-0296/(800)435-2266

BTE Computers Inc.
14644 N. Cave Creek #6
Phoenix AZ 85022
(602)867-8962/(800)233-6035

USA Electronics
2847 Anode Lane
Dallas TX 75220
(214)350-0212

MFG Inc.
212 Fourth Ave, N.
Nashville TN 37219
(615)244-3844

Advanced Intelligence Technology
4100 Spring Valled Rd Suite 400
Dallas TX 75244
(214)490-0344

Mendelson Electronics Co., Inc.
340 East First St
Dayton OH 45402
(513)461-3525

Electronics Parts Outlet
1921C W. Thunderbird
Phoenix AZ 85023
(602)375-0181

Timeline Inc.
1490 W. Artesia Blvd
Gardena CA 90247
(213)217-8912/(800)872-8878

Information Processing Systems, Inc.
2470 Joe Field Rd
Dallas TX 75229
(214)243-4221/4222

Owl Services
Box 116
Mertztown PA 19539
(215)682-6855

Computer Supply
Box 164
Valley Stream NY 11582
(516)239-1939

Altex Electronics Inc.
10731 Gulfdale
San Antonio TX 78216
(512)828-0503/(800)531-5369

Imperial International Co.
935 S. San Gabriel Blvd
San Gabriel CA 91776
(818)285-1256/1257/1258

Tinma Technology
16560 Harbor Blvd Ste-R
Fountain Valley CA 92708
(714)775-3968

Computer Parts Galore, Inc.
56 Harvester Ave
Batavia NY 24020
(716)343-6133/(800)431-9008

Micro Accessories Corp.
Box 4512
Bay Terrace NY 11360
(718)428-4943

Microland
6808 Bissonnet #159
Houston TX 77074
(713)668-4695

Gems Computers
3064 Postwood Dr
San Jose CA 95132
(408)946-2494

Orientec Corp. of America
748 Monterey Pass Rd
Monterey Park CA 91754
(213)268-4001

Complete Computer Systems
3413 Long Prairie Rd
Flower Mound TX
(214)539-8355

J.M.System Inc.
1633 W. Washington Blvd
Montebello CA 90640
(213)724-5585/5586

KJT Corp.
4505 Ratliff Dr
Addison TX 75001
(214)931-7181/(800)442-2065

Texas Electronic Sales
Box 13476
Arlington TX 76094
(817)274-6723

Otto Electronics
895 Rt 130
E. Windsor NJ 08520
(609)448-9165

Computer Component & Service Center
10501 Decatur Rd
Philadelphia PA 19154
(215)824-1085/(800)556-5353

Microperch Computers
4768 Lincoln Ave
Cypress CA 90630
(714)995-3600

Walker Electronics Co
3521 Hacienda
Dallas TX 75233
(214)339-4916

Z-Nix Co., Inc.
9537 Telstar Ave, Suite 121
El Monte CA 91731
(818)575-3700

Compstar Inc.
1348 Elm Ave
Glendale CA 91201
(818)842-4606

Computer Expert Inc.
1501 Broadway
NY NY 10036
(212)840-7871

Lolir
13933 N. Central #212
Dallas TX 75243
(214)234-8056

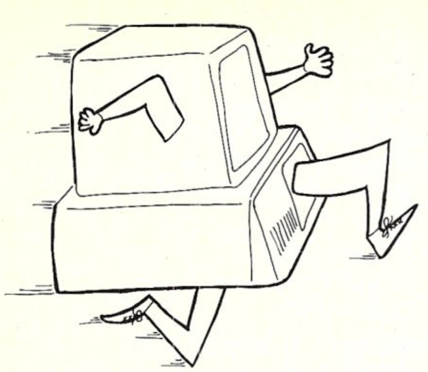
True Data Products
195 Linwood St
Linwood MA 01525
(617)278-6555

JDR Microdevices
1224 Bascom Ave
San Jose CA 95128
(408)995-5430/(800)538-5000

Floppy Disk Services Inc.
39 Everett Dr, Bldg D
Lawrenceville NJ 08648
(609)799-4440/(800)233-0306

NDRC
8511 Manderville
Dallas TX 75231
(214)750-9889

Magitronic Technology, Inc.
9-02 43 Road
Long Island City NY 11101
(718)706-7670
info/(800)227-5454 orders



Speeding Up The PC/XT

by Doug Severson

Build PC-SPRINT, A Circuit That More Than Doubles The Processing Speed of The PC/XT

PC-SPRINT is a low cost, speed up modification for the PC/XT and most clones. It can double processor performance, does not use an expansion slot, does not permanently change the computer, and maintains software compatibility without restrictions. As an additional bonus, it adds a true hardware reset, no more *keyboard lockup*.

The claim is real. This modification was done on an IBM PC running an 8Mhz V20 processor at 7.37Mhz. The actual performance increase provided by PC-SPRINT depends on what your particular computer can handle, and on choice of processor. In my case, the faster clock provides a factor of 1.55X (7.37/4.77); the V20 is variously rated at 1.1 to 1.4X over an 8088 at the same clock, giving an overall factor somewhere between 1.7 and 2.1X. The Norton SI index, widely used as a measurement of PC speed (standard PC running at 4.77Mhz has an SI = 1.0), registers 2.8 using PC-Sprint with an NEC V20 8Mhz processor.

Unfortunately, a 7.37Mhz clock, with no *wait states*, pushes the PC's design to its limit. Your particular setup may not be capable of full speed, but with a simple component change to PC-SPRINT (explained later), you can find your machine's maximum. If your PC has a 5Mhz 8087 (number cruncher chip), it must also be replaced.

Before deciding to try this project, check one very important detail. PC-SPRINT depends on replacing the PC's 8284A clock IC with a circuit board. On most PC's and XT's this chip is socketed and easily removed. However, this is not always so. Therefore, first check that the 8284A is in a socket. If not, you must unsolder it from the motherboard and put a socket in its place. This isn't too difficult, but may not be worth the effort.

HOW IT WORKS

The basic idea behind PC-SPRINT is simple: run the processor at a higher clock rate and processing speed goes up proportionately. On an AT this can be done merely by replacing the processor's crystal, but for the PC/XT things aren't quite so easy. The operation of the timer circuits (e.g. interrupt, sound) and some plug-in cards (e.g. color graphics adapter) require the original clock frequency to operate correctly.

PC-SPRINT attempts to solve this problem by running two clocks simultaneously (the original and a higher frequency). Only the processor's clock is increased, while keeping all other frequencies constant. Obviously, the processor chip must be replaced with a higher speed version.

To gain easy access to the needed connections, PC-SPRINT replaces the 8284A clock generator on the PC's motherboard, but reconnects some of the 8284A's signals through itself, including the oscillator signal, thus keeping that constant.

By adding another 8284A and crystal, a new frequency source is generated which can then be used to drive the unused external frequency input of the original 8284A. The original 8284A then chooses which source to operate from, based on the logic level at its F/-C input (high = external, low = crystal), and generates the processor's clock accordingly.

The 8284A contains additional circuits to generate processor reset and to select and synchronize the processor's *ready* line (used to cause the processor to wait for slow devices). By cross wiring the oscillator outputs of each 8284A to the other's external frequency input and using the ready select circuits of the new 8284A, the PERIPHERAL CLOCK output (which feeds the timer circuits on the motherboard) is kept constant at its normal frequency as its source is selected by the same logic signal used for switching the processor's clock.

The 8284A also generates the processor's reset. By *diode ORing* the motherboard reset line with the debounced output of a momentary switch, a true reset (as opposed to the software generated "ctrl-alt-del") is provided. This *diode OR* violates normal logic thresholds, but works in this case because the reset input of the 8284A is a Schmitt trigger type with higher than normal levels (when the hysteresis is added).

The unused reset circuit of the added 8284A is combined with the output of a toggle switch to generate the logic level which switches the processor's clock, giving manual control of speed. Both the reset switch and the speed switch use a pull up resistor to supply logic *high*, and a capacitor to filter switch *bounce*. These slow transitions are *squared off* by the Schmitt trigger inputs.

CONSTRUCTION

A complete, tested circuit board is available from Exec-PC (see sidebar). However, for those wanting to acquire the parts themselves and make the printed circuit board, here is the information you will need.

The parts list shows all the needed components. Radio Shack catalog numbers are given for some parts. Most items are available by mail order. The resistor/capacitor values are not critical and substitutions can be made (but try to keep within +/- 10%). Be sure the processor (8088 or V20) is rated at 8Mhz. It is also a good idea to get extra crystals (18.342Mhz or 20Mhz) in case your PC won't run at full speed. The external switches can be mounted in any con-

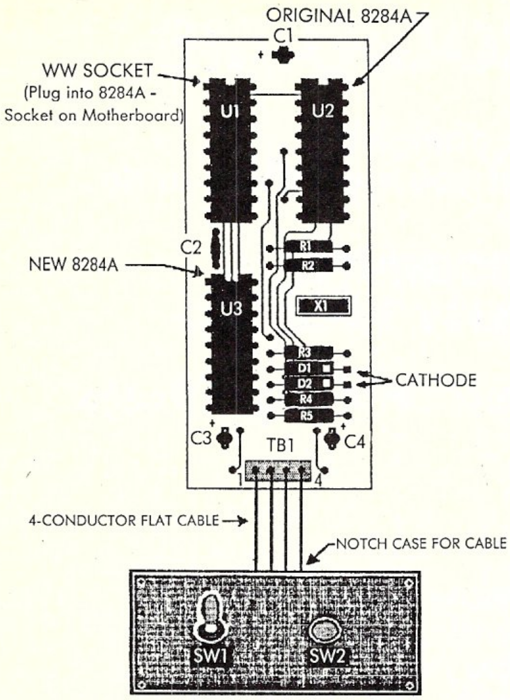


Figure 1. Parts Placement

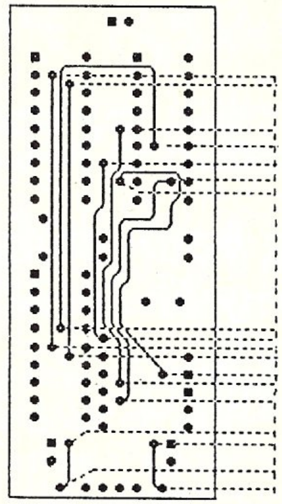


Figure 2. Solder "feed-though" wires top and bottom if board is doubled sided. If single sided, duplicate the top connections with 30 gauge wire on bottom of board.

PARTS LIST

Resistors (all ¼W)
R1 & R2 - 560 ohms
R3 - 3.3Kohms
R4 & R5 - 10Kohms

Capacitors
C1, C3, C4 - 10ufd 16V Tantalum (RS# 272-1436)
C2 - .01ufd ceramic disk (RS# 272-131)

Diodes
D1 & D2 - 1N914/1N4148 (RS# 276-1620)

IC's
U2 & U3 - Intel 8284 Clock Generator
U4 - Intel 8088-2 8Mhz

Crystal
X1 - 22.11 Mhz HC/18 case, see text

Miscellaneous
U1 - 18 pin wire wrap socket or header
TB1 - 4 conductor terminal strip (RS# 276-1389)
SW1 - SPDT Toggle switch (RS# 275-326)
SW2 - Normally open push button switch (RS# 275-1549)
Switch Mounting case (RS# 270-230)
4 Conductor cable (RS# 278-757)18 pin Solder tail IC sockets for U2 & U3 (RS# 276-1992)

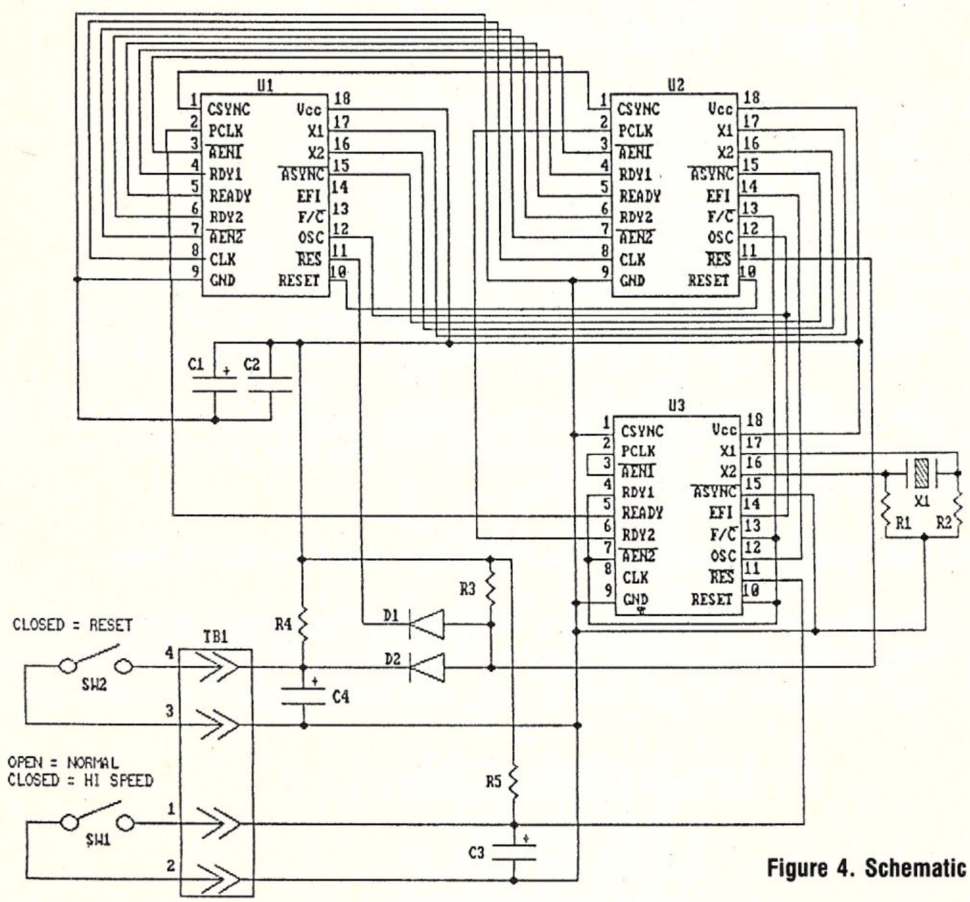


Figure 4. Schematic Diagram

venient manner, but the plastic box specified is handy and cheap. If you can't locate the terminal strip for the cable, it can be directly soldered in the board, but this will make routing the cable more difficult. The square pads on the top of the 2x artwork (Figure 3) mark pin 1 of IC's, + side of polarized capacitors, and cathode side of diodes.

The key to building PC-SPRINT is making a printed circuit board from the artwork below. You do not have to make the board doubled-sided, though it's neater; just do the bottom and duplicate the top connections with 30 gauge wire. The 2X artwork is very accurate and *camera ready*. The *top*, or component, side is shown as you would actually view the board from the top. The bottom (solder side) is shown as you would see it if you could look through the board from the top; thus, it is a *mirror* image and must be flipped over to see how it would be if you saw it from the bottom. This may be confusing, but is standard practice, and really presents no problem for photographic techniques as the negative can be turned over for actual transfer to the board.

After the board is etched, cleaned and trimmed to size, check it for shorts and breaks. Drill all holes with a #64 wire gauge drill bit. Slightly enlarge the holes for the cable terminal strip.

Next, solder in the jumpers if you're using a single sided board. If double sided, solder pieces of bare wire on both top and bottom to make the feed through connections (see *feedthru sketch*, Figure 2). Trim off excess wire.

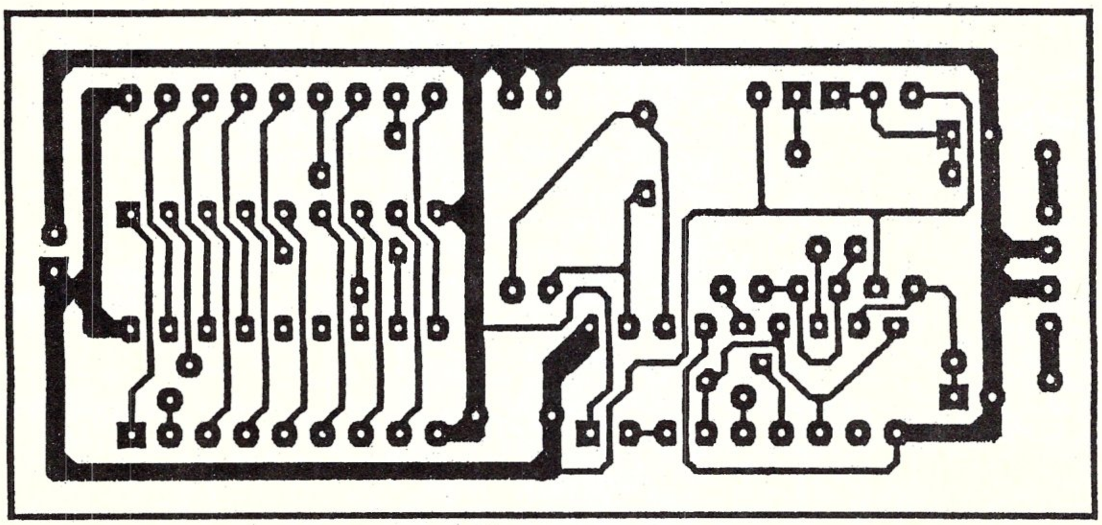
Using the *partlist* drawing (Figure 1), place the components in their proper holes, forming the leads as you go, and solder in place. Place a small piece of cellophane tape on the top of the board where the crystal will go to prevent the crystal's case from shorting. Install the crystal vertically and solder in place.

Install two 18-pin *solder tail* IC sockets at U2 and U3, and an 18-pin wire wrap socket at U1. When you insert the cable on the PC-SPRINT board, make sure you have used the *normally-open* contacts on the push button switch, otherwise, the computer will be held reset.

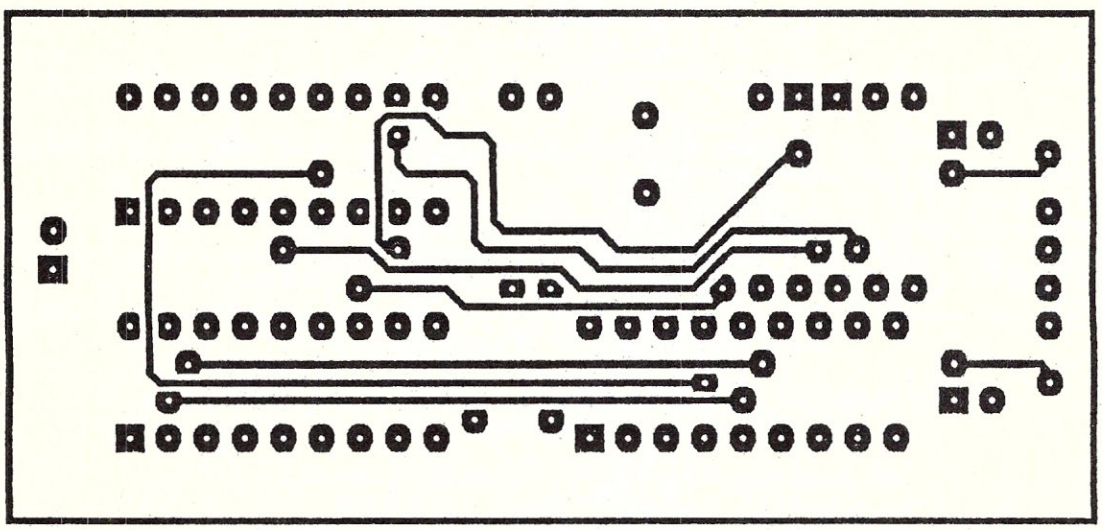
INSTALLATION

If you've gotten this far, the rest is easy. Unplug the computer from the AC power to prevent any shock. While you're at it, disconnect all external cables (CRT, keyboard, printers, modem, etc.) to protect them from you and you from their power systems. Replace the 8088 with an 8Mhz rated 8088 or V20. Move the original 8284A to the PC-SPRINT board, route the switch cable and install PC-SPRINT board in place of the 8284A. Reassemble the system.

If everything is OK, the computer should *boot* normally. At the DOS prompt, push the reset button. The system should boot the same as *power-up*. Now run some of your programs (use backup copies just in case) to verify that everything else works.



bottom layer (as viewed from above)



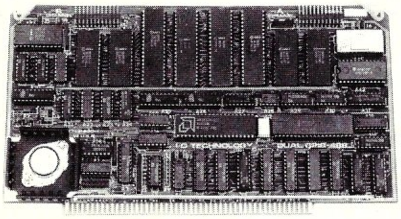
top layer

Figure 3. Artwork for printed circuit board, twice actual size. Finished size should be approximately 1.15 x 2.65 inches.

INPUT/OUTPUT TECHNOLOGY, INC.

25327 Avenue Stanford, Unit 113, Valencia, CA 91355 • (805) 257-1000

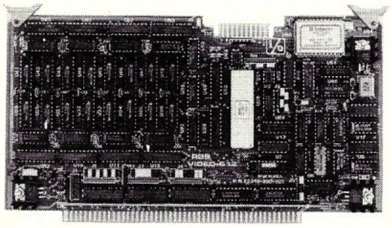
Uncompromising Additions to your S-100/IEEE-696 BUS



DUAL GPIB-488 INTERFACE BOARD

A Stand-Alone, Independently Controlled Dual Channel IEEE-488 I/O Processor. Interface Activity Modes for Controller-in-Charge, Controller Assigned or Terminal Bus Slave, and all Interface Functions are handled transparent to Host System CPU through an on-board CPU and DMA controller. User Friendly operation.

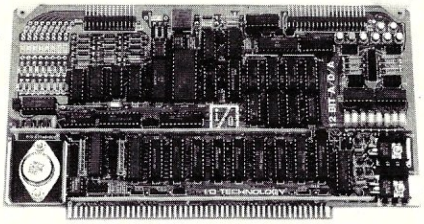
A&T, P/N 52748-800-102



RGB COLOR GRAPHICS BOARD

Programmable resolution up to 512 x 512 pixels with 4 local video planes and on-board graphics processor. Color mapper allows 16 colors from a palette of 4096. Light pen input. Plus more ...

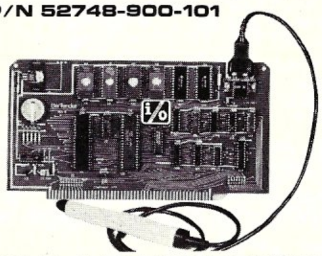
A&T, P/N 52748-300-101



12-BIT A-D-A CONVERTER BOARD

8 Channel A-D: 12 microsec. Conversion, 50KHz Sample Rate, Programmable Gains, Offset and Diff./Single Modes.
8 Channel D-A: 2 microsec. Settling, Bipolar V or Unipolar I Output. Programmable Reference levels, Dual-Ported Channel Refresh RAM. **16/8-Bit Data Transfers** via I/O or Memory Mapped

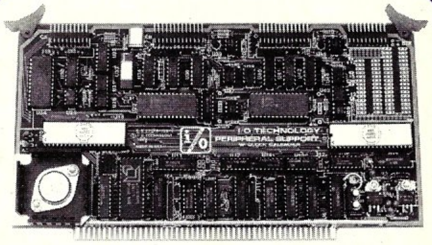
A&T, P/N 52748-900-101



BAR CODE PROCESSOR BOARD

The BarTender is a stand-alone I/O Processor that reads and prints most common Bar Codes. Includes bi-directional reading, wand interface, clock/calendar with battery. Extensive documentation and software.

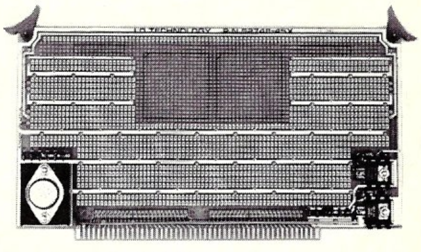
A&T,52748-500-101 Without Wand
A&T,52748-500-201 With Wand



PERIPHERAL SUPPORT BOARD

Two Serial SYNC/ASYNC Ports with RS-232, TTL or Current Loop Outputs, three 8-Bit Parallel Ports, three Timers, Real Time Clock/Calendar and Response Programmable Interrupt Controller. Small Proto Area with +5 and ±12v.

A&T, P/N 52748-150-101



MULTI-PURPOSE PROTOTYPING KIT

Industrial Quality with Plated-Thru holes for Wire-Wrap or Solder projects. Complete with +5, ±12v Regulators, Bus Bar, Filter Capacitors, and Manual.

P/N 52748-450

ALSO AVAILABLE: MULTI-FUNCTION I/O BOARD, SMART PROTOTYPING KIT, 128Kx8/64Kx16 STATIC RAM MODULE

SPECIFICATIONS SUBJECT TO CHANGE WITHOUT NOTICE.

Flip the speed switch to high and rerun the same programs. They should run noticeably faster. Since PC-SPRINT speeds up only the processor, programs that do a lot of I/O work (disk, screen, etc.) won't show much improvement. Processor intensive operations, like spreadsheet recalcs, graphics, in-memory database searches and so on, should be much faster.

If the system does not run, or does not run reliably, recheck all your work. Also keep in mind that there may be components which can't handle the speed increase. Except for the processor, the basic design of the IBM-PC/XT should be capable of operating at 7.37 Mhz, but some adapter and ram expansion boards may not, particularly if somebody cut corners and used 200ns rams instead of 150ns types. The thing to do in this case is to back off the speed a bit. Remove the 22.11Mhz crystal from the PC-SPRINT board and replace it with a 20Mhz or even an 18Mhz version, still a substantial speed up. The following crystals yield the following effective computer speeds:

14.3181 Mhz	4.77 Mhz (standard)
18.0000 Mhz	6.00 Mhz
20.0000 Mhz	6.67 Mhz
22.1184 Mhz	7.37 Mhz

OPERATION

There are two public domain programs I use to speed up my PC. NOPRTYCK.COM stops all memory parity checks (my regards to the unknown author). Some expansion boards will work OK at high speed, except that their parity check circuits are too slow. If you can't boot up at high speed without the infamous "PARITY CHECK x" message, restart at normal speed and execute this program (once between resets) *before* switching to high.

The other, WARMBOOT.COM, is a simple program to set the flag which causes the BIOS to skip memory tests on reset. If you include it in your AUTOEXEC.BAT file, each time you restart with PC-SPRINT's reset button, the computer will come up very quickly (the same as a ctrl-alt-del).

Operation of PC-SPRINT's switches is simple. The toggle switch determines the speed (high or normal) and the push button switch causes a hardware restart without powering down. In use, the speed switch can be changed anytime, on-the-fly, even while a program is running, without problem, but try to do it at DOS prompt level. Keep the switch box out of the way so that you don't inadvertently hit the reset button. When you really want to restart, tap the button firmly but quickly, otherwise the warmboot flag will be lost.

Exec-PC Review
Continued on page 94

Editor's Note: This article is actually a highly condensed version of a set of files that Doug Severson has placed in the public domain. The files include all the drawings and can be printed out on a standard PC-compatible dot matrix printer. As such, it is an experiment in *Freeware Hardware*, which Doug has pioneered. The entire PC-Sprint package of files can be downloaded from the Exec-PC electronic bulletin board system (414-964-5160). It is in a file called PCSPRINT.ARC The ARC extension means that it is an ARChived file which contains all the PC-Sprint files in compressed format. The ARC utility program, also in the public domain, will be needed to un-ARChive the file.

The complete PC-Sprint product can be purchased from Exec-PC, Box 11268, Shorewood WI 53211. For $89.95 you get the completely assembled board and the three crystals mentioned in Doug's article. They also sell the NEC V20 and Intel 8088-2 ICs for $10 when purchased with PC-Sprint. They also have versions for systems other than the IBM-PC/XT (e.g. PCjr, Tandy 1000 and Compaq). A 6" cable for connecting the PC-Sprint board to systems such as the Compaq is available for $5. They offer a 90 day, 100% money back guarantee.

Note that this project is recommended only for people who are experienced in handling ICs and have previous experience with projects of this level of sophistication. Standard industry practice and manufacturer's recommendations should always be observed. (μ)

POPPY. A true multi-user computer.

Charter Information's POPPY is a new variety that will make your office blossom.

POPPY handles all facets of your business — purchasing, sales analysis, and general ledger functions. Accounts payable, inventory control, materials requirements planning, and payroll. Plus desktop publishing and a host of other wonderful things that you need to do.

POPPY consists of a processor board, SmartCable, keyboard, and monitor. Plant multiple POPPY boards in a System 4 and watch your productivity bloom.

POPPY runs programs that operate on an IBM PC, including those that use graphics. And when it comes to dazzling displays, POPPY provides text and graphics in both color and black and white. Roses pale in comparison. Each POPPY has its own processor, one that is powerful enough to make a PC wilt. So you get to see what you want to see and do what you need to do as soon as you are ready.

What is really outstanding is POPPY's ability to fit in. The System 4 gives POPPY access to shared hard and soft disk drives, tape drives, printers and other devices. All under the control of a master processor that fulfills each POPPY's every need. The master takes care of spooled printing, controlled access to discs, file and record locks and operation of printers.

POPPY is no garden variety product.

If you are looking for a rainbow of colors to make your business ideas look better, wait until you see what POPPY can do — for all your users. You see, unlike standalone computers which require separate cards for each graphics mode, POPPY provides all the most popular graphics modes as standard equipment. It lets you switch among them through software — no fumbling with tiny dipswitches. That includes monochrome, color, EGA and Hercules graphics.

It's no wall flower, POPPY communicates.

Each POPPY comes with a PC compatible serial port. That means that each user can have a serial printer, a mouse or an external modem of its very own. And there is an optional 1200/2400 baud modem built in, if you specify it. Rooted in the S-100 technology of the System 4, communications between your users and shared information is faster than summer lightning. And that cultivates user productivity like never before.

How did they do it?

The designers of POPPY and System 4 have had years of experience implementing multi-user installations, in both manufacturing and office situations. They realize that multi-user environments demand something more than just another garden variety machine to insure adequate throughput and make information sharing practical among all your users. So, they created a multiple processor system. Every POPPY has its own processor and memory. Then, using the fastest communications available over a common data bus and the power of TurboDOS, they designed in a master processor to take care of the things that usually slow you down. And, up to sixteen POPPYs can be arranged in one System 4, working like one big bouquet of computing power.

How do you use it?

Our consultants can help you design the system that best fits your needs. The System 4 can be a complete computer for a small company or a departmental system. Or it can be a part of a distributed network for larger companies, providing multi-user access to data. If you have ordinary PCs that you want to link into a POPPY network, you can do so over a conventional serial LAN.

If I pick it, will it be easy to use?

Absolutely. If you already know PC-DOS, you're in the driver's seat. And if you need assistance, applications have help screens that take you through the rough spots. TurboDOS, the operating system for the System 4, lets you use all the most popular programs that run on the IBM PC — so if you're already using programs like Lotus 1-2-3, Wordstar, dBASE III+, and Flight Simulator, you won't have to transplant anything. You can merge data on your PC with information from POPPY. And all your users can share all the information.

CP/M already in use? No problem. You can run CP/M 80, CP/M86, and PC-DOS programs at the same time under TurboDOS, with file and record locking. (So the data goes to the people you want sharing it and no one else.)

Let's plant the seed.

Before you settle for just a garden variety computer, you're invited to pick POPPY. A computer with the power and flexibility to grow with you. Please contact one of our representatives today by phone or fill out the coupon and mail it. Who knows? Our relationship may bloom into something big.

Specifications:

Workstation:	
Processor	80188
Clock Speed	8MHz (no wait states)
RAM	512K
Display	Monochrome, CGA, EGA, and Hercules
Keyboard	Selectric style
Port	Serial port (IBM PC Com 1 compatible)
Modem	Integral 1200/2400 baud modem (optional)
SmartCable	11 pair shielded (200′ max.)
System 4	
Backplane	IEEE 696
Master processor	ICM memory mapped
Diskette	720K or 320K (software controlled)
Hard Disk	20, 46, or 76 MB
Tape backup	Stream out, media or file by file restore (optional)
Max. Poppys/case	Sixteen
Max. Printers/case	Sixteen

You've planted a seed in my mind.

Please send me more information on how I can use POPPY to cultivate productivity.

Name ________________

Title ________________

Company ________________

Address ________________

City ________________

State ________ Zip ________

Telephone ________________

Charter Information Corp

Where good ideas improve.

2421 Rutland
Austin, Texas 78758
(512) 835-1111

Cloning In The Fast Lane

by Sol Libes

AT clone vendors are keeping one step ahead of IBM. When IBM offered a 6Mhz AT, the cloners offered a 8Mhz clone. No sooner did IBM offer a 8Mhz AT then the cloners countered with 10 and 12Mhz versions. The result is that IBM has been losing market share in the AT marketplace (they no longer have the lion's share of the PC/XT marketplace).

Several companies are offering 10Mhz AT clones. PC's Limited was the first company to announce and ship a 12Mhz AT system. We had the loan of the new PC's Limited 286^{12} system for a few weeks. The following is our review.

A Review of PC's Limited New 12Mhz AT Clone

FEATURES

The PC's Limited 286^{12} offers several features in addition to its 6/12Mhz clock speeds. I would say that PC's Limited has set an example for others (particularly IBM) to follow. First of all, the set-up software is in ROM; hence, it does not require buying an IBM set-up disk if you want to make changes (the IBM set-up disk is also copy-protected). Second, the 286^{12} takes up less room (almost 30% less, same as the PC/XT footprint) on the desktop and has a very quiet fan.

Of course, its most outstanding feature is speed. In processing speed, it is twice as fast as a standard (6Mhz) IBM-AT. Further, its hard disk is also faster. Using the Core International benchmark test the 286^{12}'s 40Mbyte half-height hard disk showed an access time of 27msec compared to 38msec for the IBM-AT 20Mbyte drive. As a result, processing intensive tasks (e.g. recalculating a spreadsheet) took half the time while tasks that involve disk accesses (e.g. sorting a large data base) typically took about 30% less time.

You can switch from 6 to 12Mhz operation, or vice-versa, at any time from the keyboard by pressing Ctrl-Alt-\. When you go from 6 to 12Mhz, the machine produces a beep that goes higher in pitch and when you switch in the other direction it goes from higher to lower pitch. Neat! The 6Mhz speed can be used for loading copy-protected software that check disk timing, and then running them at a higher speed out of memory.

The 286^{12} also contains some diagnostic routines in ROM. If the unit detects an error during its power-up tests it prints a descriptive error message on the screen (not an error code as on the IBM-AT) and switches to the diagnostic/setup program in ROM. And, at any time, pressing Ctrl-Alt-Enter will put you into the program and you can return to your program where you left off. It's really nice for changing the time or date. The program contains an option to turn off parity-error checking which is handy if a RAM chip goes bad and you want to take a chance and continue anyway. Also, the routine to park the hard disk heads is immediately available.

A 1.2Mbyte floppy is standard. Space for three half-height drives, stacked one above the other, is provided. Thus you can have either two half-height floppies and a half-height hard disk or two hard disks and one floppy. Another possible configuration is a floppy, hard disk and tape backup unit.

The hard disk is a 40Mbyte Tandon. We put the hard disk through several hours of work and found it very fast and reliable. The disk controller is the same Western Digital controller used in the 8Mhz PC's Limited machine. We can verify that this is a very reliable performer. A device driver program was provided to configure the hard disk as two logical drives (C and D) to overcome the DOS limitation of addressing 32Mbytes of disk space.

The keyboard has a lighter touch than the IBM, but we found it very comfortable to use. The key layout is the same as the original IBM-AT keyboard.

No I/O ports are provided (as in the 8Mhz unit). A combination serial/parallel I/O card is available for $99.

The unit is certified as complying with FCC Class-B regulations. And PC's Limited has informed me that the unit has been submitted for UL listing; the power supply is already listed by UL.

The unit contains 1Mbyte of 100nsec RAM on the motherboard. We also marvelled at the motherboard which contains half the number of chips as the IBM AT motherboard, contains twice as much memory, and is about 30% smaller in size. This is mainly due to the use of an LSI chip set from Chips and Technology.

The 286^{12} has a 4-character LED and bar chart display on the front of the unit referred to as a "SmartVu" panel. It provides checkout information during the power-up test and indicates the active drive and track (or cylinder in the case of the hard disk) being accessed. It also indicates the clock speed, for a few seconds, after it is changed. When an error is detected, an error message is displayed (Times Square news marquee fashion) on the panel.

There is one caution that should be kept in mind when considering the 286^{12}. Some plug-in cards will not operate at this high data transfer speed. Therefore, if you plan to use memory-expansion cards, special display controllers, network interface cards, etc., check that these cards will work in the 286^{12}.

DOCUMENTATION

There is no doubt that IBM goes to great expense in preparing documentation providing manuals that set a standard in the industry. However, I often find the IBM manuals to be very redundant, so that I have to flip through an inordinate number of pages to find what I am looking for.

The 286^{12} system came with a small manual (by IBM standards). However, it did contain all the information I was looking for, and was nicely printed and bound. The manual lacked an index and there was one entire section missing with a note that a complete manual would be provided at a later date.

AS WE GO TO PRESS

PC's Limited has just announced the 286^{16}, the new fastest AT-class machine available today. It runs at 16Mhz and is intended to be used as a network file server, multi-user system or CAD/CAM workstation. The price is $2,995.

BENCHMARKS

We ran the same benchmark programs on the 286[12] that we ran on the PC, PC with V20 chip, Turbo XT, IBM-AT and AT-clone, the results of which appeared in the May/June 1986 issue of Micro/Systems Journal (page 63). These benchmark programs are in the public domain and are available on PC/Blue Volume 135. The times shown are in "minutes:seconds.tenths of seconds."

We also ran the Norton Utilities SI test (index for PC = 1.0) and found that an 8Mhz AT produced an SI index of 7.7 and the 286[12] produced an index of 11.7. This test means that the 8Mhz AT is 7.7 times faster than a standard PC while the 286[12] is 11.7 times faster.

Table 1.

		IBM-AT		PC's Limited		
test	IBM-PC	6Mhz	8Mhz	6Mhz	8Mhz	286[12]
1	:35	:15	:11	:11	:09	:07
2	:33.9	:05.7	:06	:09.4	:09.4	:05.6
3	:72.4	:20.4	:20.5	:18.6	:18.6	:16.75
4	1:47.9	:34.2	:34.4	:27.5	:27.5	:27.35
5	:52	:20	:15	:18	:13	:08
6	:42	:16	:12	:14	:10	:07
7	2:00	:43	:34	:40	:29	:19
8	1:35	:33	:27	:35	:26	:17
9	2:11	:47	:38	:48	:36	:21
10	:10	:05	:03	:03	:02	:02
11	1:36	:26	:24	:47	:33	:23

Tests:

1. Prime Number Calculation - compiled Basic
2. Disk Access - Contiguous reads/writes
3. Disk Access - Random & contiguous reads/writes
4. Disk Access - Random reads/writes
5. Basic Program - Integer addition #1
6. Basic Program - Integer addition #2
7. Basic Program - Floating point arithmetic
8. Basic Program - String Calculation
9. Basic Program - Data Look Up
10. Basic Program - Empty Loop
11. Basic Program - File Update

No technical documentation was provided (e.g. schematic diagrams).

RELIABILITY

We are located in an area that has large swings in AC line voltage. Although typically about 120VAC, it frequently drops to under 115VAC, sometimes to under 110VAC and we have observed during peak load period drops to under 100VAC. During these low voltage periods we experienced problems with the PC's limited machine. When the line voltage dropped to under 114VAC and a surge occured (e.g. an air conditioner cycled on) it caused the system to reset. A drop in line voltage to under 104VAC caused the system to quit entirely. As soon as the line voltage came back up the system resumed working (naturally everything in memory was lost).

We had an IBM-AT on the same AC circuit as the PC's Limited machine. This system kept right on working down to well under 100VAC with a high degree of resistance to line surges.

We checked the IBM specifications for the power supply. They rate their power supply to be good down to 104VAC. We measured the "Power Good" line coming out of the power supplies on both the PC and IBM-AT machines. This is normally +5VDC. We checked this line on both the 286[12] and IBM-AT. We found that when we dropped the AC line voltage to 103VAC, this line dropped to 0V on the 286[12]. This certainly meets the IBM specs. However, the IBM-AT's power-good line remained active all the way down to 82VAC. We therefore concluded that the IBM-AT has much better resistance to AC line voltage drops than the 286[12].

We reported the problem to PC's Limited. They told us it was the first report they had of this nature. They were very concerned about the problem and immediately shipped us a new power supply. The new supply (made by Tri-Mag Inc.) worked all the way down to 80VAC, curing the problem.

CONCLUSIONS

The PC's Limited 286[12] costs about $1,400 more then their 6/8Mhz system (after you add two serial ports and a parallel port which is included on the slower system). The increase in cost almost doubles the cost of the basic unit. Of course you could just opt for the 286[10] (6-10Mhz) which only ups the cost about $1,000. The question is.....is the extra speed worth the added cost?

If you are running mundane applications (e.g. word processing, small databases or spreadsheets) you will see little, if any, performance improvement and the added cost, in my opinion is not worth it.

But if you are using the system for applications such as sophisticated graphics (e.g. CAD, desktop publishing), very large spreadsheets or databases, compiling and linking large programs, or running a multi-user system, then the PC's Limited 286[12] provides the ultimate in 80286 system performance and is definitely the way to go.

Product Information

PC's Limited 286[12] (6-12Mhz):
$2,695
PC's Limited 286[10] (6-10Mhz):
$2,295
40Mbyte Hard Disk option:
$ 895

PC's Limited
1611 Headway Circle
Building 3
Austin TX 78754
(512)339-6800

Program Interfacing To MS-DOS

by William G. Wong

Part X - Why Move To DOS 3.X?

DOS 3.X, it's new, but is it worth using? If you are happy with DOS 2.X, then there may not be a reason to; but, if you need network support or would like some of the additional functions, then DOS 3.X might be just the thing. DOS 3.X is upward compatible with DOS 2.X.

DOS 3.X adds a number of useful commands, new device driver functions and a number of new or enhanced DOS function calls. The implications of these new features depends upon what type of programming is being done.

The new DOS commands are JOIN and SUBST. The JOIN command is used to logically connect a drive to a subdirectory on another drive. The subdirectory must be in the root directory and may only contain the two default subdirectory entries, '.' and '..'. The files on the drive can then be accessed as if they were in the subdirectory. SUBST works in the opposite direction. In this case, a drive can be logically linked to any subdirectory. These commands are available from the DOS COMMAND interpreter, but the interface at the DOS program level is not described in the documentation.

In addition, CONFIG.SYS file may contain an additional line which indicates what the last logical drive will be in the system. This allows the SUBST command to refer to a drive which is not a physical drive. The syntax for the last drive command in CONFIG.SYS is LASTDRIVE=x where 'x' is a drive letter.

DEVICE DRIVERS

The device driver INIT and MEDIA CHECK function have been enhanced. Three additional functions are included. Support for the latter is indicated by setting a previously reserved bit; bit 11 is the Removable Media (RM) bit in the device driver attribute word.

The INIT command parameter block has a single byte after the end of BPB array address field which contains the next physical block device number. This is useful when a device driver operates based upon its physical device number. In addition, the BPB array address is initialized by DOS to reference information in the text read from the CONFIG.SYS which caused the device driver to be loaded. The address points to the text after 'DEVICE=filename'. This information is often used to set initial baud rates for serial devices or the size of a RAM disk. Valid DOS function calls which can be used during the INIT are 01 through 0C hex and 30 hex.

The MEDIA CHECK function operation does not change if the bit is reset. An additional parameter is returned when the bit is set. The parameter is a four byte long pointer at the end of the parameter block which should be set to the volume name string associated with the previous disk if the disk has been removed, i.e. the return value is -1. The address should point to an ASCIIZ string of "NO NAME" if there was none. This allows DOS to determine whether the disk has been removed and replaced again.

The three new function device driver function codes are OPEN (13), CLOSE (14) and REMOVABLE MEDIA (15). These functions are only used if the RM bit in the device attribute word is set. The OPEN function is called when a file is opened on a block device or when a character device is used. The CLOSE function is called when the corresponding file close operation is performed. This allows the device driver to determine when the device is being used. Block devices which buffer information will normally flush the buffers when the all files are closed. Character devices can be setup to perform initialization and termination procedures when the driver is opened and closed respectively. This type of support was not available under DOS 2.X unless the device driver patched the DOS service interrupt to keep track of what files were being accessed.

The REMOVABLE MEDIA function simply sets the BUSY bit in the request block status word to 0 if the media is removable and 1 if the media is not removable. It is appropriate for only block devices.

The boot record for block devices has been altered to let a device driver know what version of DOS was used to format the media. DOS 2.X uses a three byte relative jump (opcode E9 hex) while DOS 3.X uses a two byte relative jump (opcode EB hex) followed by a single byte NOP (E9 hex). DOS 3.X also supports 16 bit FAT entries in place of the original 12 bit FAT entries. The larger FAT entry is used when the maximum number of clusters is 20740 or larger. This provides support for larger capacity disks and also allows smaller cluster sizes to be used on existing disks. The latter is preferable when a disk contains a large number of small files.

Adding this support for device drivers is relatively simple and is quite useful. However, it is not a requirement for device drivers used with DOS 3.X.

NEW DOS FUNCTIONS

The new and enhanced DOS functions can be divided into the categories: country information, extended error support, enhanced file creation support, file locking, and device redirection/network support. The following is a list of new or enhanced DOS functions:

Function	Description
38	Get/Set Country Information
3D	File Open
59	Get Extended Error
5A	Create Unique File
5B	Create New File
5C	Lock/Unlock
5E00	Get machine name
5E02	Set printer setup
5E03	Get printer setup
5F02	Get redirection list
5F03	Redirect device
5F04	Cancel redirection
62	Get program segment prefix

The Get/Set Country Dependent Information function (38 hex) is an enhanced version which defines additional fields. Also, if the country code in the AL register is FF hex then BX contains a sixteen bit country code number. This allows more than the 255 configurations set in DOS 2.X. The field definitions in the parameter block are:

Bytes		Description
2		Date format
5	*	Currency symbol
2	*	Thousands symbol
2	*	Decimal symbol
2	*	Date separator

2 *	Time separator
1	Currency format 0 = 0 symbol precedes value = 1 symbol after value bit 1 = x number of spaces between value and symbol
1	Significant decimal digits
1	Time format bit 0 = 0 12 hour/ = 1 24 hour
4	Case Map Call address
2 *	Data list separator
5	Reserved

* indicates an ASCIIZ string

The Case Map Call routine can be entered by an indirect long call using the address of the routine in the parameter block. The AL register is used to pass the character to be converted and the converted character is returned in AL. The function normally returns an uppercase version of the parameter.

The File Open function (3D hex) has been enhanced to support the network file locking support. Existing programs will still work as long as the program set the top five bits in the file mode control byte to zero as indicated by the DOS 2.X documentation. The new definition for the mode byte is:

Bit: 76543210
Name: ISSSRAAA

Name	Description
I = 0	inherit, = 1 private
S = 000	compatibility
001	deny read/write
010	deny write
011	deny read
100	deny none
R = 0	reserved
A = 000	read
001	write
010	read/write

The inheritance bit allows a process to hide certain open files from a subprocess. The security bit field allows a program to prevent other programs from simultaneously accessing the same file. A file can be opened using any mode associated with the file until it is closed. File locking, described later, can be used to reserve a specific portion of a file.

DOS normally returns an error indication in the AL register if the carry bit is set and if a DOS function fails to perform as requested. DOS 3.X can return extended error information upon request by using the Get Extended Error function (59 hex). The BX register should be set to zero and the results come back in the AX register with a suggested action indicator in BL. The error class is returned in BH.

Two new file creation functions have been added: Create File (5A hex) and Create New File (5B hex). The create unique file requires a path name, referenced by DS:DX, which ends with a backslash, '\'. The new file attribute is passed in CX. A new file is created which did not exist previously in the directory specified by the path name. The file should be deleted after it is closed. The path name is modified to reflect the name of the file created. An error occurs if the file cannot be created.

The create new file function uses the same parameters as the normal UNIX-file create function except that the function will fail if the file specified already exists. The normal function deletes any existing file before creating a new file.

File locking (5C hex) has a byte level granularity. Parameters include:

Register	Description
AL	0 lock/1 unlock
BX	file handle
CX:DX	offset
SI:DI	length

The file handle is that returned by an open or create operation. The offset is the logical byte offset to the start of the region to be locked or unlocked. The length is the number of bytes involved in the operation. Any portion or the entire file can be locked by one or more programs. An error code is returned in the AX register with the carry bit set if an error occurs.

The machine and printer status is supported by the DOS function 5E hex. The AL register is used to specify a subfunction. For example, 5E00 specifies the get machine name operation. The results are returned in CH (0 if name not defined and non-zero if defined), CL is the NETBIOS name number, AX contains the error code if the carry flag is set, and the machine name is copied into the buffer referenced by DS:DX. The buffer should be 128 bytes long.

The DOS functions 5E02 and 5E03 hex invoke the set and get printer setup functions. The printer setup string is sent to the printer when the printer file is opened by an application. The Set operation takes the redirection list index in BX, the length of the setup string in CX and the string is referenced by DS:SI. The get printer setup is the same except the setup string is copied into ES:DI. The buffer should be at least 128 bytes long when getting the printer setup string.

Network support is provided through redirection of the standard DOS devices such as the printer and disk drives. The three redirection functions are: Get Redirection List (5F02 hex), Redirect Device (5F03 hex), and Cancel Redirection (5F04 hex). These functions only operate if a DOS redirector, normally part of the network support, is loaded and active.

The redirection list is the logical to physical name translation table used to specify the devices being translated and their physical counterpart. The contents of the redirection list are accessed sequentially using the Get Redirection List function (5F02 hex). The buffers for the logical and physical device names are referenced by the DS:SI and ES:DI registers respectively. The list index is passed in the AX register. The list index is a zero or a positive integer. The function should be called starting with a zero parameter and incremented until the carry bit is set. The AX register returns with a value of 1 if there is no NETBIOS, the redirector, and 18 hex if the list index is past the end of the redirection list. The BH register contains a zero if the device is valid and 1 if the device is invalid. The BL register contains the device type (3 for a printer and 4 for a logical disk). CX contains a stored parameter which can be passed when initializing the redirection. The ES, DS, SI and DI register remain unchanged.

Device redirection is done using the Redirect Device function (5F03 hex). The device type, mentioned in the last paragraph, is passed in the BL register. The logical device name is passed in DS:SI. This may be a null terminated string if the operation is simply to verify the ability to access the device. The ES:DI should point to the network device name which is terminated by a zero. This should be immediately be followed by an optional password and a second terminating zero. The carry bit is set and AX contains the error code if the network device is not on the network.

Redirection of a device is cancelled using the Cancel Redirection function (5F04 hex). DS:SI should reference the logical device or path name of the redirected device. The device name is removed from the redirection list if it is in the table.

A Program Segment Prefix (PSP), is associated with each program. It contains various application parameters. The value is normally available to an application which was started by DOS but may be inaccessible. The PSP function provides access from any point within the program. There is an undocumented way to set the PSP for an application. This is often used by RAM-resident software and operating environments to change the logical application which DOS thinks is running. All memory allocation blocks and file handles are PSP specific. The Get PSP function (62 hex) returns the PSP segment number in BX.

SUMMARY

DOS 3.X represents increased functionality in the area of redirection and networks. It also provides better disk support than DOS 2.X because of the 16-bit FAT support. The major cost is in terms of memory, which is minor, and the hassle of moving to a new version of DOS.

My next article will address Microsoft Windows running under DOS. This graphics based front-end supports a program and a user interface which is an enhancement of that found under DOS 3.X. μ

Bill Wong is the President of Logic Fusion, Inc., 1333 Moon Drive, Yardley, PA 19067, a systems software development firm.

MSO

MICRO SUPPLY ORGANIZATION INC.

dare to compare.

Don't be fooled. No hidden cost!

Complete MS-DOS/CPM Super Turbo

In keeping with industry trends MSO is bringing our customers high performance P.C. compatibles and accessories. MSO takes the P.C. compatible to maximum performance with its SUPER TURBO. The SUPER TURBO features the V20-8 chip which runs at three times the speed of the IBM-PC XT* and also runs CPM 8080 software.

The SUPER TURBO comes complete with the MS-DOS operating system, Read and Run CPM, full Instructional Documentation, Utility software, **plus for our first 100 customers MSO is offering the Micropro Wordstar Professional Software package. This package includes: Wordstar, Mailmerge, Correctstar, Starindex, Datastar and a G.L. Accounting System.** The SUPER TURBO is a complete turnkey system with everything necessary to plug in and operate.

USER SUPPORT HOT LINE

805/393-2247

All systems carry full 90 day warranty.

CASH PRICE ONLY

Check in advance. Add 3% for VISA/MC. Shipping & handling charges will be added to each order.

For our catalog with complete details and prices, send $2.00 to:

Micro Supply Organization, Inc.
4909 Stockdale Hwy. #180
Bakersfield, CA 93309

MON. - FRI. 7am - 5pm PST•SAT. 9am - 5pm PST

*IBM is a registered trademark of the IBM Corporation.

15% restocking on returned orders

Super Turbo
Super Price:

$1299

The Super Turbo P.C. runs IBM software and CPM 8080 programs

- CPU - V20-8 8mHz Super Chip runs *IBM compatible software at 3 times the speed of the IBM-XT and CP/M 8080 software.
- 8087 Math Processor optional
- 256K RAM on mother board expandable to 640K
- ROM 8K Bios
- 6 empty slots for expansion
- 2 serial port one optional with expansion kit
- 1 parallel port
- 1 game port
- Clock calendar with software
- Hi-Res monographics video board
- Floppy controller
- Dual Floppy Drives 360K ea.
- 135 watt XT Power Supply
- 5150 style compatible keyboard
- Hi-Res TTL Green or Amber 12″ monitor
- MS-DOS operating system and manual.
- Instructional Documentation and Utility Software
- Assembled and tested in U.S.A.
- **Optional internal 20 meg sub system for Super Turbo add $499**

Special printer pricing with purchase of above computer.

MSO

MICRO SUPPLY ORGANIZATION INC.

the giveaway.

MSO, the leader in cost effective computer systems, is offering the Sanyo MBC 1200 at a **giveaway price!**

The MBC 1200 Accounting Software -G/L, A/R, A/P, Inventory and Payroll modules retail for $398 per module. **Now from MSO you pay only $99 per module or all five for $299!** Second disk drive with fan-option available. **COMPAT** disk utility available.

SPECIFICATIONS:

- Two Z-80A (main and subsidiary) CPUs with no-wait mode for fast execution, substantial memory capacity (RAM 64KB, ROM 4KB).
- High-resolution full graphic function with 640 x 400 dot matrix display.
- Choice of 33 or 40 line text mode.
- CP/M operating system complete with interpreter, editor and all utilities.
- Easy-to-use Sanyo graphic BASIC.
- One (MBC 1200) or two (MBC 1250) internal double-sided, double-density, double-track, 5¼" slim-type 640KB formatted mini floppy disk drives.
- Special design featuring detachable ergonomic keyboard with coiled cable.
- Interfaces for Centronics printer and one RS-232C port provided.

20 meg internal hard drive sub-system for Sanyo MBC 1100, 1160 & 1200 $599 (with purchase)

Printer specials with purchase

805/393-2247

All systems carry full 90 day warranty.

CASH PRICE ONLY

Check in advance. Add 3% for VISA/MC. Shipping & handling charges will be added to each order.

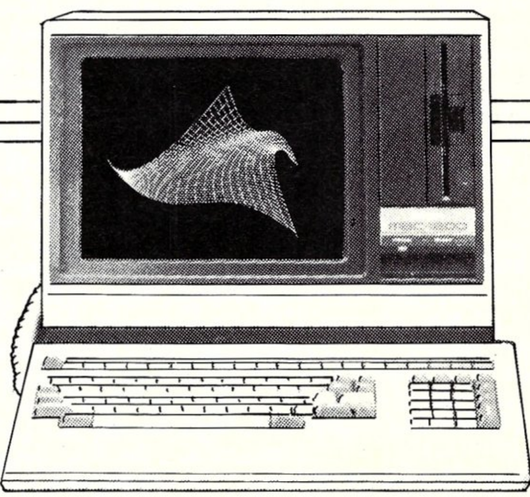
For our catalog with complete details and prices, send $2 to:

Micro Supply Organization, Inc.
4909 Stockdale Hwy. #180
Bakersfield, CA 93309

VISA MasterCard

15% Restocking on Returned Orders

SANYO MBC 1200

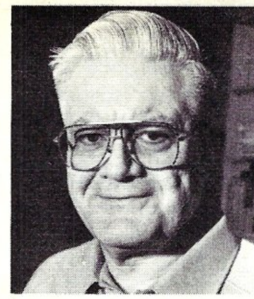


FREE SOFTWARE

- Wordstar • Spellstar
- Calcstar • Mailmerge
- Infostar • Basic

$399

The Scientific Computer User



by A. G. W. Cameron

MathCAD, Plot Packages, and Updated Ratfor

MATHCAD

MathCAD ($189, MathSoft, Inc.) is a kind of "electronic scratchpad" for scientists and engineers. Avram Tetewski gives a detailed description of the program in this issue. He based his review on his experience with Alpha and early Beta versions of the program. Here I supplement his remarks based on my experience with the released version 1.0.

MathCAD is based on a very promising concept, but its present implementation suffers from a poor user interface. The program would have benefitted by a few more months of development, but evidently MathSoft decided to get it out on the street in a form that would be useful to many groups of people. Although the following remarks are primarily critical, they are meant to be constructive, because I would like to use MathCAD when the bulk of the following problems are corrected.

In MathCAD you work with three different kinds of regions: text, equations, and plots. The major type of region that is missing is the array, or table, type of region. MathCAD does not yet support matrix operations. It does support one-dimensional arrays in the form of range variables, but you cannot output a region containing a tabulation of the values of a dependent variable unless you type in all of the individual elements and ask for each of their values.

It is useful to think of MathCAD as an extended text editor which has been enhanced in capability through the addition of the equation and plotting functions. The editor should be powerful and convenient, but here the functions are awkward and sometimes frustrating. Because of the equations and plots, MathCAD requires a graphics controller of some kind (I used a Hercules monochrome controller). Painting a screen is slow compared to the display of text by an editor that writes directly to the screen, and thus it is very important that graphics display algorithms be very efficient. For example, consider the instant command to create a blank line at the cursor. If I want to open up several blank lines, the screen is redrawn (slowly) after each command. This is very frustrating.

You can append one MathCAD document to another. But you cannot *insert* one such document inside another. Insertion can only be achieved by appending and then cutting and pasting individual regions from one side of the new material to the other; there is no way to cut and paste a section composed of several regions all at one time.

You must learn several conventions in order to be able to enter equations properly, but this is not too difficult. Equations are displayed approximately the way they would look if you were writing them down on paper, and the position where the next symbol should be entered is usually indicated by a square dot acting as a placeholder. However, fraction numerators and denominators and superscripts and subscripts are displaced vertically by a full line. It would have given a more natural appearance to the equation to have made these displacements half a line. As you develop an equation, it may overlap on another region. There is an instant command to separate such overlapping regions, but that separation should occur automatically.

There are two kinds of assignment operations. The local assignment operator, ":=", establishes a relationship that is supposed to apply from that point onward in the document, and the global assignment operator, "≡", applies everywhere throughout the document. There is presently a bug in the way these operate. I appended the CGS.MCD file to a document I was working on; this contains relationships between many different units of time, distance, mass, and charge, all defined globally. Nevertheless, when I tried to use these units in the formula I was developing, the program persisted in telling me that they were undefined. MathSoft told me that that these global definitions do not take effect until they are scrolled onto the screen and a recalculation takes place while they are there. They promise a fix in the next release.

Variables may have single or multiple values; the latter are called subscripted variables, as in f_j, where j is a range variable. This notation prevents you from using variables with ordinary subscripts, which is a common scientific notation. I would much prefer a more conventional notation like $f_e(j)$. When you declare a range for the variable j, the program will automatically calculate the value of a subscripted expression for all values of j, but I was unable to define a series of such subscripted variables in which each depended on its predecessors; this is a serious deficiency.

In contrast, setting up plots is a relatively pleasant experience. Typing in the "@" symbol creates a small plot with placeholders for the quantities to be plotted on the abscissa and ordinate as well as starting and ending values for variables on the axes. When these are typed in the plot is drawn. Manipulating the size and position and putting in grid lines are simple operations. I missed the ability to add additional numeric labels to some of the grid lines.

When I hack a small Fortran program to serve as a "scientific scratchpad", I frequently do a derivation of the formula that I want to use, and I do it right on the screen using comment symbols. That way I can see how I got my formula if I want to write something about the result later. In MathCAD there is no way to "comment out" an equation region or to write an equation in a text region, either of which would have served the same purpose.

MathCAD serves such an important need that apparently it has already obtained a following who are willing to put up with the deficiencies of the user interface. I am looking forward to to the next major release in which MathSoft promises to have cured many of the problems that I have outlined.

INTERACTIVE PLOTTING

I have taken a look at two interactive plotting programs intended for scientists and engineers: PLOTZ GRAPHICS (copy-protected, $350 from Curtis Technical Software) and SCI-GRAF (not copy-protected, $99.95 from Microcomputer Systems Consultants; the FONTEDIT font editor is included for an extra $30). PLOTZ displays plots on a screen using IBM standard monochrome or color graphics (but not EGA) and Hercules; SCI-GRAPH displays on IBM-compatible 640 × 200 pixel monitors only. The hard copy output from PLOTZ goes to pen plotters using the Hewlett-Packard HP-

GL command format only. SCI-GRAPH gives hard copy output only on Epson-compatible dot-matrix printers, including the JX-80 color dot-matrix printer.

In the use of an interactive plotting package, the user gains a (hopefully) easy-to-use interface at the expense of some restrictions on the generality of the plots that can be produced. Each of these packages has an easy-to-use interface and each has made various compromises with the generality that you could obtain by using a graphics library attached to a higher-level-language compiler, or by writing a PostScript program.

PLOTZ is operated through a series of menus and submenus displayed on the screen. In these menus ten outline boxes appear at the left of the screen and are labelled to represent the ten keyboard function keys; some or all of the keys in each menu are also labelled with a function name that is executed when that particular key is pressed. The right part of the screen usually shows the values of a fairly large number of parameters. Sometimes all of these values must be selected from function keys; when a function key is pressed that corresponds to a particular parameter, the cursor moves to the right of that parameter and the entered text is displayed as its value. If only a choice must be made among several discrete values of a parameter, then generally you are presented with a submenu having the available values associated with the function keys, so that pressing one of the keys transfers that value to the right of the parameter. I use the term "value" here in a very general sense; it might be a particular color to associate with an axis or a label. The two main major menus are Data Operations and Plotting Operations.

The principal Data Operations are fetching and storing data and entering data from the keyboard. Data may also be edited, linearly scaled, linearly fitted, smoothed, fitted by a polynomial, differentiated, and fitted by a histogram. Plotting Operations specify the appearance of a plot and do the plotting. There is a long list of appearance options that can be specified, including the positioning of axes, the numerical ranges along the axes and the labelling of them, the choice of line types to be drawn, and the choice of symbols. There is an art to using a pen plotter; hence the program lets you specify the pen speed within a continuum ranging from fast to slow. You can also choose whether to plot on the plotter or on the screen. You can add text, symbols, and arrows to the plot, but only on the hard copy, not on the screen. I consider this last restriction to be unfortunate.

SCI-GRAF assumes that your data has been prepared for plotting by separate procedures, so its menus establish the appearance of the plot. Actually, these are not really menus, but rather a series of questions that must be answered by entering text from the keyboard. In the case of multiple choice questions, usually it is only necessary to enter the first letter of the answer. There are usually reasonable default values for many of the parameters, so that much of the menu can be skipped if desired.

Both packages allow you to store the plotting parameters in batch files, so that the same graph layout can be used with different sets of data.

The PLOTZ user interface is superb. Because the program uses a pen plotter, it draws vectors. You are given no choice of letters; you must stick to a bold sans-serif letter style and you have no choice about the letter size that will be used. Thus you are quite out of luck if you want to use Greek or mathematical symbols or a different font style. While you have a choice of several line styles, you have no choice of line thickness, except by your independent choice of the pen nib.

The user interface for SCI-GRAF is good; it has a smaller job to do. However, the program does not use the full capability of the Epson FX-series dot-matrix printers. It produces a rasterized plot, but it only gives 144 pixels per inch in the vertical direction and 120 pixels per inch horizontally for "enhanced print", and half of each of these values for "draft print". Thus even the enhanced print plots are not what I consider to be of publication quality. I consider SCI-GRAF to be mainly useful for producing working plots. The only program I know that produces publication quality plots with an FX-series printer is SciPlot ($59.95 from MicroGlyph Systems), which uses the full 216 pixel vertical by 240 pixel horizontal resolution which is available (at a corresponding cost in plot time, but usually shorter than a pen plot time). I reviewed this program in the July-August, 1986, issue of *Micro/Systems Journal*; it is a Fortran rather than an interactive package.

SCI-GRAF uses three different character sizes, small, medium, and large. These are rasterized fonts, and so can only be presented vertically or horizontally. The optional FONTEDIT program allows these fonts to be modified and other characters to be designed and substituted. This approach again is not as good as that used by SciPlot, which uses many fonts selected from the Hershey character set of vectorized symbols; SciPlot plots them as vectors which can thus have any orientation or size desired.

RATFOR UPDATED

Ratfor ("Rational Fortran") was originally devised to allow one to write a structured coding language in the style of C, and to translate this code into standard Fortran statements, accompanied by lots of "GOTO's". I reviewed the Software Tools version of Ratfor in *M/SJ*'s predecessor magazine (*Microsystems*, September, 1983), and hence I will be brief here. That Ratfor produced Fortran-66 code. Fortran-77 introduced the IF-THEN-ELSE-ENDIF control structure, and this made it desirable for Ratfor to produce Fortran-77 code. RF77 ($65 from Logical Developments) is a new version of Ratfor that does this.

RF77 uses C syntax in several control structures. These include WHILE, REPEAT-UNTIL, and a generalized version of the FOR statement. DO loops do not take statement numbers. Additional statements that can be used inside block control structures are BREAK, LOOP, and NEXT. There are also #include and #define directives, although the latter is less powerful than in the old Ratfor.

Potential users must decide for themselves whether they would benefit from the use of this preprocessor. The output Fortran code is not easily readable, principally because RF77 omits unnecessary spaces (except for a space after a GOTO). This can be bothersome when one traces down errors in the Fortran compiler, and it can be bothersome when one wishes to transport the Fortran to another computer. But it can be a boon to people who like to use the control structures in C or Algol-like languages. µ

A. G. W. Cameron is Professor of Astronomy at the Harvard-Smithsonian Center for Astrophysics, 60 Garden Street, Cambridge, MA 02138.

Addresses

Curtis Technical Software Corp., P. O. Box 178, Pennington, NJ 08534; (609) 737-8844.

Logical Developments, P. O. Box 55798, Houston, TX 77255; (800) 835-2246, ext 41.

MathSoft, Inc., One Kendall Square, Bldg. 100, Cambridge, MA 02139; (617) 577-1017, out of state (800) 628-9247.

Microcomputer Systems Consultants, 32 West Anapuma, Suite 190, Santa Barbara, CA 93101; (805) 966-9247.

MicroGlyph Systems, P. O. Box 1066, East Arlington MA 02174.

WATCOM Systems Inc., 415 Philip Street, Waterloo, Ontario, Canada N2L 3X2; (519) 886-3700.

AS WE GO TO PRESS

Microcomputer Systems Consultants has informed us that the current version of SCI-GRAF (1.1, released in July) now supports the Hercules graphics board and Epson, C.Itoh, NEC and IBM dot-matrix printers.

MathCAD PUBLISHER RESPONDS

Continued on page 78

MathSoft's MathCAD

by Avram K. Tetewsky

MathCAD is a wordprocessor-style tool for MS-DOS based systems that can handle dual window operation, execute typeset-style equations with real and complex numbers, manipulate, convert and check fundamental units of mass, length, time and charge, as well as handling derived units, and embedded graphics within a document. MathCAD works with CGA, Hercules, and EGA cards, with or without 8087, and requires a minimum of 384K.

After invoking MathCAD, you are presented with a wordprocessing-like screen. By typing a quote ("), MathCad understands that what follows is text; anything else is taken as a variable or equation. As you enter equations, MathCAD automatically reformats them so that square-roots, fractions, and nested super and subscripts automatically display in WYSIWYG form.

MathCAD understands three types of equations: one that is a global definition (an equal sign made up of 3 horizontal lines); an equation that is a definition (the Pascal/Ada := symbol); and an equal sign (=) which forces an evaluation. An approximate rendition of MathCAD's capability is shown in Figure 1. By repeating the equations, other evaluations can be done. MathCAD also understands subscripts and do-loop like constructs, as shown in Figure 2.

You can also define functions, as well as make use of a wide range of intrinsic functions. When it comes to plots, MathCAD allows you to define each axis with an

A Scientific Calculator Wordprocessor

equation. You then have the option of setting the grid, size, and plot type. Plots, within the text can be of any size. For plots larger than a page, MathCAD simply prints page-sized sections so you can manually tape them together. Note that the first release of MathCAD does not paginate.

MathCAD treats each new equation or text region as an isolated region. Once defined, you can jump to each region to make further modifications.

The first release of MathCAD is targeted at non-matrix engineering and scientific applications, but quarterly releases are planned in order to cover increasing technical needs. Integration is not defined, you have to define numerical integration rules. MathCAD's first release has: a

EDITOR'S NOTE

Avram's review of MathCADwas based on a pre-release version. Some comments on the final released version of the program program are contained in "The Scientific User" column written by A. G. Cameron.

DOS shell command, one-key insert and delete line, and the ability to read/write ASCII data from foreign files. The price is $189.00 and it is not copy-protected. Planned improvements are to add linkage to other languages (C, FORTRAN, Pascal), matrix operators, IF-THEN-ELSE-ENDIF logic, more Greek/math symbols, numerical differentiation and integration and simple backsolving. MathCAD is written in Microsoft C, and should be linkable with other Microsoft languages.

MathCAD has the potential of being a product that could be the primary interface to all other CAD/CAM packages. Consider first that all planning documents could now serve as test documents. When a problem/program is specified, it can be executed and tested by MathCAD. Because MathCAD stores documents in ASCII form (doing all calculations on the fly), it should be possible to transport documents among other systems. Consider goverment B5 and C5 documents. B5's are system specifications and C5's document the specific implementation. MathCAD could add a whole new factor of reliability and utility to the B5/C5 process for both DoD and commerical ventures. MathCAD can also simplify many other scratchpad calculations, i.e. MathCAD is like a hybrid of APL and a wordprocessor that also understands units. Teachers could type many of their homeworks with this and white-out the answers.

Although MathCAD excels in its equation capability, it does lack some features in

```
km   =  1L             global definition of units.

sec  ≡  1T          3
               5  km
MU   ≡  3.98*10 * ---  global definition of the
                    2  gravitational constant
                  sec

A    :=  10.00000*km   temporary assignment
e    :=  0.            of the semi-major axis
                       and eccentricity.

         MU    1 - e      0.5
VMIN :=  ( — ( ----- ) )
         A     1 + e
                                 -1
V    =  199.4993734 length * time      evaluation
```

Figure 1.

```
You type          You see
--------          -------

j: 1,1.5;5        j := 1,1.5..5  1 to 5 in .5
                                 increments
                            2
X[j : j^2+1       X  :=  j    + 1
                   j

where X is now a vector with each of its 5
components defined as an equation.
```

Figure 2.

its wordprocessing and Greek display ability. Because MathCAD is an important product, I felt it was necessary to outline its current strengths and weakness. A flow chart is shown in Figure 3, presented in an order that I find desirable. The manual is good, and has indices by key, alphabet, and function.

There are several problems with this initial version. Although most of the equations worked as advertised, some of the keys are strangely placed. For example, [F2] being a quit is placed next to the help [F1]. Although it prompts for Y/N before quiting, quit should be elsewhere. ESC should be the abort, and say make [F2] the toggle between the command line and text window. It would also be nice to have some type of page-breaking display function.

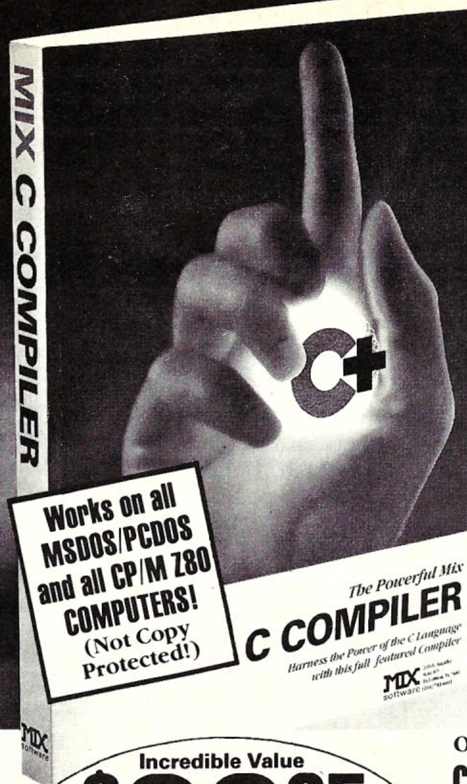
MathCAD is definitely worth a try and a demo disk is available for only $10.

Product Information
MATHCAD
$189 (demo disk $10)
MathSoft
1 Kendall Square, Bldg 100
Cambridge MA 02139
800-628-4223 (617)577-1017

Avram K. Tetewsky has BSEE and MSEE degrees from RPI and MIT. He holds an Intern Engineers License and FCC First Class Radio/Telephone License. He is currently a navigation/communication analyst at the Charles Stark Draper Laboratory. He also writes the PC Technical Column which appears in Boston Computer Society PC Report. Avram's interests include science and engineering applications. His hobbies include music, cats, swimming and biking. Note that the opinions expressed are his own, and do not reflect the view of the Charles Stark Draper Laboratories.

Figure 3.

```
A>MCAD          (note CNTL A is a general abort
                      CNTL R repaints the screen)

1) cursor sits in text region by default

  1.1  type equations or text.
       you type                you see
       ---------               ---------

           ~                    ≡   global definition
           :                   :=   equation/function
           =                    =   numerical eval
          X[J                   X   subscript
                                 J
                                 J
          X^J                   X   superscript (exponent)
          ()                  left or right paranthesis
           '                  (  ) pair of
                                   paranthesis
           i                   complex number
        ALT A,B,...           IBM Greek Characters
          "                  start text, hit return to
                              mark right margin for word wrap.
        o,h                   octal/hex suffix
        L,M,T,Q               length, mass, time and charge
                              fundamental unit suffixes.
  1.2  Movement of the cursor
        arrow keys
        Tab                   move by one variable
                              within a region.
        Shift Tab             move back by one variable
        PgUp/PgDn             move by 10 lines
        CTRL arrows           move left/right by 10 cols
        HOME/END              move forward/back by 1
                              region
        CTRL HOME/END         move to Top/Bottom of
                              document
        BACKSPACE             delete character
        INS                   toggle insert/append mode

   1.3 Command Line Commands (hit ESC to go to command line)

        SAVE filename   [F6]
        LOAD filename   [F5]
        QUIT            [F2]
        CALCULATE       [F9]
        PROCESS
        WIDTH                   set text width
        MARK          \
        TEXTCUT        >        cut & paste
        TESTPASTE     /
        FILENAME
        INSERTLINE      [F10]
        DELETELINE      [Ctrl][F10]
        GOTO row column
        WIDTH number
        CLEAR                    clear the document
        APPEND filename
        AUTOMATIC                goto automatic mode
        MANUAL                   goto manual mode
        HELP            [F1]     help screen
        PRINT
        DOS
        ... other commands listed in alphabetical order
        FORMAT
        MARK
        MEMORY                 current version can't
                                  page off of disk.
        SEPARATE               separate overlapping reions
        MOVE
        PLOT

        PRINT
        SPLIT/UNSPLIT   [F7]/[CNTL F7] windows
        SWITCH          [F8]           windows

  1.4) Predefined Function Keys/ cntl keys

        F1    F2   help    quit
        F3    F4   cut     paste
        F5    F6   load    save
        F7    F8   split   switch       CNTL F7 unsplit
        F9    F10  calc    delete line  CNTL F10 delete line
          CNTL A      general abort
          CNTL R      refresh screen
          CNTL S      separate overlapping displays

2)    Functions and Equations

      2.1  variables (upper and lowercase are unique).
           following are valid names.

           ALPHA   as many characters as needed.
           B_1     underscores allowed.

      2.1  operators, built-in functions (real and
           complex for all operators), and ranges

           what you type    what you see     operator
           -------------    ------------     ---------
           x!                  x!            factorial
           \x                                square-root
           x^y                               power
           |x                  |x|           absolute val
           x/y                               division
           X*Y                 X.Y           multiply
           +,-                               add/sub
           X$Y                               summation
           sinh,cosh,tanh,
           asinh,acosh,atanh,
           sin,cos,tan,
           asin,acos,atan,
           exp,ln,log,
           re,im,
           rnd,
           PHI(X)   step function
           Bessel functions
           ceil, floor, mod, size, until
           var: s,n;e       var := s,n..e    start,
                                             nextval,
                                             end

                                             next is optional
                                             and if ommitted
                                             is either s+1 or
                                             s-1, as needed

          my_function(argl,..) :=            define your own

3)    Cut and Paste Features (equations and text)

 ESC MARK       mark beginning of block at current
                cursor location.
 ESC CUT/COPY   mark block end at current cursor location.
                Block disappears.  Use COPY instead of CUT
                if you do not want text to disappear.
 ESC PASTE      paste in block at current location.

4)    Margins
      Margins are taken as first cursor position and first
      return of each region.  Rewrap is automatic.
```



C FEVER *catch it!*

It's becoming an epidemic . . . everyone is switching to C! First there were a few hackers, then came the college students, next the major software houses, and now the rest of the programming world. Programmers everywhere are infected with the desire for SPEED, POWER, and PORTABILITY.

It's time to face the inevitable. You're going to catch the fever too! When you do, give us a call. We've got the best cure — an illustrated guide to the C language, plus a complete program development system. Everything you need to master the C programming language . . . all at a price that's less than the cost of a book!

But don't let this price fool you. Our system is powerful; it compiles twice as fast as the others, is completely standard, and it's very easy to use. Most C compilers are designed for wizards. We have designed ours for you!

What do you get for a mere $39.95?

- **A 450 Page book filled with sample programs, plus . . .**
- **A fast, standard, full featured C compiler** that supports all data types and the latest features like bit fields, enumerations, structure assignment, and passing/returning structures.
- **A fast linker** that loads separately compiled files, searches libraries, and builds an executable program.
- **An extensive library of more than 170 functions** (including the standard C functions and the computer specific functions that provide direct access to the operating system and BIOS).
- **Tools that allow you to optimize** your programs for minimal space or maximum speed.

Incredible Value AT ONLY $39.95 With 30 Day Money-Back Guarantee

Operators are standing by . . . Please use this Number for ORDERS ONLY!

CALL TOLL FREE FOR RUSH ORDER DELIVERY!

1-800-523-9520

IN TEXAS, PLEASE CALL TOLL FREE 1-800-622-4070

For Technical Support Please call ***1-214-783-6001***

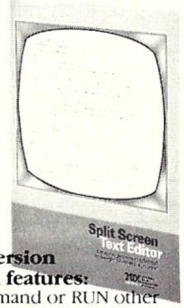
MIX Software, Inc. / 2116 E. Arapaho / Suite 363 / Richardson, Texas 75081

Or contact our Worldwide Distributors direct in:

Canada:	Saraguay Software	1-416-923-1500	Switzerland:	DMB Communication	CH-1-825-53-29
Australia:	Techflow	047-586924	France:	Info/Tech	1-43-44-06-48

Split Screen Text Editor

an Incredible Value AT ONLY $29.95 *(Not Copy Protected!) Works on all MSDOS/PCDOS and CP/M Z80 Computers*



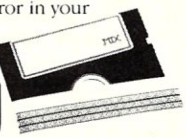
Our high powered editor is great for editing high level languages. **It works just like Micropro's Wordstar**™ but macros allow you to create your own custom editor, and the split-screen feature lets you edit two files at the same time.

The MSDOS/PCDOS version is loaded with special features:

- Execute any DOS command or RUN other programs from the editor.
- Quickly edit files as large as 300,000 characters.
- Compile MIX C programs directly from memory. The editor automatically positions the cursor to the first error in your program.

ASM UTILITY

an Incredible Value AT ONLY $10



Call assembly language routines from your C programs. The ASM utility works with Microsoft's MASM or M80 assembler. Macros make it easy! Works just as if you were calling a C function, and you can even call C functions from assembly language. Lots of useful assembly language functions are included as examples.

RUSH REPLY ORDER FORM!

SAVE $14.95 Off Our Regular List Price!

Limited Time Only **$54.95** C Compiler & Text Editor

Description	Quantity	PRICE	Total Order
Split-Screen Text Editor	______	$29.95	$______
C Compiler	______	$39.95	$______
C and Text Editor (Special)	______	$54.95	$______
ASM Utility	______	$10.00	$______
Texas Residents Add 6.125% Sales TAX			$
Shipping Charges (See at Right)			$
TOTAL OF YOUR ORDER:			$

Please check method of payment:
☐ Check ☐ Money Order ☐ MasterCard/VISA

Your Card #: ____________

Expires ____________

Shipping Charges: (No charge for ASM Utility)
In the U.S.A.: Add $5.00 per Order.
In CANADA: Add $10.00 per Order.
OVERSEAS: Add $10.00 per Text Editor. Add $20.00 per C Compiler. Add $30.00 for combined C Compiler and Text Editor.

Operating System: ***(Check one)***
☐ CP/M Z80 ☐ MSDOS/PCDOS

Specify Your Computer Name ____________

Specify Disk Format ____________

NAME ____________

Telephone A/C (______) ____________

Street ____________

City ____________

State ____________

Country ____________ ZIP ____________

MIX software
2116 East Arapaho
Suite 363
Richardson, Texas, 75081

Ask about our Volume Discounts!
Call 1-214-783-6001

M

Concurrent DOS

by Alex K.H. Soya

Part 2 Processes and Their Data Structures

The first part of this article (M/SJ, May/June) examined the differences between DOS and Concurrent CP/M-86 (also known as Concurrent P/) from the users' point of view and also presented most of the modules which make up the operating system.

This part discusses how Concurrent DOS keeps track of multiple processes and how the system programmer can access the data structures associated with processes.

SYSTEM & APPLICATION PROCESSES

In a concurrent environment, we can distinguish two categories of processes: system and application processes. System processes are tasks which serve the operating system and its environment in some way. While system processes can gain access to certain structures (such as privileged queues) in the operating system, application processes may not be granted such access.

Normally a system process is incorporated into the operating system at generation time by means of RSP (Resident System Process) files. However, it is also possible for a program, which has been loaded from disk at any time, to become a system process.

Typically three major system processes can be found under Concurrent DOS: The Physical INput process (PIN) which is responsible for obtaining characters from the keyboards; the CLOCK process which updates the status lines at regular intervals; and the Terminal Message Process (TMP) which interacts with the user at the system prompt level. Other system processes may be added in particular OEM implementations or configurations.

Applications are usually transient programs that are loaded and executed as required by the user. Under certain conditions, such as in a dedicated system, applications can be incorporated into the operating system as Resident System Processes (RSP). However, such situations are rare and usually not found in user oriented configurations of Concurrent DOS.

WHAT HOLDS IT ALL TOGETHER

Concurrent DOS contains data structures describing the state of the operating system, process information, console states, memory descriptors, and many other items. One structure, the SYSDAT table, can be considered as the glue between the various operating system modules, processes, and structures. A programmer who wishes to write programs which perform more sophisticated actions relating to the operating system must be familiar with the contents and meaning of the various fields in the SYSDAT table.

Figure 1 shows the layout of the SYSDAT table and the various entries. The address of the SYSDAT table itself can be obtained by using the S_SYSDAT function call (number 154). The SYSDAT segment is returned in the ES and the offset in the AX register. The table usually starts at offset 0000h in the SYSDAT segment, thus the AX register will contain a zero on return from S_SYSDAT. Programmers would be

00H	SUP ENTRY				RESERVED			
08H	RESERVED							
10H	RESERVED							
18H	RESERVED							
20H	RESERVED							
28H	XIOS ENTRY				XIOS INIT			
30H	RESERVED							
38H	DISPATCHER				PDISP			
40H	CCPMSEG		RSPSEG		ENDSEG		RESER -VED	NVCNS
48H	NLCB	NCCB	N- FLAGS	SYS- DISK	MMP		RESER -VED	DAY FILE
50H	TEMP DISK	TICKS /SEC	LUL		CCB		FLAGS	
58H	MDUL		MFL		PUL		QUL	
60H	QMAU							
68H	RLR		DLR		DRL		PLR	
70H	RESERVED		THRDRT		QLR		MAL	
78H	VERSION		VERNUM		CCPMVERNUM		TOD_DAY	
80H	TOD _HR	TOD _MIN	TOD _SEC	NCON DEV	NLST DEV	NCIO DEV	LCB	
88H	OPEN_FILE		LOCK_ MAX	OPEN_ MAX	OWNER_8087		RESERVED	
90H	RESERVED							
98H	RESERVED							XPCNS
A0H	OFF_8087		SEG_8087		RESERVED			

Figure 1. SYSDAT Table. (from "Concurrent CP/M Programmer's Guide")

Offset					
00h	LINK	THREAD	STAT	PRIOR	FLAG
08h	NAME				
10h	UDA	DISK	USER	RESERVED	MEM
18h	RESERVED				PARENT
20h	CNS	RESERVED	LIST	RESER	SFLAG
28h	RESERVED				

Figure 2. Process Descriptor

tempted to use the offsets to various fields in the SYSDAT table as absolute offsets when addressing. However, there is no guarantee that future versions will have the SYSDAT table at offset zero. To address a field in the SYSDAT table, the value returned by the AX register should be used as a base-index value and field offsets should thus be added to that value, even though in current versions of Concurrent DOS the base offset is always zero.

The values found in the SYSDAT table can be considered as being global to the entire operating system environment. Thus no matter which process looks at the entries in SYSDAT, the values found will be the same for all processes. As we shall see later, Concurrent DOS also contains data structures which hold values that are only valid for a particular process. For a detailed description of each field in the SYSDAT table refer to the *Concurrent CP/M Programmer's Guide* by Digital Research.

Many of the items found in the SYSDAT table describe configuration dependent parameters such as the number of virtual and physical consoles, list devices, system ticks/second, maximum number of open files per process, etc. Other items describe where to find structures which reflect the current state of the system including which processes are currently running, or waiting for a certain event to complete, as well as lists of unused memory areas, the state of the consoles, and the current owner of the 8087 math coprocessor.

PROCESSES DESCRIPTORS

As Concurrent DOS can run several processes simultaneously, the operating system must keep track of which tasks are running in the system and in what state these tasks are in. Two structures are used to describe the state of a particular process. For each process, Concurrent DOS keeps a PD (Process Descriptor) and a UDA (User Data Area). These two structures are updated every time a dispatch occurs. The PD (Figure 2) contains information of interest to the operating system about the process, while the UDA (Figure 3) reflects the state of the process when last dispatched.

All of the Process Descriptors in the system are contained in the SYSDAT segment and kept on singly linked lists (Figure 4). Most of the root pointers to the various lists are kept in the SYSDAT table described earlier. A process will be kept on two lists at all times.

One such list is always the Thread List which links all of the process descriptors in the system together. The root of the Thread list is at the THRDRT field in the SYSDAT table. Each process descriptor contains a THREAD field which points to another process or contains a 0000h if no more process descriptors follow. By traversing the Thread List, all of the processes in the system can be examined.

The second list on which a process is kept depends on the current state of the process. CDOS provides for four state dependent process lists with their roots in the SYSDAT table: The READY LIST for processes ready to run, the DELAY LIST for processes delayed for a set number of system ticks, the DISPATCHER READY LIST which is a temporary holding place for processes which have just been made ready to run, and the POLL LIST which contains processes that are polling a device. The root pointers to the above lists are kept in the SYSDAT table as the RLR, DLR, DRL, and PLR fields. Processes on one of the above lists are linked to each other by the LINK field in the PD. The LINK field will either contain the offset of the next Process Descriptor in the list, or a 0000h if the process is the last item on the list.

Other process-state lists have their roots in structures directly related to the state of the process. The READ QUEUE list and the WRITE QUEUE list have their root pointers in the particular QUEUE DESCRIPTOR that the process is trying to read or write

Offset						
00H	RESERVED	DMA OFFSET	RESERVED			
08H	RESERVED					
10H	RESERVED					
18H	RESERVED					
20H	AX	BX	CX	DX		
28H	DI	SI	BP	RESERVED		
30H	RESERVED	SP	RESERVED			
38H	INT 0	INT 1				
40H	RESERVED	INT 3				
48H	INT 4	RESERVED				
50H	CS	DS	ES	SS		
58H	INT 224	INT 225				
60H	RESERVED					
68H	USER SYSTEM STACK					6FH
.						
.						
F8H						FFH
100H	CW	SW	RESERVED			Optional 8087 Extension
.	RESERVED					
.	:					
.	RESERVED					
158H	RESERVED					15FH

Figure 3. UDA - User Data Area. The length of the UDA is 256 bytes (352 bytes if 8087 processing is required), and it must begin on a paragraph boundary. (from "Concurrent CP/M Programmer's Guide")

"How to protect your software by letting people copy it."

By Dick Erett, President of Software Security

Inventor and entrepreneur, Dick Erett, explains his company's view on the protection of intellectual property.

"A crucial point that even sophisticated software development companies and the trade press seem to be missing or ignoring is this:

Software protection must be understood to be a distinctively different concept from that commonly referred to as copy protection.

Fundamentally, software protection involves devising a method that prevents unauthorized use of a program, without restricting a legitimate user from making any number of additional copies or preventing program operation via hard disk or LANs.

Logic dictates that magnetic media can no more protect itself from misuse than a padlock can lock itself.

Software protection must reside outside the actual storage media. The technique can then be made as tamper proof as deemed necessary. If one is clever enough, patent law can be brought to bear on the method.

Software protection is at a crossroads and the choices are clear. You can give product away to a segment of the market, or take a stand against the theft of your intellectual property.

". . . giving your software away is fine . . ."

We strongly believe that giving your software away is fine, if you make the decision to do so. However, if the public's sense of ethics is determining company policy, then you are no longer in control.

We have patented a device that protects your software while allowing unlimited archival copies and uninhibited use of hard disks and LANs. The name of this product is The BLOCK™.

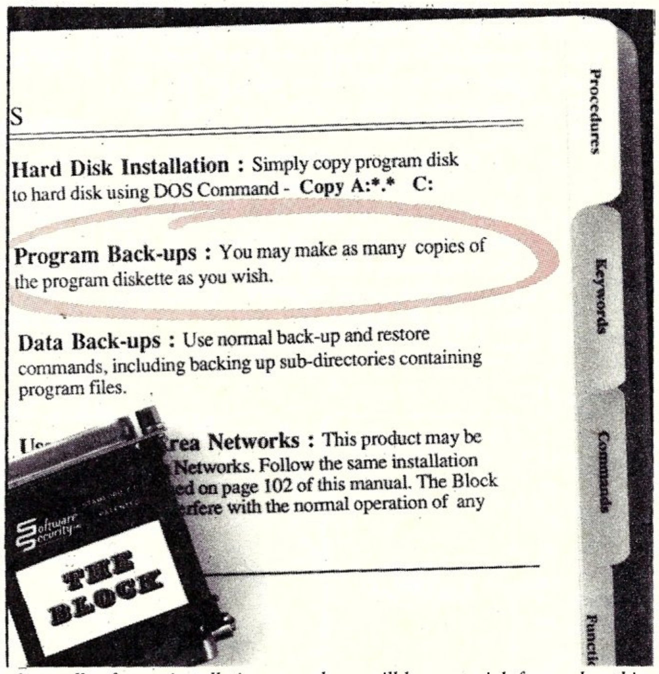
The BLOCK is the only patented method we know of to protect your investment. It answers all the complaints of reasonable people concerning software protection.

In reality, the only people who could object are those who would like the option of stealing your company's product.

". . . eliminating the rationale for copy-busting . . ."

Since The BLOCK allows a user to make unlimited archival copies the rationale for copy-busting programs is eliminated.

The BLOCK is fully protected by federal patent law rather than the less effective copyright statutes. The law clearly prohibits the production of work-alike devices to replace The BLOCK.



Soon all software installation procedures will be as straightforward as this. The only difference will be whether you include the option to steal your product or not.

The BLOCK attaches to any communications port of virtually any microcomputer. It comes with a unique customer product number programmed into the circuit.

The BLOCK is transparent to any device attached to the port. Once it is in place users are essentially unaware of its presence. The BLOCK may be daisy-chained to provide security for more than one software package.

Each software developer devises their own procedure for accessing The BLOCK to confirm a legitimate user. If it is not present, then the program can take appropriate action.

". . . possibilities . . . limited only by your imagination . . ."

The elegance of The BLOCK lies in its simplicity. Once you understand the principle of The BLOCK, hundreds of possibilities will manifest themselves, limited only by your imagination.

Your efforts, investments and intellectual property belong to you, and you have an obligation to protect them. Let us help you safeguard what's rightfully yours. Call today for our brochure, or a demo unit."

Software Security inc.

870 High Ridge Road Stamford, Connecticut 06905
203 329 8870

Circle **no. 170** on reader service card.

from (more about Queues in Part 3). A process which is waiting to attach to a character I/O device (console or printer), while another process owns the same device, will be kept on the CQUEUE LIST. The root of the latter can be found in yet another structure, the Character Control Block (CCB) of the requested device. The last state a process can be in is a FLAG WAIT. When a process has to wait for a particular device to complete its function the process will issue a flag wait request through the device driver contained in the XIOS module. Once the device completes the I/O it will issue an interrupt request and the corresponding system flag will be set by the interrupt service routine. While the process is waiting for the flag and thus is in the FLAG WAIT state, its process descriptor offset is placed in the flag table. The offset of the flag table can be found in the SYSDAT FLAGS field.

THE UDA

As the CPU has only one set of registers, these registers must be shared among all of the processes in the system. Each time a process is dispatched, all of the current CPU register contents must be saved. The next time the process is allocated to the CPU, all of the registers must be restored to the values at the time the CPU was released from the process.

To preserve the register contents, each CPU register is copied to the UDA of the active process every time the CPU is given to another task. Before the CPU continues executing the next task, the register contents of that task are copied from the UDA back to the individual CPU registers. In addition to the register contents the UDA also holds other values which are unique to the state of the process. The interrupt vectors 0,1,3,4,224, and 225 are considered to be unique to a process. A dedicated interrupt service routine for the above vectors can thus be set up individually for each process. At every process dispatch the interrupt vectors must also be updated with the values contained in the UDA. By default, the interrupt vectors of the parent process are inherited by a newly created task.

Another area in the UDA is used as the User System Stack. This stack area is distinct from the stack used by the process internally. The User System Stack is employed by the process while it is executing CDOS function calls. Once the function call has been completed, the process' internal stack is used again. This allows the programmer to set up his own stack requirements without having to consider the requirements of the operating system services.

If the process is allowed to access the 8087 numeric coprocessor, additional information must be kept about the current state of the 8087 for that process. As the 8087 is a resource which cannot be shared, all of its activity must be stored and restored during process dispatches.

THERE IS MORE

The *Concurrent CP/M Programmer's Guide* lists many of the fields in the various structures, such as the SYSDAT table, the Process Descriptor and the UDA, as reserved. This does not mean that these fields are currently unused. Altering the values in the reserved fields could cause catastrophic results and should thus be avoided.

Through experimentation, it is possible to determine the meaning of many of the reserved fields. While there is no guarantee that undocumented items in Concurrent DOS data structures will still be available in future versions, it may in certain cases allow programmers to *perform the impossible*. A future article may deal with this topic.

NOW FOR THE REAL THING

To illustrate access to the SYSDAT table, and the process descriptors I wrote the utility PANA (Process ANAlyzer) shown in Listing 1. The program is written in TURBO Pascal.

The program displays a list of all processes in the operating system and then pre-

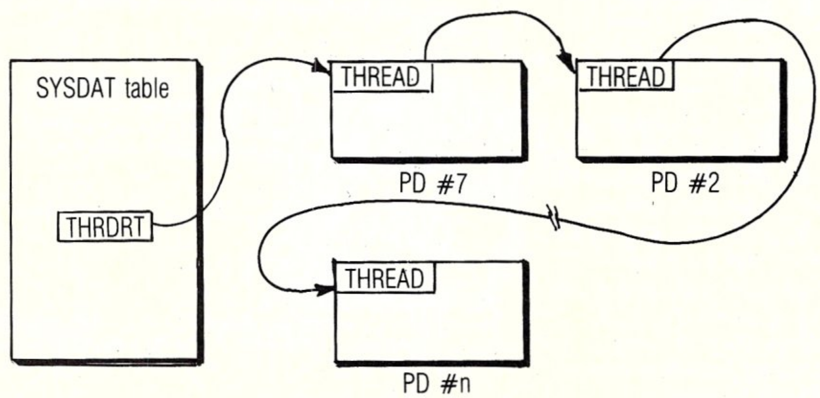


Figure 4. All Process Descriptors are Linked on the THREAD List.

sents the user with a choice to examine a particular process in more detail, as shown in Figure 5. Two procedures form the key elements in the PANA utility, they are GET_SYSDAT and TRAVERSE_PROCESS_LIST.

The procedure GET_SYSDAT obtains the offset and segment address of the SYSDAT table. The values are obtained by making a CDOS system call using the TURBO Pascal BDOS procedure.

The procedure TRAVERSE_PROCESS_LIST uses the Thread List root pointer of the SYSDAT table to obtain the offset of the first process descriptor in the system. The list of PDs is then followed through the THREAD field of each process descriptor. As each node in the list is visited, the procedure copies the data from the process descriptor into the PD_Contents record. All of the located PDs are kept in an array which is returned with the number of located processes.

Note that while the thread list is being traversed, the CPU interrupts *must be disabled*. If the interrupts would not be disabled a dispatch could occur and remove or insert a new process descriptor in the list. As a consequence the thread may be lost and result in a never ending list of garbage.

The remaining procedures finally ask for a particular process and then display the information extracted from the process descriptor.

COMING ATTRACTIONS

This concludes the second part of this series on the Concurrent DOS operating system. Part Three will deal with system queues, how they are used from a program, and how processes can use queues to communicate with each other. (μ)

Alex Soya has been involved with CP/M for over 9 years. He is currently a senior in computer science at Florida Institute of Technology. His interests include Operating Systems, Data Communications, System Utilities, Real Time and Scientific Programming. Alex is a member of the Association for Computing Machinery (ACM).


```
PANA - (C) 1986,  Alex K.H. Soya

          Process    Virtual    Descriptor
Number     Name      Console      Offset
------   -------    -------    ----------
  1      PANA          0          9340h
  2      PCTmp1        1          93C0h
  3      PCTmp2        2          9400h
  4      PCTmp3        3          9440h
  5      PCTmp4        4          9480h
  6      PCTmp5        5          94C0h
  7      Redir         0          9500h
  8      PCTmp0        0          9540h
  9      PIN0          0          9580h
 10      PIN1          0          95C0h
 11      PIN2          0          9600h
 12      CLOCK         0          9640h

Enter process number to display ==>1

Process Name       : PANA
Console Number     : 0
List device        : 0
Process Status     : RUN
Priority           : 200 (00C8h)
Default Disk       : C
Default User area  : 0
Attributes         : TABLE
Suspend flag       : Not set
State list LINK    : 0000h
Thread list THREAD: 93C0h
Parents PD offset  : 9540h
```

Figure 5. PANA Program Output

Listing 1. **PANA Program**

```
{
Version      :  1.0   - 04/11/86

Description:  Program to demonstrate access to the SYSDAT table and process
              descriptors in the Concurrent CP/M operating system.

              Traverses all process descriptors in the system and
              copies their contents to the PROCESS_LIST array.

              Presents user with menu to look at a particular process
              in more detail.

Language     :  Turbo Pascal Version 3.01A for CP/M 86

Environment:  Concurrent CP/M Version 4.1 on CompuPro S-100 Computer
              System.

              The procedures in this program are written in a generic
              manner and should work with any Concurrent CP/M based
              Computer system.

Author       :  Alex K.H. Soya  - P.O.Box 510121, Melbourne Beach, FL 32951
                                 Compuserve [70406,1452]
                                 Goliath Concurrent RCPM node 001
                                         (305) 727-0331 (300/1200/2400 bps)

}
Const
   S_SYSDAT = 154;          { CCP/M-86 function call to return SYSDAT address }
   Max_Process = 256;       { Max number of processes }

         { Field Offsets into process descriptor }

         PDO_LINK   = 0;             { Link to next PD in same state }
         PDO_THREAD = 2;             { Thread to next PD             }
         PDO_STAT   = 4;             { Process State flag            }
         PDO_PRIOR  = 5;             { Current Priority field        }
         PDO_FLAG   = 6;             { PD characteristics field      }
         PDO_NAME   = 8;             { Name field                    }
         PDO_UDA    = 16;            { field containing UDA segment  }
         PDO_DISK   = 18;            { default disk                  }
         PDO_USER   = 19;            { default user area             }
         PDO_MEML   = 22;            { field containing memory ptr   }
         PDO_PARENT = 30;            { parrents PD offset            }
         PDO_CNS    = 32;            { Current console number        }
         PDO_LIST   = 36;            { Current list device number    }
         PDO_SFLAG  = 38;            { Second flag field             }

Type
   Hex_String = String[4];
   PD_Contents = Record
                 OFF    : Integer;      { Offset of this PD               }
                 LINK   : Integer;      { Link to next PD in state list   }
                 THREAD : Integer;      { Link to next PD in Thread list  }
                 STAT   : Byte;         { process state activity code     }
                 PRIOR  : Byte;         { Process priority                }
                 FLAG   : Integer;      { Process run-time characteristic}
                 NAME   : String[8];    { The name of the process         }
                 UDA    : Integer;      { UDA segment address             }
                 DISK   : Byte;         { Default drive code              }
                 USER   : Byte;         { Default user number             }
                 MEML   : Integer;      { Pointer to Memory list          }
                 PARENT : Integer;      { Parents PD offset               }
                 CNS    : Byte;         { Default console number          }
                 LIST   : Byte;         { Default list device number      }
                 SFLAG  : Integer;      { Second flag                     }
               end;

   PD_List = Array [1..Max_Process] of PD_Contents;

Var
   SYS_SEG      : Integer;                { Segment  of SYSDAT table      }
   SYS_OFF      : Integer;                { Offset of SYSDAT table        }
   PROCESS_LIST : PD_List;                { List of processes             }

   regs  : Record          { The 8086 registers  }
             ax,bx,cx,dx,bp,si,di,ds,es,flag :Integer
           end;

   Num_Processes : Integer;               { Number of processes located   }


Function Int_To_Hex_Word(Value:Integer):Hex_String;

    {Converts an integer word to a Hex string }

Const
   Hex_Digits : Array [0..15] of char =
                ('0','1','2','3','4','5','6','7','8','9','A','B','C','D','E','F');

Var
   I  :Integer;

Begin
   Int_To_Hex_Word:=  Hex_Digits[Trunc(Hi(Value) div 16)]+Hex_Digits[Hi(Value) mod 16]
                    + Hex_Digits[Trunc(Lo(Value) div 16)]+Hex_Digits[Lo(Value) mod 16];
end;


Procedure Get_Sysdat(Var Offset,Segment:Integer);

    {Returns offset and segment of CCP/M SYSDAT table }


Begin
   Regs.CX := S_SYSDAT;
   Bdos (Regs);
   Offset := Regs.AX;
   Segment:= Regs.ES;
end;


Procedure Traverse_Process_List(Var P_List:PD_List; Var PD_Num:Integer);

{Procedure traverses all processes in the system and returns the
 process descriptor contents in P_List array}


Const
   THRDRT = $72;       {Thread List Root offset in SYSDAT}

Var
   I        : Integer;    { Loop control      }
   PD_Off   : Integer;    { Current PD offset }
```

```
Begin

   PD_Num := 0;
   inline ($FA);                  { Disable Interrupts }
   PD_Off := MemW[SYS_SEG:SYS_OFF+THRDRT];

   While (PD_Off <> 0) do begin
      PD_Num:=PD_Num+1;

      With P_List[Pd_Num] do begin
         OFF    := Pd_Off;
         NAME   := '';
         LINK   := MemW[SYS_SEG:Pd_Off+PDO_LINK];
         THREAD := MemW[SYS_SEG:Pd_Off+PDO_THREAD];
         STAT   := Mem [SYS_SEG:Pd_Off+PDO_STAT];
         PRIOR  := Mem [SYS_SEG:Pd_Off+PDO_PRIOR];
         FLAG   := MemW[SYS_SEG:Pd_Off+PDO_FLAG];
         For I:= 0 to 7 do NAME:=NAME+Chr(Mem[SYS_SEG:Pd_Off+PDO_NAME+I]);
         UDA    := MemW[SYS_SEG:Pd_Off+PDO_UDA];
         DISK   := Mem [SYS_SEG:Pd_Off+PDO_DISK];
         USER   := Mem [SYS_SEG:Pd_Off+PDO_USER];
         MEML   := MemW[SYS_SEG:Pd_Off+PDO_MEML];
         PARENT := MemW[SYS_SEG:Pd_Off+PDO_PARENT];
         CNS    := Mem [SYS_SEG:Pd_Off+PDO_CNS];
         LIST   := Mem [SYS_SEG:Pd_Off+PDO_LIST];
         SFLAG  := MemW[SYS_SEG:Pd_Off+PDO_SFLAG];

         PD_Off := THREAD;
      end;
   end;
   inline ($FB);                  { Enable Interrupts  }
end;

Function Choice(P_List:PD_List;Num_P:Integer):Integer;

{ Shows all processes in P_List on screen and requests a selection from
  operator being the process number of further interest}

Const
   Lines_Per_Screen = 24;

Var
   I,J          : Integer;            { Loop Control                  }
   Answer       : String[5];          { users response                }
   Code,Select  : Integer;            { Conversion code, User selection }
   d            : Char;

Begin
   Writeln('         Process   Virtual   Descriptor');
   Writeln(' Number    Name    Console     Offset   ');
   Writeln(' ------  --------  -------  ----------');

   For I:= 1 to Num_P do begin
      Write (I:5,'   ');
      For J:=1 to 8 do write(P_List[I].NAME[J]);
      Write(P_List[I].CNS:7);
      Writeln('          ',Int_To_Hex_Word(P_List[I].OFF),'h');
      If I mod Lines_per_Screen=0 then begin
         Write('[more..]',^M);
         Read(KBD,d);
      end;
   end;

   Writeln;
   Repeat
      Write (^M,'
      Write ('Enter process number to display ==>');
      Read (Answer);
      Val (Answer,Select,Code);
   until (Code=0) and (Select>0) and (Select<=Num_P);
   Choice := Select;
end;

Procedure Display (Process:PD_Contents);

{ Displays contents of process descriptor }

Var
   I: Integer;

Begin
   Writeln;Writeln;
   Write  ('Process Name       : ');
   For I:=1 to 8 do write(Process.NAME[I]);
   writeln;
   Write  ('Console Number     : ');
   Writeln(Process.CNS);
   Write  ('List device        : ');
   Writeln(Process.LIST);
   Write  ('Process Status     : ');
   Case Process.STAT of
      00: Write('RUN         ');
      01: Write('POLL        ');
      02: Write('DELAY       ');
      06: Write('READ QUEUE  ');
      07: Write('WRITE QUEUE');
      08: Write('FLAG WAIT   ');
      09: Write('CIO WAIT    ');
   else
      Write    ('Code:',Int_To_Hex_Word(Process.STAT),'h ');
   end; {Case}
   Writeln;
   Writeln('Priority           : ',Process.PRIOR:3,' (',Int_To_Hex_Word(Process.PRIOR),'h)');
   Writeln('Default Disk       : ',Chr(Process.DISK+65));
   Writeln('Default User area  : ',Process.USER);
   Write  ('Attributes         : ');
   For I:=0 to 15 do begin
      If (((1 shl I) and Process.FLAG) <> 0) then begin
         Case (1 shl I) of
            $0001: Write('SYS ');
            $0002: Write('KEEP ');
            $0004: Write('KERNEL ');
            $0010: Write('TABLE ');
            $8000: Write('8087 ');
         end;
      end;
   end;
   writeln;
   Write   ('Suspend flag       : ');
   If (process.SFLAG and 1=1) then
      writeln ('Set')
   else
      writeln ('Not set');
   Write   ('State list LINK   : ');
   Writeln (Int_To_Hex_Word(Process.LINK),'h');
   Write   ('Thread list THREAD: ');
   Writeln (Int_To_Hex_Word(Process.THREAD),'h');
   Write   ('Parents PD offset : ');
   Writeln (Int_To_Hex_Word(Process.PARENT),'h');

end;


Begin
   Writeln ('PANA - (C) 1986,  Alex K.H. Soya');
   Writeln;
   Get_Sysdat (SYS_OFF,SYS_SEG);
   Traverse_Process_List(Process_List,Num_Processes);
   Display(Process_List[Choice(Process_List,Num_Processes)]);
end.
```

µ

Recovering PC-DOS Files

by Edwin Thall

Sooner or later you accidentally erase a diskette file and, of course, did not make a back-up. You remember reading somewhere that the file is recoverable. But can the *non expert* bring back that file without the aid of a sophisticated utility? The answer is "yes," provided you are familiar with the Debug utility and you can count in hexadecimal. In this article, I explain diskette organization for the MS/PC-DOS version 3.00 and demonstrate the recovery of an actual file.

Investigate PC-DOS disk file System using the Debug utility

DISKETTE ORGANIZATION: A LITTLE BACKGROUND

Before I show you how to recover a file, it is essential that you understand diskette organization. The 5¼" MS/PC-DOS floppy is organized into 40 (80 for high capacity) tracks that hold 512 bytes of data per sector. DOS 3.00 supports the following diskette formats:

160K single-sided	8 sectors/track
180K single-sided	9 sectors/track
320K double-sided	8 sectors/track
360K double-sided	9 sectors/track
1200K double-sided	15 sectors/track (high capacity)

All examples and displays cited in this paper pertain to the DOS 3.00 360K format.

Approximately 2% of the diskette's storage space is devoted to overhead (boot record, File Allocation Table (FAT), and directory). Any diskette that starts up the operating system must have the boot record as its first sector. The next sector is the start of the FAT. The FAT stores the information regarding sectors which are in use and provides for file access. The FAT always occurs in duplicate. For example, a four-sector FAT contains 1024 bytes followed by the identical 1024 bytes. The directory immediately succeeds the FAT and records special information for each file. All sectors following the directory are data sectors.

The Debug utility relies on logical sector numbering (LSN) to locate sectors on the diskette. This method assigns the first sector on the diskette (track 0/side 0/sector 1) as LSN 0, the second sector as LSN 1, etc. The diskette organization by LSN appears in Table 1.

DISPLAYING OVERHEAD SECTORS

To recover erased files, you will need to modify FAT and directory sectors. I will demonstrate how to display these sectors using the Debug utility furnished on the DOS supplemental diskette. I suggest that you follow along step by step with an experimental diskette. This diskette should be a 360K format with the operating system and Debug included in the directory. Place your diskette in drive A and type:

```
A) DEBUG
```

The dash that you see is the Debug prompt. Use the "L" command to load the 12 overhead sectors into memory.

```
-L 0 0 0 C
```

The first parameter specifies that the data is to be stored beginning at offset 000 of the current data segment. The second parameter designates that the sectors are loaded from drive A (0 = drive A, 1 = drive B, etc.). The third parameter identifies LSN 0 as the first sector to be read from the diskette. The last parameter, C, calls for a total of 12 sectors to be read. Debug commands are in hexadecimal, and C corresponds to 12. From now on, all hexadecimal system numbers will end with H.

The "D" command, followed by the offset, displays memory to the screen from the current data segment (DS). Each time you enter the "D" command, 128 bytes of memory are displayed (Figure 1). The numbers to the left specify the addresses of the memory locations. The data segment (left of colon) is assigned by the system, while the offset (right of colon) defines the number of memory locations displaced from the beginning of the data segment. The ASCII characters are also displayed on the screen (right side) and I will include them only when meaningful. The hexadecimal addresses for the 12 overhead sectors that were loaded into memory are:

LSN 0	BOOT	DS:0000H
LSN 1	FAT	DS:0200H
LSN 2	FAT	DS:0400H
LSN 3	FAT	DS:0600H
LSN 4	FAT	DS:0800H
LSN 5	DIR	DS:0A00H
LSN 6	DIR	DS:0C00H
LSN 7	DIR	DS:0E00H
LSN 8	DIR	DS:1000H
LSN 9	DIR	DS:1200H
LSN 10	DIR	DS:1400H
LSN 11	DIR	DS:1600H

You may view any of these sectors by entering "D" and the appropriate offset. For example, the first directory sector may be displayed with,

```
D A00H.
```

THE FAT

The organization of the FAT holds the key to a successful file recovery. The FAT is arranged by cluster and maintains records concerning the allocation of diskette space.

Table 1.

DOS 3.00 Diskette Organization by Logical Sector Numbering

Diskette Format	Number of Sectors	Logical Sector Number: BOOT	FAT	DIR	DATA
160K	320	0	1-2	3-6	7-319
180K	360	0	1-4	5-8	9-359
320K	640	0	1-4	5-11	12-639
360K	720	0	1-4	5-11	12-719
1200K	2400	0	1-14	15-28	29-2399

```
-D 200
DS:0200  FD FF FF 03 40 00 05 60-00 07 8F 00 09 A0 00 FF
DS:0210  CF 00 0D E0 00 0F 00 01-11 20 01 13 40 01 15 60
DS:0220  01 17 80 01 19 A0 01 1B-C0 01 1D E0 01 1F 00 02
DS:0230  21 20 02 23 40 02 25 60-02 FF 8F 02 29 A0 02 2B
DS:0240  C0 02 2D E0 02 2F 00 03-31 20 03 33 40 03 35 60
DS:0250  03 37 80 03 39 A0 03 3B-C0 03 FF EF 03 3F 00 04
DS:0260  41 20 04 43 40 04 45 60-04 47 80 04 49 A0 04 4B
DS:0270  F0 FF 00 00 00 00 00 00-00 00 00 00 00 00 00 00
```

Figure 1. The first 128 bytes of the FAT.

```
DS:0200  FD FF FF  03 40 00  05 60 00  07 80 00  09 A0 00  FF
DS:0210  CF 00  0D E0 00  0F 00 01  11 20 01  13 40 01  15 60
DS:0220  01  17 80 01  19 A0 01  1B C0 01  1D E0 01  1F 00 02
DS:0230  21 20 02  23 40 02  25 60 02  FF 8F 02  29 A0 02  2B
DS:0240  C0 02  2D E0 02  2F 00 03  31 20 03  33 40 03  35 60
DS:0250  03  37 80 03  39 A0 03  3B C0 03  FF EF 03

         FFD FFF  003 004  005 006  007 008  009 00A
         FFF 00C  00D 00E  00F 010  011 012  013 014  015 016
         017 018  019 01A  01B 01C  01D 01E  01F 020
         021 022  023 024  025 026  FFF 028  029 02A
         02B 02C  02D 02E  02F 003  031 032  033 034  035 036
         037 038  039 03A  03B 03C  FFF 03E
```

Figure 2. The first 93 bytes of FAT in coded (above) and decoded (below) form.

```
-D A00
DS:0A00  49 42 4D 42 49 4F 20 20-43 4F 4D 27 00 00 00 00   IBMBIO  COM'....
DS:0A10  00 00 00 00 00 00 00 78-E5 08 02 00 04 23 00 00   .......xe....#..
DS:0A20  49 42 4D 44 4F 53 20 20-43 4F 4D 27 00 00 00 00   IBMDOS  COM'....
DS:0A30  00 00 00 00 00 00 00 78-E5 08 0B 00 10 6D 00 00   .......xe....m..
DS:0A40  43 4F 4D 4D 41 4E 44 20-43 4F 4D 20 00 00 00 00   COMMAND COM ....
DS:0A50  00 00 00 00 00 00 00 40-0E 09 27 00 1A 56 00 00   .......@..'..V..
DS:0A60  44 45 42 55 47 20 20 20-43 4F 4D 20 00 00 00 00   DEBUG   COM ....
DS:0A70  00 00 00 00 00 00 00 40-0E 09 3D 00 85 3B 00 00   .......@..=..;..
```

Figure 3. The first 128 bytes of the directory.

Table 2.

Chaining of First Three Files in 360K FAT

Cluster	Value in cluster		
2	003	first file begins	
3	004	chains to cluster	4
4	005	" "	5
5	006	" "	6
6	007	" "	7
7	008	" "	8
8	009	" "	9
9	00A	" "	A
A	FFF	first file ends	
B	00C	second file begins	
C	00D	chains to cluster	D
D	00E	" "	E
.			
.			
.			
26	FFF	second file ends	
27	028	third file begins	
28	029	chains to cluster	29
29	02A	" "	2A
.			
.			
.			
3C	FFF	third file ends	

All numbers are in hexadecimal

One cluster occupies 1.5 bytes (12 bits) and points to either one sector (160K/180K/1200K formats) or two sectors (320K/360K formats). Every three bytes (24 bits) of the FAT are coded to represent two clusters as:

```
12 34 56      coded FAT (3 bytes)

12 34 56      decoding scheme

412   563     decoded FAT (2 clusters)
```

The first 93 bytes of the FAT are illustrated in Figure 2, with every 24 bits underlined in coded (above) and decoded (below) form. The first cluster (cluster 0) of a FAT identifies the diskette format as one of the following:

FF8H	fixed disk
FF9H	1200K
FFCH	180K
FFDH	360K
FFEH	160K
FFFH	320K

The example FAT is confirmed as 360K since FFDH is the value stored in cluster 0. The value in the second cluster (cluster 1) is always FFFH.

The FAT accesses files by a *chaining* process. The first data file begins with cluster 2. The value stored in cluster 2 (003) chains to the next cluster (cluster 3) in the file. The value stored in cluster 3 (004) chains to cluster 4, and so on. Cluster 10 (FFFH) signals the last cluster in the file. The second file begins with cluster 11 and chains through cluster 38, while the third file begins with cluster 39 and chains through cluster 60. System files (IBMBIO.COM, IBMDOS.COM, COMMAND.COM) must be stored in the first data sectors of the diskette. The clusters of system files will always chain consecutively. However, all files need not be stored in consecutively numbered clusters. This is especially true for files of a well-used diskette. The chaining of clusters for the three system files are summarized in Table 2.

Beginning with cluster 2, every cluster in a 360K FAT specifies two sectors on the diskette. The general equation for determining the LSN from the cluster number (C) is:

$$\text{LSN} = (C - 2)(\text{sectors per cluster}) + \text{LSN of first data sector}$$

The equation for the 360K format becomes:

$$\text{LSN} = (C - 2)(2) + 12$$

For example, cluster 2 represents LSN 12 and LSN 13. The first data file in the FAT chains from cluster 2 through cluster 10. Therefore, the first data file is contained in LSN 12-29.

THE DIRECTORY

Vital information is stored in the directory every time you create or update a file. The first 128 bytes of the directory are shown in Figure 3. Each file is assigned a field of 32 bytes holding the following information:

byte 0-7	filename
8-10	filename extension
11	attribute
12-21	reserved
22-25	time and date
26,27	cluster entry in FAT
28-31	file size

The first byte of an active directory file begins with an alphabet character. Filenames may be up to eight characters in size with blanks (20H) used as filler for filenames occupying less than eight bytes. The attribute byte controls whether the file is visible (20H) or hidden (27H) with a DOS directory search.

Bytes 26 and 27 are crucial for file access or file recovery. They point to a file's cluster entry in the FAT, with the low order at byte 26. For example, COMMAND.COM points to a FAT entry at 0027H. The last four bytes of the field store the file size. These bytes are reversed, with the low order at byte 28. The file size for COMMAND.COM is 561AH or 22,042

```
-D 200
DS:0200  FD FF FF 03 40 00 05 60-00 07 8F 00 09 A0 00 FF
DS:0210  CF 00 0D E0 00 0F 00 01-11 20 01 13 40 01 15 60
DS:0220  01 17 80 01 19 A0 01 1B-C0 01 1D E0 01 1F 00 02
DS:0230  21 20 02 23 40 02 25 60-02 FF 0F 00 00 00 00 00
DS:0240  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00
DS:0250  00 00 00 00 00 00 00 00-00 00 00 E0 03 3F 00 04
DS:0260  41 20 04 43 40 04 45 60-04 47 80 04 49 A0 04 4B
DS:0270  F0 FF 00 00 00 00 00 00-00 00 00 00 00 00 00 00

-D A00
DS:0A00  49 42 4D 42 49 4F 20 20-43 4F 4D 27 00 00 00 00   IBMBIO  COM'....
DS:0A10  00 00 00 00 00 00 00 78-E5 08 02 00 04 23 00 00   .......xe....#..
DS:0A20  49 42 4D 44 4F 53 20 20-43 4F 4D 27 00 00 00 00   IBMDOS  COM'....
DS:0A30  00 00 00 00 00 00 00 78-E5 08 0B 00 10 6D 00 00   .......xe....m..
DS:0A40  E5 4F 4D 4D 41 4E 44 20-43 4F 4D 20 00 00 00 00   eOMMAND COM ....
DS:0A50  00 00 00 00 00 00 00 40-0E 09 27 00 1A 56 00 00   .......@..'..V..
DS:0A60  44 45 42 55 47 20 20 20-43 4F 4D 20 00 00 00 00   DEBUG   COM ....
DS:0A70  00 00 00 00 00 00 00 40-0E 09 3D 00 85 3B 00 00   .......@..=..;..
```

Figure 4. The FAT and directory after erasing COMMAND.COM.

```
-E 23A

DS:023A  0F.8F  00.02  00.29  00.A0  00.02  00.2B

DS:0240  00.C0  00.02  00.2D  00.E0  00.02  00.2F  00.00  00.03

DS:0248  00.31  00.20  00.03  00.33  00.40  00.03  00.35  00.60

DS:0250  00.03  00.37  00.80  00.03  00.39  00.A0  00.03  00.3B

DS:0258  00.C0  00.03  00.FF  E0.EF  <ENTER>
```

Figure 5. Reconstructing clusters in FAT for COMMAND.COM.

bytes. This decimal value may be seen in the listing with a DOS directory search.

FILE RECOVERY

When you erase a file, changes are made to the directory and the FAT. In the directory, the first character of the filename is changed to E5H. The FAT is modified by storing 000 in the file's clusters. The zeros signify that these clusters are now free. The contents of an erased file are preserved until the clusters are reassigned to a new file. We are now ready to perform a file recovery.

```
-Q
A) ERASE COMMAND.COM
A) DIR
```

COMMAND.COM should no longer appear in the directory listing. The first 128 bytes of the revised FAT and directory are listed in Figure 4. The information concerning COMMAND.COM remains in the directory until the 32 byte field is reassigned. The "e" indicates that the file is no longer active. Observe how the clusters of COMMAND.COM have been filled with zeros to signify they are free. The successful recovery of this file depends upon restoring the directory and the FAT as they existed prior to the erasure.

The directory (offsets 0A5AH and 0A5BH) indicates cluster 27H was the FAT entry point for the erased file. The number of clusters that were allocated to COMMAND.COM are determined by:

$$\text{Number of clusters} = \frac{\text{file size}}{\text{bytes per cluster}} =$$

$$\frac{22042}{1024} = 21.5 = 22 = 16H$$

COMMAND.COM started in the FAT with cluster 27H and chained through cluster 3CH (27H + 16H = 3CH). You may have noted when chaining is consecutive, the value stored in the cluster is one greater than the cluster number. To restore the FAT, cluster 27H should be filled with 028H, cluster 28H with 029H, etc. The last cluster (cluster 3CH) should be filled with FFFH. Use the "E" command to reconstruct clusters 27H-3CH (Figure 5). Since the size of the COMMAND.COM file and its position in the FAT varies from one DOS version to the next, remember to make modifications if you are using a DOS version other than 3.00.

The FAT contains the vital information for accessing files. To protect against the loss of this information, DOS maintains a duplicate FAT at DS:0600H. However, the duplicate FAT is not involved in file access. The duplicate FAT is restored by reconstructing offsets 63AH-65BH as done in Figure 5.

The directory is easy to restore. Use the "E" command to alter the first character of the filename from E5H to 43H (C).

```
-E A40
DS:0A40  E5.43
```

To complete the file recovery, the changes made to the FAT and the directory are written to the diskette as follows:

```
-W 0 0 0 C
-Q
A) DIR
```

The COMMAND.COM file now has been recovered. You can verify it by booting the system and attempting such commands as DIR, RENAME, COPY, and ERASE.

SUMMARY

The method presented for file recovery works only for those files chaining consecutively in the FAT. The clusters of a well-used diskette may no longer chain consecutively, and an attempted recovery may result in the generation of a damaged FAT. It is considerably more difficult to recover a file whose clusters do not chain consecutively. The file must be pieced together by examining the sectors of those clusters containing 000. Therefore, it is a prerequisite to know the content of the file. The recovery of such a file may be accomplished by reconstructing the chaining of clusters in the FAT. (μ)

Dr. Edwin Thall is Associate Professor of Chemistry at The Wayne General and Technical College of The University of Akron. He teaches chemistry and computer programming.

DR. DOBB'S C TOOLBOX: POWERFUL

DR. DOBB'S C SOFTWARE TOOLS ON DISK

To complement the *Toolbook, Dr. Dobb's* offers the following programs on disk. Documentation and full C source code are included.
Except where indicated, both CP/M and MS/PC-DOS versions are available.

SMALL-C COMPILER

Jim Hendrix's Small-C Compiler is the most popular piece of software published in *Dr. Dobb's* 11-year history. Like a home-study course in compiler design, the Small-C Compiler and the *Small-C Handbook* provide everything you need but the computer for learning how compilers are constructed, and for learning C at its most fundamental level.

Available for MS/PC-DOS or CP/M systems. Please specify format.

Small-C Compiler **Item #007** **$19.95**

SMALL-MAC:
AN ASSEMBLER FOR SMALL-C

This assembler features simplicity, portability, adaptability, and educational value. The package includes:

- a simplified macro facility
- C language expression operators
- object file visibility
- descriptive error messages
- an externally defined instruction table

You get the macro assembler, linkage editor, load-and-go loader, library manager, CPU configuration utility, and a utility to dump relocatable files. Documentation is also included.
For CP/M systems only. Please specify format.

Small-Mac **Item #012A** **$29.95**

SMALL-TOOLS:
PROGRAMS FOR TEXT PROCESSING

This package of programs performs specific, modular operations on text files, including: editing; formatting; sorting; merging; listing; printing; searching; changing; transliterating; copying; concatenating; encrypting and decrypting; replacing spaces with tabs and tabs with spaces; counting characters, words, or Small-Tools is supplied in source code form only. With the Small-C Compiler you can select and adapt these tools to meet your own needs. Documentation is included.

Small-Tools **Item #010A** **$29.95**

DR. DOBB'S TOOLBOOK OF C

Over 700 pages of C material, including articles by such C experts as Kernighan and Ritchie, Cain and Hendrix, Skjellum and Holub! The level is sophisticated and pragmatic, appropriate for professional C programmers.
The most valuable part of the ***Toolbook*** to many will be the hundreds of pages of useful C source code, including:

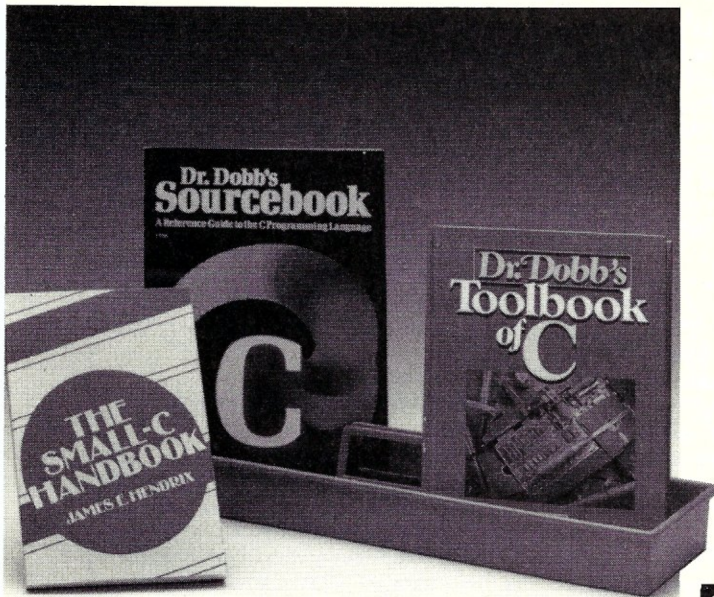
- Jim Hendrix's famous Small-C Compiler and New Library for Small C—Also available on disk!
- NEW! Hendrix's Small Mac: An Assembler for Small C and Small Tools: Programs for Text Processing—Both also available on disk!
- All of Anthony Skjellum's C Programmer's Notebook columns distilled by Tony into one thought-provoking chapter

The accompanying text explains, in the programmers own words, why they did what they did.
You'll find all the best C articles and code published in ***Dr. Dobb's*** over the years updated for the ***Toolbook***, including Ron Cain's original Small-C article.
From M&T Publishing and Brady Communications

Dr. Dobb's Toolbook of C **Item #005** **$29.95**

TO ORDER: RETURN ORDER FORM, OR
CALL TOLL-FREE 1-800-528-6050 EXT 4001
AND REFER TO PRODUCT ITEM NUMBER, TITLE AND DISK FORMAT
(ORDERS ONLY)

TOOLS FOR SOFTWARE DEVELOPERS



SPECIAL PACKAGES 20% OFF

CP/M C PACKAGE

Receive this special package and save $20!
You'll get: Dr. Dobb's Toolbook for C; The Small-C Handbook; The Small-C Compiler on disk; The Small-Mac assembler on disk, with documentation; and The Small-Tools text-processing programs on disk, with documentation all for only $99.95!
Please specify format.

CP/M C Package **Item #005A** **$99.95**

MS/PC-DOS C PACKAGE

Save $20 when you order this special package.
You'll receive: Dr. Dobb's Toolbook of C; The Small-C Handbook with the MS/PC DOS Addendum; The Small-C Compiler on disk; and The Small Tools text-processing programs on disk, with manual all for only $82.95.

MS/PC-DOS C Package **Item #005B** **$82.95**

THE SMALL-C HANDBOOK

Jim Hendrix's *Small-C Handbook* is the reference book on his Small-C Compiler. In addition to describing the operation of the compiler, the book contains complete source listings to the compiler and its library of arithmetic and logical routines.
A perfect companion to the Hendrix Small-C Compiler available from *DDJ* on disk, the *Handbook* even tells you how to use the compiler to generate a new version of itself!
While both the *Handbook* and the *Toolbook* provide documentation for the Small-C Compiler, the *Handbook* contains a more detailed discussion and is available with addendum for the MS/PC-DOS version.
From M&T Publishing and Brady Communications

The Small-C Handbook **Item #006** **$17.95**

The Handbook with MS/PC-DOS Addendum **Item #006A** **$22.95**

C DISK FORMATS

When ordering, please indicate MS/PC DOS or CP/M. For CP/M disks, please specify one of the follwing formats: Apple, Osborne, Kaypro, Zenith Z-100 DS/DD, 8" SS/SD. Special order formats are available for an additional $10 each.

TO ORDER

Return this coupon with your payment to:
M&T Publishing, 501 Galveston Dr., Redwood City, CA 94063

Name ______________________

Address ______________________

____

City State Zip

Item #	Description	Price

Subtotal ________

CA residents add appropriate sales tax ____% ________

Shipping ________

In U.S., add $2.25 per item, $8.75 for special C packages.

TOTAL ________

For disk orders, please indicate format

☐ MS/PC-DOS ☐ Zenith Z-100 DS/DD
☐ CP/M ☐ Kaypro ☐ Apple
☐ Osborne ☐ 8-inch SS/SD

Special order formats available for additional $10 per disk.

☐ Check enclosed. Make payable to M&T Publishing.
Charge my ☐ VISA ☐ M/C ☐ American Express
Card # ______________________
Expiration Date ______________________
Signature ______________________

3032A

Echelon's Z-SYSTEM

by Morris Simon

Rich Conn's ZCPR has delighted serious CP/M users since it first appeared on SIG/M boards in the early Eighties. You probably know ZCPR simply as a replacement for CP/M 2.2's console command processor (CCP), the operating system component which parses user input and contains primitive utilities for renaming, listing, erasing and a few other file operations[1].

Now in its third extensive revision, Conn's ZCPR3 has been combined with Dennis Wright's ZRDOS to form Echelon's "Z-System," which completely overwrites CP/M 2.2's CCP and BDOS, solving many annoying CP/M problems while adding significant new features to Z80 systems in the bargain. If you've been tempted to 'upgrade' to MS-DOS because of CP/M's weaknesses, take a serious look at Echelon's Z-System before you join the Big Blue legion. If you enjoy eight-bit speed and efficiency as much as I do, this combination of ZCPR3 and ZRDOS, plus some powerful new utilities, may provide all the computing power you need for the next decade or so.

VERSIONS OF Z-SYSTEM

Z-System comes in several flavors. The basic distinction is between auto-install and manual-install versions. If you like to tinker and plan to customize the component files, the manual-install version is probably what you should acquire, along with all of the appropriate documentation. In addition to standard CP/M system utilities (MOVCPM, SYSGEN, etc.), you'll also need a good relocating Z80 assembler and other software. Echelon offers some inexpensive 'Z-Tools' for this purpose (the ZAS/ZLINK/ZLIB assembler set and other utilities) which are probably the best Z80 development packages you'll find anywhere.

If you have a basic Z80 system with at least 48K of RAM and a minimum of 110K storage per floppy disk, you can avoid the headaches of modifying, reassembling, and loading the ZCPR3 source code by purchasing the auto-install version called "Z-COM." The difference between Z-COM and the manual version is greater than you might think. Z-COM doesn't actually poke Z-System components into and above your CBIOS, as you do with the manual-install version. Instead, all Z-System segments and special ZCPR3 buffers will be written automatically below your current CBIOS, followed by ZRDOS and ZCPR3 itself. Z-COM requires around 512 bytes more of your RAM space than the manual version, but it's a fast way to get a fine new operating system into your machine.

Whichever route you select, the effects of installing and using Z-System are the same. You'll be removing CP/M 2.2 completely by overlaying the CCP with ZCPR3's ZCP and the BDOS with ZRDOS. You'll also be modifying your CBIOS by the addition of memory-resident ZCPR3 packages of code called "system segments." Figure 1 compares the memory structure of both basic forms of Z-System with that of standard CP/M 2.2.

RESIDENT ZCP COMMANDS

Z-System's command processor (ZCP) has ten internal utilities: DIR, LIST, TYPE, GO, ERA, REN, SAVE, GET, JUMP, and NOTE. If you're an experienced CP/M user, you'll recognize a few of these commands, although they're more versatile and convenient than the CP/M 2.2 equivalents. For example, DIR permits you to view $SYS file names and TYPE has a default console paging toggle - - *no more ^S!* ZCPR3's version of SAVE is more convenient than CP/M's because it allows you to specify memory in 256-byte blocks with hexadecimal notation and avoid the conversion required by debuggers like DDT. The REN command is identical to CP/M's except that it prompts you to approve any file erasures resulting from duplicate filenames, just as the ZCP ERA command permits optional verification of all deletions.

The internal commands GO, GET and JUMP provide system monitor features which are absent in CP/M 2.2's CCP. GO reruns the last TPA program by calling whatever code still resides at 100H. JUMP is similar, permitting you to execute a routine at any address in memory ("JUMP 100" is equivalent to "GO"). The GET command loads a file starting at any page of memory. You can use these and other utilities in combination with Z-System's internal *Flow Control Package* (FCP) to load and run programs under conditional and automatic direction of the operating system. You'll like it.

LIST sends a file to the current printer, without pagination or formatting, much like echoing the output of CP/M's TYPE command with ^P. NOTE is a helpful command, used most often to insert a comment in "aliases," one of Z-System's brand of batch files which consist of a series of commands strung together on a few lines. This is necessary because ZCP uses semicolons, the usual comment marker, to separate multiple commands on a single line.

In the manual-install version of ZCPR3, you can toggle each of these internal commands on or off. The reason you might want them off is because you may wish to use more powerful versions of these utilities. Z-System will allow you to place a limited number of memory-resident utilities in your CBIOS as *Resident Command Packages* so that they'll be available instantly, replacing weaker ZCP routines.

RESIDENT COMMAND PACKAGES (RCP's)

Z-System uses the RCP concept to extend its repertoire of memory-resident commands to include powerful utilities whose codes are too large to reside in the ZCP, which is limited to 2K for compatibility with CP/M. Each RCP is just a transient .COM file which has been relocated to a portion of the permanent CBIOS. You can adjust the size of the RCP buffer and may wish to do some space-trading with other system segments to get exactly what you want in the way of superfast memory-resident utilities inside your CBIOS.

Space saved anywhere in high memory can be used for other purposes in Z-System, so you should avoid duplication whenever you can. Generally, you have a choice between weaker basic utilities in the ZCP and stronger basic utilities in the RCP's. Opting for ZCP utilities frees room in RCP's for a wider selection of fast specialized programs. I prefer to use ZCP versions of common utilities like TYPE, REN and ERA, reserving RCP space for more powerful things.

When you install Z-System manually, you must plan your RAM map carefully so that any customized RCP you choose to place in the CBIOS has enough room to operate. Make sure its special stack requirements do not exceed the value allowed for RCP and other Z-System utilities (48 bytes is the default). The system segment loader, LDR, permits you to install revised RCP's and other segments whenever you change your mind. It's a little more difficult to modify the auto-install version, since the actual RCP space allotted is fixed. Echelon needs to provide a set of utilities which will facilitate customization of each Z-System component in the Z-COM version.

In the auto-install version, the default 16-block (2K) RCP buffer is designed for a preselected group of transient commands, including CP, ECHO, ERA, LIST, P, POKE, PROT, and TYPE. Notice that some of these routines use the same file/command names as

---

[1] *The CCP has a ragged life, coming and going according to the whims of programs in the Transient Program Area (TPA). Unlike CP/M's BDOS and your CBIOS, which must stay in place, the CCP is often overwritten by program segments which need its RAM space at the top of the TPA. That's why some CP/M programs end with a 'warm boot' routine which writes a fresh copy of the CCP in its usual place below the BDOS. For a similar reason involving BDOS system calls #13 (Reset Disk System) and #28 (Set Disk to R/O Status), you must always recopy the CCP into place when you change a disk if you intend to write anything to it.*

their ZCP equivalents, and mean to replace the internal procedures with more powerful versions of the same routines. For example, the RCP commands LIST and TYPE commands allow ambiguous filenames while the ZCP versions do not, and the RCP form of ERA allows an "inspect" option prior to erasures.

Among the new commands, CP is a weak, but fast, copying utility and ECHO is an enhanced version of the ZCP's NOTE command. PROT provides a resident STAT-like utility which sets and resets the file attribution bits for R/W, R/O and DR/SYS options. P is a "peek" routine allowing you to dump a sector of memory while POKE permits you to change bytes wherever you like. With both P and POKE in RCP's, you have instantaneous memory editing power in high memory. Since TPA programs never overwrite the CBIOS, you'll be able to monitor them inside RAM without ever warmbooting! I use POKE or MU3, a screen-oriented Z-System memory editor, more often than a debugger for fast trial runs of programs in which I change only a few bytes. Both are great for testing temporary configuration parameters if you know their RAM locations. Experimental modification of programs was never easier!

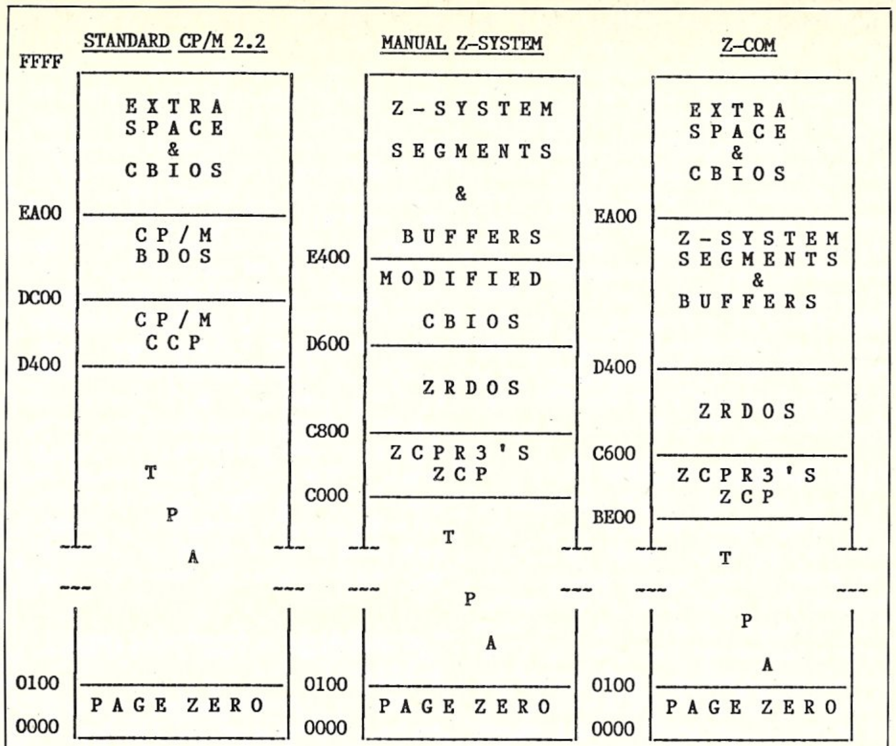


Figure 1. A comparison of RAM maps for standard CP/M 2.2, the manual-install version of Z-System, and the auto-install version (Z-COM). All addresses are hexadecimal and approximate. The locations will vary depending upon hardware components and implementation of Z-System.

INPUT/OUTPUT PACKAGE (IOP)

One of the many improvements Z-System offers CP/M users is the reduction of CBIOS space by eliminating antiquated READER and PUNCH routines and substituting other I/O code only when needed. A space of 1.5K (buffer size is always up to you in Z-System) is reserved inside the CBIOS for whatever I/O drivers you wish to include in a separate IOP. You mi t wish to maintain different I/O files for configurations of various peripherals and load them as a resident IOP whenever switching hardware components. If you have a variety of I/O devices, the redirectable IOP will be such an improvement over the CP/M IOBYTE-STAT redirection process that you'll wonder why Digital Research ever did it that way. You'll also be able to use special Z-System utilities (DEV and DEVICE) to display and assign named devices.

FLOW COMMAND PACKAGE (FCP)

Four blocks of the CBIOS (512 bytes) are reserved for a memory-resident *Flow Command Package*. When installed, the FCP permits testing of nine boolean arguments and IF/ELSE flow direction for TPA programs. Each of the arguments (e.g., a test for user input) is optional and may be disabled or enabled during manual installation by modifying the SYSFCP.LIB header file. It would be nice for users of the auto-install version to be able to toggle each of these conditions individually, either from the keyboard or within a program, but I haven't found an *easy* way to do that yet.

Once more, you have a trade-off decision to make. Do you want the speed and efficiency of a resident FCP, or would you use the flow arguments so seldom that you could call them into the TPA in a transient file? If you opt for a TPA file, you could also use logic tests on lists of filenames, whereas the FCP version permits only a single, albeit ambiguous, file designation. Such file lists and the extra CBIOS space (512 bytes) are the only advantages of TPA-based flow controls. I like to have flow control as a resident feature, particularly for customized error processing and conditional assembler aliases.

NAMED DIRECTORIES

On small systems with two 5¼" floppy drives, CP/M user areas have often been more trouble than they're worth. Z-System now lets you do something useful with user areas, even on tiny systems. It converts them into named directories and lets you use the directory names (e.g., "ROOT:") as well as drive/user designations ("A15:") in most Z-System applications.

For example, my user area A5: is named "DEBUG" and contains all of my debugger tools and utilities. Whenever I change to the DEBUG directory by the command "CD DEBUG:" (or "CD A5:"), a batch file named ST.COM executes changing directory names and the command path to suit my debugging habits. (If I want to enter the DEBUG area without changing anything, I can use "DEBUG:" or "A5:". CD.COM is a transient utility used to *C*hange *D*irectories, while ST.COM is an 'alias' file used as a batch command by CD.COM).

Notice that ZCPR3 recognizes either the drive/user or the named designation, and that the internal CP/M USER command is absent because it's no longer necessary. In the manual version of Z-System, the prompt can be tailored to suit your purposes, but the default is usually something like "A5:DEBUG)". Unless you use an enhanced directory utility (such as XDIR) with a parameter to display all user areas, only the files in the DEBUG directory area will be visible to you, a handy feature in turnkey secure systems when used with internal passwords. Z-System's transient utility, PATH, can be used to change the command path to search whatever directories and/ or drives you like for command files, and a recent ZRDOS upgrade allows you to declare directories of overlay files to be "public" areas accessible to everyone on a multiuser system.

Since the named directories are memory-resident in CBIOS, any of my disks can be segmented into the same logical categories, or they can be redivided by loading a different directory (NDR) file. You can automate the process by loading different NDR files as soon as you log into a different directory, producing a virtual hierarchy of subdirectories although all named directories exist on the same system level.

ENVIRONMENT DESCRIPTORS (Z3ENV & TCAP)

The environment descriptor is one of the most essential components of Z-System. It is simply a two-part header file containing addresses of all ZCPR3 packages in its Z3ENV half and a table of terminal capabilities in its TCAP portion. The memory-resident Z3ENV/ TCAP system segment was a stroke of genius. It allows Z-System developers to put the address of the environment descriptor at the start of any transient program and thus utilize whatever particular features the host system or terminal provides, including special video commands. One of the nagging problems with CP/M was its inability to adjust itself to particular strengths and weaknesses of individual hardware. CP/M's famous transportability was purchased at the cost of special terminal features, which is just one of the reasons for the greater popularity of single-standard operating systems like MS-DOS. Transportability of programs is little problem among machines with the same hardware.

With Z-System, however, users with very different hardware components can run the same flashy, powerful program (with customized graphics, menus and even windows) simply by changing one word: the address of ZCPR3's environment descriptor with its tables of system and terminal equates. Any transient program can use whatever special system or terminal features are listed in Z3ENV or TCAP tables after being installed with the special Z-System utility, Z3INS.COM. The underlying concept of the Z-System environment descriptor is so simple that I often wonder how Digital Research's developers missed it. We might still be using CP/M if they had provided something like it.

Of course, you can change the contents of Z3ENV simply by editing addresses and terminal specifications to suit your needs. Z3ENV also contains such information as CPU speed, maximum numbers of disks and users, CRT and printer specifications, and filename specifications used by various utilities. The TCAP component specifies terminal features such as arrow keys, cursor sequences, clearing and highlighting sequences, and entries for business graphics, pull-down menus and windows.

ZCPR3 COMMAND PARSING

Under ZCPR3, transient programs run just as they do under CP/M 2.2, with the lowest TPA address set at 100H. The ZCP inputs and parses a command line either from the user, from a Z-System SUB file (which resemble a combination of CP/M SUB and XSUB files), or from a memory-resident extension to the ZCP called ZEX. After input and parsing, the ZCP checks, in order, the resident FCP, the current system flow state, and resident command tables of both the RCP and CP. If the FCP's conditional arguments are true *and* if the flow state permits execution *and* if either the RCP or the ZCP recognizes the command, the routine is called. For example, the system logic might say, "IF USER IS A PRIVILEGED *WHEEL*, SET FLOW STATE TO TRUE AND RUN PROGRAM, ELSE FLOW STATE IS FALSE SO DO NOT RUN PROGRAM."

Only after all of these steps does ZCP assume that the command is a transient and begin to look for the appropriate .COM file along the command path. If found, the transient is loaded into the TPA and executed in the usual manner. If ZEX (or any other) ZCP extension is resident, control is transferred to that shell for execution of the command. Finally, the ZCP (or its extension) may invoke an error handler and print a customized message on the screen.

Since most of this process is done by memory-resident utilities and packages, Z-System is much faster than CP/M for complicated command processing, and slightly faster for simpler commands. These advantages are available to users of ZCPR3 alone, without modifying the CP/M BDOS. When ZCPR3 is supported by the optimized Z80 code and system calls of Echelon's ZRDOS, though, you will experience true eight-bit elegance and power.

ZRDOS

The optimized Z80 code used in ZRDOS is based upon Rich Conn's excellent SYSLIB3 and Z3LIB, and is designed as an integral component of the total Z-System environment. ZRDOS developer Dennis Wright has finally severed the umbilicus between Digital Research and CP/M programs, although most of the system calls will seem unchanged on the surface from the original CP/M 2.2 versions. Wright's Z80 optimizations are hidden behind the traditional system calls to permit upward compatibility for CP/M programs, but the code inside his black boxes is very elegant and fast. You'll need to disassemble ZRDOS to really appreciate the improvements, which include every trick in the SYSLIB3 and Z3LIB books. Neither Echelon nor Dennis Wright has released the ZRDOS source code because the system is still evolving. Good guys have to make money, too, you know!

Of course, system programmers won't need to see the ZRDOS code to use it. Just plug the call numbers and required data in the appropriate registers, just as you've always done with CP/M programs. Your assembler and linking loader will do the rest, as long as they understand Zilog mnemonics and/or macros. There are just a few alterations and additions which you'll need to remember.

The major changes in CP/M 2.2 system calls occur in Numbers 10 (*Read Console Buffer*), 13 (*Reset Disk System*), 28 (*Write Protect Disk*), 30 (*Set File Attributes*), and 37 (*Reset Drive*). In addition to these changed calls, there are four new ones specific to ZRDOS: #47 (*Return Current DMA Address*), #48 (*Return ZRDOS Version Number*), #50 (*Set Warm Boot Trap*) and #52 (*Reset Warm Boot Trap*).

These enhancements fix three of the more persistent problems which have plagued CP/M 2.2 users through the years: the nagging warm boot nuisance (#13), unfriendly error messages (#50 & #52) and clumsy disk changing procedures (#28 & #37). ZRDOS also provides new file archiving and 'wheel' protection to improve system backup efficiency and security (#30), like similar enhancements in CP/M 3.0 ("CP/M Plus").

You can't really appreciate the convenience and power of ZRDOS until you use it and suddenly realize that you're not having to press Control-C every five minutes. You'll also enjoy customizing your own error handlers and messages, and running the countless new Z-System utilities pouring out of Echelon's growing community of serious users.

CONCLUSIONS

Enhanced options are the real advantage of Z-System. Wherever you look in the excellent manuals or through some 400K of online documentation files, you'll face choices rather than restrictions and be advised on how to make them. Z80 users finally have an operating system which not only *permits* different hardware environments, but actually encourages and exploits them to strengthen programs!

Eight-bit machines never worked as well as they do with Z-System in control. As for the future, Echelon continues to anticipate new user requirements with the onset of Hitachi HD64180 and (finally) Zilog Z800 boards. In fact, most of the "Z-Tools" sold by Echelon already provide mnemonic upgrades for the HD64180, and a "ZRDOS3" with special "super-chip" features is waiting in the wings. A Micromint SB180 board run by Z-System is Number One on my 'want' list.

I think the best component of Z-System is Echelon's team of fine professionals. It's a small company, staffed and engineered by some of the more prominent micropioneers of early SIG/M days. They seem to be fully committed to the support of eight-bit users in general and of Z-System users in particular. Echelon's staff impresses me as being both energetic and knowledgeable in their support of the products they sell. My questions have been answered immediately and clearly, without any of the usual bureaucratic shuffling I've come to expect from software companies.

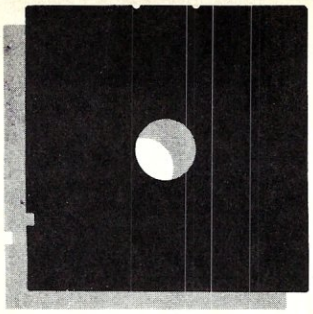
Many Z-System components can be downloaded from public domain bulletin boards, particularly those called "Z-Nodes" which constitute an international network of Z-System users. Most of the ZCPR3 source code, along with SYSLIB3 and assembly versions of the major utilities will be available for the price of a phone call and diskettes. You'd pay Echelon about the same price for the same software, and receive *Z-NEWS*, the best users' newsletter I've ever seen.

Individual members of Echelon's development team retain all rights to other Z-System materials, such as ZRDOS and Z-COM, the auto-install version of Z-System. You'll have to buy those items, either directly from Echelon or from a licensed SYSOP on one of the Z-Nodes. The prices range from $44 to $200 for Z-System, depending upon the version and package you purchase. Rich Conn's two books, *ZCPR3: The Manual* ($19.95), and *ZCPR3: The Libraries* ($29.95), are well worth their prices. These, plus *Z-System Users' Guide* ($14.95), should tell you everything you need to know in order to build an incredible eight-bit operating and development system which will meet your exact requirements for years to come.

For more information and current price lists, write:

Echelon, Inc.
885 North San Antonio Road
Los Altos, CA 94022
(415) 948-3820

Morris Simon is an anthropologist specializing in the relationships humans have with their machines. He's also a freelance writer, documentation consultant and author of more than fifteen interactive novels. His favorite pastime is developing public domain software for ZCPR3, CP/M, MS-DOS and other systems.



In the SIG/M Public Domain

by Stephen M. Leon

Steve Leon is the SIG/M Disk Editor. In other words, he is the person who assembles, compiles, and edits all of the SIG/M public domain software disks. Thus, he speaks with the greatest authority as to what is going on in the SIG/M public domain software area.

If the airlines and the truckers can deregulate and go into any kind of business they want to, then SIG/M users group should be equally free to enlarge its sphere of operations.

We have been complaining long and loud about *beggar-ware*. We have received some nasty letters and some nice ones about our attitude. An interesting one came the other day from Paul Naitoh, who wrote to the editor of M/SJ:

> *I have been sensing a deep sense of 'frustration' by Steve Leon towards PC Public Domain programs. I have been wondering about it as I have had very satisfying experiences with SIG/M Public Domain. I wondered why Steve feels a sense of even 'indignation' towards some of the PC public domain programs. A couple of months ago, I got my own Zenith, an IBM PC/XT clone, and I began to acquire PC Public Domain programs. Now I know. Some programs offered in PC Public Domain as 'shareware' are absolutely worthless or very misleading. Some of these programs offer attractive graphic display but they do NOT work unless I pay $35 - 50. Why should I pay $6.00 to pay to get that program to start with? I realize now that Steve Leon has been sheltering us CP/M public domain users from all those hungry (and greedy) programmers by refusing to include 'shareware' asking money for giving you a demonstration program.*

This kind of mail and comment made us try an experiment with our latest release of SIG/M software. As we have already indicated, we have now made the full library available in most 5" formats. We have issued two volumes of PC/MS-DOS software (SIG/M Vols. 284 and 285). What we are trying to do is make available Public Domain software, including non-CP/M material, that is not sitting there as *beggar-ware*, asking for a donation. What we seek is good quality material, preferably with source code. It cannot be a demo of commercial software or make any request for a contribution.

The Australian, English and Japanese libraries have long included both CP/M and MS-DOS material. SIG/M has stayed away from MS-DOS because of the availability of PC/BLUE as outlet for that material. There is still a great deal of good material that is getting into PC/BLUE, but unfortunately all too much of it caters to the needs of the appliance user as opposed to the advanced user. Most of it lacks source code and all too much of it asks for money.

We see adding PC/MS-DOS material to the SIG/M library as a means of providing an outlet for people who want to donate their software without the stigma of the *beggar-ware* group. We also see it as a means to get into the public domain much more of the machine and language specific software that just cannot get into PC/BLUE because of the large number of panhandlers pressing for space. Thus, we have been promised some twenty volumes of Zenith material. Also promised is a medical office management program written in Knowledgeman. If you have some programming material you think your fellow computerists could use, and will release it for non-commercial public domain use, why not contribute it to the SIG/M library?

We think Hank Kee does a remarkable job with PC/BLUE. We disagree strongly over the value of some of the material in his library, such a PC-RIM, which is an RBASE knockoff distributed as send-me-money-software. We believe that application software written for dBASE or Knowledgeman or RBASE, etc. is needed in the libraries for both teaching and use. The same holds true for LOTUS spreadsheets. If someone wants to write a public domain version of LOTUS or RBASE, that too properly belongs in a public domain library. However, if someone wants to market either of these knockoffs, our sympathy with his high cost of marketing does not extend to marketing it for him for free.

There are commercial operations that market shareware. PC-SIG is such a company. As far as we know, they are in the business of marketing low-priced software and much of what they distribute is shareware. They provide a marketing outlet for people who have written software which they hope to exploit commercially. Unless a computer club is using a public domain library for fund raising purposes (which is not the case with ACG-NJ), computer clubs should not be competing with commercial operations and using their resources to market commercial products.

Thanks to Hank Kee, the Amateur Computer Group of New Jersey has another bulletin board system setup where all of the SIG/M and PC/BLUE material is available. As soon as the software is released, it goes up on this system. All are welcome to use it without pre-registration. You can also use it for contributions to SIG/M and/or PC/BLUE. The phone number is (201) 886-8041. The first two volumes of PC/MS-DOS material (SIG/M Vols. 284 and 285) contain the ACG-NJ Mailing List program in dBASE-III and Clipper. This is an update to our dBASEII version (SIG/M Vol. 110). We use this program for mass correspondence on a daily basis and it handles just about everything we could think of. The program includes Harry Van Tassel's BROWSE in Clipper which is also on Volume 285 as a free standing program.

Those of you who have used Clipper recognize that with all of its strengths, it also has a number of deficiencies. Harry's BROWSE neatly covers one of the big missing items. The source code to the program will show you how we wrote around some of the other deficiencies.

If you do any code editing on a PC and have not used BRIEF you don't know what you are missing. If you do use it, Volume 285 contains instructions on how to date-time stamp code files automatically when using BRIEF as well as code to add line graphics to it. Other items are a program to make .CLP and .LNK files for Clipper, a number lock fix for Microsoft Word, the addition of Clipper's Save Screen to dBASE-III and how to have more than 20 open files in dBASE-III. Volume 282 contains a pretty good Z80 mailing list program, the Reliance Mailing List, and an excellent Z80 screen editor, VDO. Volume 283 contains two libraries of routines, math routines in Basic and I/O routines in Turbo Pascal.

Sigi Kluger's contributions to the public domain are scattered throughout the library and on the BBS systems. Heretofore, we

released a volume of his CP/M utilities (SIG/M Vol. 226) with 56 files on it. He has now sent us two volumes of TurboDOS utilities which are available as volumes 286 and 287.

For those of you who run TurboDOS these are two must releases. Because there was so much material, we combined them into an .LBR file on each disk. The first volume has 62 items in it. The second has 40. Of course, with source code provided in most cases, a single routine can account for more than one item in the library. However, we think Sigi has made available a comprehensive library of professional TurboDOS routines of a caliber not available elsewhere.

Included on this disk are SWEEP for TurboDOS, a ram disk emulator, a V20 and V30 8080 emulator, a command line editor, a remote access package, a function key package and a wildcard expansion module. Our experience with TurboDOS is rather limited, but one of the problems with it related to the large size of the directories with the resultant delays in getting directory information. These volumes contain code to solve that problem as well as ways to maximize directory handling and file manipulation in TurboDOS.

SIG/M 227 contains a Church Management program in dBASE-II. Management of a religious organization using the computer is an area which we believe deserves cooperation from the amateur community. With the price of full feature computers now well below the $1,000 line, even the poorest parish can now afford a computer.

However, a computer must do more than just sit and unfortunately there appears to be a big gap in software for religious organizations. A computer in a church can and should do more than keep a list of contributions. It should be able to handle many of the ministerial aspects of the organization as well as the social aspects.

ACG-NJ is busy organizing a User Group dedicated to computer usage by religious organizations. It will shortly set up its own BBS system and will distribute public domain software in this area both on the BBS and through SIG/M. Software contributions are welcome either to this new group or via SIG/M. For further information, please contact Burke Mawby at (201) 566-7445.

SIG/M Volumes are available on 8″ SSSD Disks for $6 each ($9 foreign) directly from SIG/M, Box 97, Iselin, NJ 08030. Volumes may also be ordered in most 5″ formats (other than Apple, Commodore 64 or high density format) for $7 each ($10 foreign). There is an additional charge of $2 for formats which require more than one disk to hold a 240K volume (such as SSSD formats). Printed catalogs are $3 each ($4 foreign). Disks in a variety of formats may also be obtained through the worldwide SIG/M distribution network. The distributor list is included with the printed catalog. A disk version of the catalog (Volume 00) is available for $6. It also contains the distribution list. Many bulletin boards have the software for downloading and most new releases are available on the CP/M Sig on Compuserve.

NEW SIG/M RELEASES

Volume 282
Reliance Mailing List Program
Screen Editor

Volume 283
Turbo Pascal I/O Routines
Math Routines in Basic

Volume 284 MS-DOS
ACG-NJ Mailing List Program,
dBASEIII (Vol 1 of 2)
Clipper compiled version
Database file

Volume 285 MS-DOS
ACG-NJ Mailing List Program, dBASEIII (Vol 2 of 2)
Source code
Time-Date stamp in BRIEF
Graphics in BRIEF
Browse in CLIPPER
Make .CLP and .LNK files in CLIPPER
More than 20 files open in dBASEIII
Save screen in dBASEIII
Microsoft WORD fix

Volume 286
TurboDOS Utilities (Vol 1 of 2)

Volume 287
TurboDOS Utilities (Vol 2 of 2)

Continued from page 51

MathCAD PUBLISHER RESPONDS

We appreciate Micro/Systems Journal's interest in presenting reviews of MathCad. Naturally we're disappointed that Prof. Cameron isn't satisfied with the current state of MathCAD's user interface, but the ease of use is very subjective.

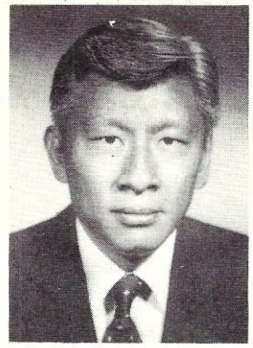
Much of the industry attention which MathCAD has received so far has focused on the power and simplicity of its user interface. Surveys of our user base have established that MathCAD's ease of use is a major factor in customer satisfaction.

Several of the specific points which Prof. Cameron raised have been addressed in the Release 1.1 which will be out by the time you read this. Inserting blank space has been speeded up. Global definitions in appended files are now processed immediately. It is now possible to use subscripts in variable names without invoking range calculations. And equations may now be "commented out," to use Cameron's phrase, so that MathCAD's equation formatting capabilities may be used without triggering calculations.

MathCAD comes with a 30-day money-back guarantee. We think it represents a real breakthrough for anybody who has ever used a scientific calculator. We encourage prospective users to try it out for themselves.

PC/Blue Report

by Hank Kee



Editor's Note: Hank Kee is the librarian for the PC/Blue public domain software library. He is the person who collects, assembles, and checks all the software issued by PC/Blue and then compiles and edits them into the released volumes.

Public Domain Programs You Can Use To Test Your System

Have you ever watched a symphony orchestra perform and noticed the timpani player is constantly fussing over his kettle drums, before and during the performance. Of course, during the performance, he puts his ear close to the top of the drums and tunes it ever so gently so as not to disturb the rest of the performers. Many computer hackers, like me, are the same. We are constantly checking out parts of the system. We need to know at any instant if everything is working fine. With an IBM PC, there is a power-on-system-test program in ROM which is executed at boot time. It tests selected peripherals and memory. On the IBM PC/XT and PC/AT, you see a memory scan performed in the upper left hand corner of the screen. For many of us, this simple power-on-system-test is not enough.

There are a number of public domain software utilities available that provide system test functions that do not necessitate opening up the system. The first is called WHATUHAV. This will display on your screen a block diagram of the equipment in your system. This is useful if you are not certain if a serial port or any other option is properly functioning. I had, an occasion, a need to isolate a modem communication problem in which I had to check an external modem, cable, and telephone line to determine what was wrong with the system. The problem was in the I/O board. It was frustrating to not be able to locate the problem quickly. The last thing I thought of was a faulty serial port. Now I run a quick system scan before I start opening the systems unit.

Another addition to the PC/Blue library is PC Magazine latest Lab Benchmark series #3. This consists of a collection of programs to *verify* the status of your system. This volume includes the latest release of CORETEST version 2.7 from CORE Inc. This will measure the average access time of your fixed disk system. It really doesn't need to be used on a daily basis. This program will display in text and graphics form the average access time of the fixed disk. Many vendors quote their track-to-track time as compared with stating an average access time. The track-to-track time will always appear faster than the normal usage of the disk. I was quite surprised by the variations that exists between one fixed unit compared to another. The PLUS 20 Hardcard is a magnitude faster than any other fixed disk card unit I have tested.

What do I do afterwards when all the diagnostics have been executed. Like the window washer of a skyscraper, I go back to the beginning and do it all over again.

NEW PC-BLUE RELEASES

The recent releases of the library has been greatly enhanced by many different types of business accounting systems. There is no such thing as a generalized business accounting system. A retailing accounting system is different from distributor system, which in turn is different from a manufacturer's system.

Volume 221-222
RBBS-PC v14.1b Remote Bulletin Board System
from Capital PC User Group

Volume 223
Cantonese version 2.0
how to speak Cantonese tutor
Styled version 1.0
text analyzer
series submitted by Louie Crew

Volume 224
FreeCalc version 2.0
electronic spreadsheet
author-Stilwell Software Products

Volume 225
PC-Code3/PC-Code4 version 6.2
software encryption
author-Richard Nolen

Volume 226
MI-Anaylst version 1.10
mortgage investor's forecasting tool
author-Ronald C Pacanowski
PC-Books version 1.0
general ledger
author-Barry Edelman

Volume 227
PC Accounting I
single entry accounting system
designed for very small businesses
author-Stephen Anthony

Volume 228
PC Accounting II
double entry accounting system
with AP, AR, PR, and inventory options
author-Stephen Anthony

Volume 229
D'SCOPE
converts IBM PC into line monitor
author-Harrison Uhl

Volume 230
Spline version 1.3
calculates spline under tension
through given points
author-James R Van Zandt
Graph version 2.0
displays line graphs (CGA or equivalent)
author-James R Van Zandt
Templates of Doom
adventure learning in Lotus
author-Solar Systems Software

Volume 231
PC-GL version 2.7 general ledger
PC-AR version 3.4 accounts receivable
PC-PR version 1.3 payroll
series by Jerry Medlin

Volume 232
PC-Outline version 1.06
author-SoftWorks Development

Volume 233
MCBS General Ledger version 3.2

Volume 234
PC Magazine-Laboratory Benchmark Series
Hardware Performance Tests-release 3

Volume 235
Finance Manager II-GL Version 1.0
author-Hooper International

Volume 236
A86 Macro Assembler version 2.11
author-Eric Isaacson
Extended Disk File Manager DX v1.42
author-Robert K Blaine

Volume 237
Automotive Accounts Receivable v1.40
author-Len Thom
Portfolio ROI and Tax Software v2.08
author-Techserv, Inc.

PC/Blue disks are available from the New York Amateur Computer Club, Inc., Box 106, Church Street Station, NY NY 10008. Price is $7 per volume which includes media, postage, and handling. On foreign orders, please add $2 per disk.

The CP/M Bus

by Robert A. Blum

A Review Of A Terrific Full-Screen Debugger

DDT - If you winced at the thought of this acronym then you probably fall into one of two groups. Either you're aware of the negative long term side effects that this now regulated pesticide has on mammals, or you may have programmed on a CP/M system and used the Dynamic Debugging Tool to help find program problems. In either case, the result of their use is the same: both products are a hazard to your well being.

DSD80

DSD80 is a full screen debugger (gone forever is the scrolling screen) for the 8080 and Z80 microcomputers running CP/M-80 or compatible operating systems. It is fully downward compatible with the command structure of DDT, the standard CP/M program debugger.

HARDWARE REQUIREMENTS

The minimum hardware requirements to operate DSD80 are a CRT capable of direct cursor addressing and a display dimension of 24 lines by 80 columns. Disk space usage is very conservative at 30K total, 15K for the DSD80 program and another 15K for its online help file. I doubt that either of these requirements will be a problem even on the smallest CP/M system.

One consideration though, is the amount of memory that remains after DSD80 has relocated itself into high memory, overlaying the CCP and extending the BDOS downward by its own size. For instance, the TPA size on my development system is 64,772 bytes before loading DSD80, but once loaded the TPA is reduced by approximately 15k to 48,902 bytes, thereby limiting the size of the largest program that I can work on to just under 48k. In my experience that has always been plenty of room, but you might want to make the same calculations using your beginning TPA size to insure that enough address space remains for your needs.

GETTING STARTED

The installation procedure for DSD80 is the weak spot in an otherwise practically flawless product. The documentation begins appropriately with an introductory section explaining how the program works and its requirements, but then jumps headlong into a full description of the displays without first making any mention of how to install the program. To find the installation information a search of the table of contents pointed me to chapter 7 titled Configuration.

This short section of the manual told me what I didn't want to hear: to establish a working protocol between DSD80's terminal driver routines and my CRT would require updating a supplied assembler subroutine with the proper control codes for my CRT. Then assemble the updated subroutine with the provided public domain assembler in order to obtain a hex file for use in the last step. And finally overlaying an executable copy of DSD80 with the newly assembled terminal control subroutine.

All in all not a very difficult job if your terminal information is close at hand, but a lot more trouble than I would have expected. I think a program of this quality deserves an installation program that automatically patches the executable program module with the terminal dependent coding after its selection from a menu.

WHAT YOU SEE

One of DSD80's strongest assets is how the display is handled. The screen image is made up of four major sections: the instruction display occupies the top left quadrant, the register, flag, and stack display fills the top right portion, and the remaining bottom section of the screen is divided between two memory displays. In addition there are two minor display areas and three transient areas used for such things as program messages and help text if requested.

Each line of the instruction display area contains one disassembled program line of the target program. The mnemonics used are Intel for the standard 8080 version or Intel extended or Zilog for the Z80 variant. Additionally, if a symbol table for the target program has been loaded the disassembled code will contain label reference.

In the upper middle portion of the screen the register display shows the current values of all the standard 8080 registers and flags. The Z80 specific index registers are selected via a command as are the Z80 alternate registers.

The stack display is in the upper right portion of the screen. New values pushed onto the stack enter from the top pushing the old values downward. In this way the most current stack is always displayed even though some values may have been pushed off the display area.

The two memory display areas at the bottom of the screen can be used to view independent memory segments or can be linked to display one consecutive area. The display format is 16 bytes wide. The data is first displayed in hexadecimal and then in ASCII with nondisplayable characters replaced with periods.

TARGET PROGRAM CONTROL

The true value of program debuggers such as DSD80 is their ability to control program execution, provide ways to stop the program at desired points, and maintain a constant display of the CPU state. To this end DSD80 provides not one but two modes of operation: real time execution and simulation mode.

Real time operation is actually uncontrolled execution of portions of the target program. This mode is provided to permit time critical routines to operate at full speed. Disk controller software typically falls into this category. To maintain some control over program execution, breakpoints are physically set into the target program at strategic points to return control to the monitor program.

Simulation is totally different from real time execution in that control is never actually given to the target program. Instead the function of each instruction is emulated exactly. This mode of operation permits DSD80 to fully monitor the effects of each instruction and abort its execution should it violate a preset limit or fail simulated hardware error checking not available on the Z80 or 8080 cpu chips. There is, of course, a penalty in execution speed for using this mode of operation. Simulated execution proceeds at between 1000-2000 instructions per second or approximately 1/250th of normal processor speed.

Figure 1. The Entire DSD80 Command Set

```
^A align memory displays          ^P copy screen to printer
^B execute back to current PC     ^Q XON
^C exit DSD                       ^R backup instruction display
^F find PC in instr display       ^S XOFF
^G execute to next line           ^T single step
^H backspace                      ^U erase command line
^J single step                    ^V advance instruction display
^L redraw screen                  ^X erase command line
^M execute command line           ^Y backup memory display
^N advance memory display         ^[ meta command prefix (escape)

                    DSD80 Control Commands

~A toggle user breakpoint         ~Q quit stack checking toggle
~B back trace                     ~R reset the disk system
~C change display console         ~S toggle simulation mode
~D display breakpoints            ~T trace with no operation
~E toggle expert mode             ~U user display toggle
~H toggle highlighting            ~V display version number
~I initialize stack display       ~W toggle words display
~J toggle ^J single stepping      ~X user command X
~N display next load address      ~Y user command Y
~O open file for viewing          ~Z select Z80 register display

                    DSD80 Meta Commands

A assemble instructions           B set permanent breakpoint
C call subroutine                 D memory display
E enter symbol definition         F fill memory
G execute program                 H hex arithmetic
I initialize command line         J set real time subroutine
K kill symbol definition          L instruction display
M move memory                     N set execution limits
O write enable range              P input & output port
Q search for string               R read file
S substitute memory               T trace execution
U untrace execution               V verify memory
W write file                      X set register value
Y set watch monitor               Z zero TPA

                    DSD80 Visible Commands
```

COMMAND SET

The commands to control DSD80's numerous features are broken into three groups: control commands, meta commands, and visible commands.

Control commands are simply the set of control characters. They may be entered at any time and are executed immediately. Control commands perform such functions as editing the command line and scrolling the memory displays.

Meta commands are an extension to the control commands. They still execute immediately but are composed of a two character sequence; the escape character followed by a single letter. Meta commands perform low level functions such as establishing a mode of operation or resetting the disk system.

Visible commands form the more familiar class of commands. They are called visible because they always begin with a single letter and may optionally accept parameters. Arguments are typically sixteen bit addresses and can be expressed in hexadecimal, decimal, and symbolic form.

USER EXTENSIONS

One of the most uncommon features of DSD80 is its support of user written extensions. Up to three user written subroutines can be used to enhance the handling of breakpoints, the format of the display, and the addition of two commands to those already implemented.

Enabling the user breakpoint subroutine instructs the simulator to first call the user subroutine in order to determine whether the target program should be stopped. In addition, this feature is useful to emulate nonexistent hardware or to collect program statistics. The meta A command toggles this feature on or off.

The user display subroutine is normally used to directly drive the area devoted to the memory display. Each time DSD80 needs to write a character into this area of the display the user subroutine is called. A set of functions calls within DSD80 are also provided that can be used by the user subroutine.

Two meta commands are available for calling routines outside of DSD80. The primary use of these commands is in conjunction with the user display.

PUTTING IT TO WORK

One of the more difficult programs that I wrote lately involved capturing large amounts of real time data and storing it into multiple tables depending on the source of the data. What made this program difficult to debug was the dynamic nature of the tables and the requirement that some of them must be maintained in sorted order to facilitate rapid summarization upon demand.

The program was very large and consisted entirely of new development since the final product was to be copyrighted. Not being able to use any of the many pretested routines that I have developed could have added enormously to my development time if it were not for DSD80.

After the coding was done and an initial desk check of the program was complete it was time for a smoke test. The first problem I encountered was a bug in a subroutine dispatching routine that first calculated which level of a vector table contained a desired routines address and would then use the address word fetched from the table as a branch point into the program. A fairly simple routine that worked the first time, but not everytime. Occasionally the branch address would be off by one byte which resulted in a crashed system due to execution of an invalid instruction. After rebooting the system, any symptoms of why the program failed had been erased or overwritten.

Using DSD80 with simulate mode enabled solved the problem quickly because DSD80 first checks for a valid instruction before proceeding with its emulation. Once informed of the illegal instruction I was then able to back trace through the previously executed instructions to find my problem.

The second problem I had would have been equally as difficult to find if it were not for DSD80 because when it occurred large areas of memory were being indiscriminately overwritten. As it happens, the sort routine was the culprit.

To find the problem I first set memory write limits to protect all the vital areas of the system in order to avoid a reoccurrence of the system being completely destroyed. After examining the conditions of the program after a few memory protection violations I discovered that a particular segment of memory always contained the same value after a failure. In order to stop execution of the program at the instant that the suspect value was stored into memory I set the watch monitor of DSD80 to interrupt when the suspect memory location was updated with the proper value. The problem was then easily found by again back tracing through the last several instructions executed.

USER REPORT

DSD80 is a pleasure to use and elevates the normally frustrating task of program debugging to an art form. Its advanced capabilities make it a reasonable replacement for a costly In Circuit Emulator (ICE) in most situations outside of basic hardware development. DSD80 is one of a few programs that I have found that work as documented every time without fuss or bother.

DSD80 can be purchased from Soft Advances, P.O. Box 49473, Austin, Texas 78765, (512)-478-4763. (µ)

The UNIX File

by Ian F. Darwin

This column discusses the UNIX operating system. If you have comments or questions about UNIX or this column, please write to Ian Darwin at Box 603, Station F, Toronto, Ontario, Canada M4Y 2L8. If you have access to the uucp network, mail ''ihnp4!darwin!ian''. I can't always answer immediately, but I will get back to you; electronic mail gets answered first!.

UNIX Internals, Binary Files & Termcap Entries

UNIX SECRET INTERNALS BOOK

How did they get a book on UNIX internals out the door of AT&T? Most likely they were able to do so because The *Design of the UNIX System* by Maurice J. Bach (Prentice-Hall, 1986, ISBN 0-13-201799-7) contains no source code. Instead, it presents a very good description of the algorithms and data structures that make up the UNIX operating system kernel. Algorithms are presented in a C-like pseudo-code that should be intelligible to most any modern programmer. The book evolved from a course the author taught inside AT&T; the students there had access to the UNIX system source code, but that is not a prerequisite for reading the book and understanding the material.

There are many UNIXes in circulation; which one does the book discuss? The important UNIXes today are System V, 4BSD, and Eighth Edition. System V is AT&T's commercial offering. 4BSD is the University of California at Berkeley's system used at many universities and other institutions. And the Eighth Edition is the current Bell Labs research version. Eighth Edition (also called V8) is not available; it has been released to a hand-picked half dozen universities. But V8 has had an influence far out of proportion to the number of sites running it; 'Streams' and the Remote File System in SVR3 are derived from V8. But AT&T considers System V to be its 'standard' commercial offering. Because Bach wrote this book while working at AT&T, you would expect the book to concentrate on System V, and you would be right. But which of the many versions of System V? In the author's own words:

> *The system description is based on UNIX System V Release 2 supported by AT&T, with some new features from Release 3.*

There is also a discussion of some key kernel features and algorithms used in other versions of UNIX, but the emphasis is on System V.

The book is clearly presented and seems to be a good description of the system. In the parts I've read so far I found no serious disagreements with the presentation, and I learned a few things about System V that I hadn't known before. If you want to learn on your own how the current AT&T UNIX system operates, read this book.

LOOKING AT BINARIES

With the trend to binary-only systems, there is a generation of UNIX techies who've never seen the source code. And there is much that can't be understood without looking at the source (see my review of Bach's book above for exceptions). But there is a lot that can be done without source.

A useful tool for looking at binaries is the strings program. First written at Berkeley, this has been re-implemented several times as public-domain versions.[1] You won't see it in System V, though, although it would be just as useful there. The program looks at the ASCII string characters in a binary file. *strings file* lists all the printable character strings in *file*. The command *strings -n file* (or -o, depending on what version of the program you are running) also lists the offset into the file where each string appears, which can be useful for patching. *strings foo ¦ grep /* will show you all the absolute pathnames a binary refers to (except for string concatenation done at run time; at least it will print the names of the directories).

And now the mystery of the ages: which program in a long pipeline issued that obscure message? If all you get is some common *perror( )*-style message unaccompanied by the name of the program, you're out of luck. Why? Because all decent programs now use this standard error function or an extension such as *error(3)*, and thus most contain the same set of common error messages. But if it's an obscure message, you're still out of luck. Unless, that is, you type the following five-line command:

```
for prog in prog1 prog2 prog3
do
echo $prog
strings $prog ¦ grep "cryptic message"
done
```

which will tell you which of several programs a cryptic message is coming from! You must spell prog1, prog2, ... out in full; if they're all in the same bin directory then chdir there and just use the filenames. And of course replace "cryptic message" with the message you are looking for.

For some interesting stuff, try *strings /unix* and see all the messages your kernel can come up with. It's rather voluminous, so I'd suggest you run it through a paginator such as *pg* or *more* (unless you're *very* quick with the CTRL/S key!).

Another way of looking inside binaries is *od* which stands for octal dump (although nowadays it mostly dumps in hex). The command *od -bc file* will give you a dump of a file with the characters displayed as well as their numeric representations. Try *od -bc /* to see what a directory really contains. *Od* exists on all UNIXes, with variations in options. Any similar dump program can be used; a public domain program called *xod* dumps UNIX files in a format familiar to CP/M hackers.

And don't forget your debugger. Take a look at *adb;* if you have a more advanced debugger (sdb, dbx, ...) use it. A debugger can be useful to print parts of programs, especially if you don't have strings. And debuggers usually let you rewrite parts of strings in programs, although not very conveniently.

And how else do you change strings? Here is spatch, a string patch program that I wrote. It's quite horrible, so I've not listed the source code. It's left as an exercise to the reader, for this reason: being able to find and write over selected parts of binary files is dangerous! If you can't work your way past the system calls needed to do such a thing, you're probably not ready to.

Even the syntax is left for the reader. One version I've used is,

```
spatch file offset newstring
```

where file is the name of a binary program on disk, offset is given as an integer, and newstring is given as a string (quoted if it

---

[1] I have a version that works on V7 and 4BSD. The program needs to know the format of binary executable files in order to skip the 'header' portion of such files. The version I have will *not* work on System V Release 2 or later, or any system with 'Common Object File' format or any format other than the original V7 'a.out' format.

contains spaces or special characters). You can no doubt do better, handling multiple strings, etc. A careful programmer should be able to change most of the messages in a binary program into total gibberish (or, more usefully, into French, German, Spanish, etc.) without corrupting the execution of the program. A future release of System V promises to provide a set of tools for changing error messages properly, stay tuned.

TERMCAPS?

Several letters from readers in the last few months have asked about getting *termcap* descriptions for various terminals. *Termcap* descriptions tell the UNIX screen software what control characters are needed to operate a particular terminal. Readers sometimes ask about getting a terminal description for some personal computer, such as the IBM PC or the Apple II. The problem is that such PC's have many, many communications programs. Because each one uses a different protocol to control the screen, each needs a different termcap description. The exceptions are terminal programs that emulate an existing terminal, such as the DEC VT100. For these programs, of course, the *termcap* corresponding to the emulated terminal is the one to use. For programs that don't emulate an existing terminal, you either have to find a description, or write one. In searching for one, be sure to specify *both* what computer and which communications program you are using. Writing a terminal description is a matter of reading the *termcap* manual page, figuring out which dozen or so capabilities really matter (hint: start with cursor addressing, standout, underline, and the terminal's backspace operation), describing them in the termcap language, trying screen-based applications, and refining the termcap description until it works. The process can be as little as an hour's work if the terminal's world-view is similar to that of a *standard* ASCII terminal.

Some versions of System V provide a facility called *terminfo* that is conceptually similar to, but faster than *termcap*, and uses a slightly different format for the terminal description. The advice above applies as well to *terminfo*. There is a termcap-to-terminfo converter in the public domain; if you're doing a lot of terminfo work you might want to get a copy.

That's all for this month. I welcome letters and electronic mail on these and other topics, especially suggestions for future columns. Cheers! µ

Ian Darwin is director of Research and Development for SoftQuad Inc, a company providing supported troff *publishing software for UNIX. He and his wife live in a rural setting, north of Toronto.*

New Products

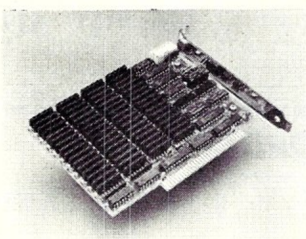
When contacting vendors please mention that you read about them in Micro/Systems Journal.

PC-Bus Products

2Mb RAM Half Card

The Earth Computer "Packram-II" is a high-speed, 2Mb RAM card designed to fit in a PC/XT short slot. It can operate at 8Mhz. It meets the Lotus/Intel/Microsoft extended memory and Intel Above Board specifications. Part of the memory can be used in the 640Kb based memory. RAM disk, cache memory and print spooler software are included.

Suggested list price is $695. To buy direct or for the name of the nearest dealer contact, Earth Computers, Box 8067, Fountain Valley CA 92728; (714)964-5784.

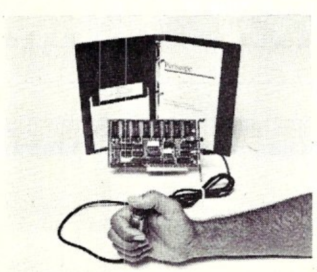


Debugging System

The Data Base Decisions Periscope I (Version 2.00) is a professional debugger for IBM PC/XT/AT and close compatibles. It is intended for software developers who need full crash recovery capability (hung systems and locked keyboards), including protected debugger software. It can be used to debug device drivers, memory-resident, non-DOS programs and even DOS itself.

It includes software, 16Kb write-protected RAM board, remote break-out switch, 150-page manual, quick-reference card and comes with 30-day money-back guarantee.

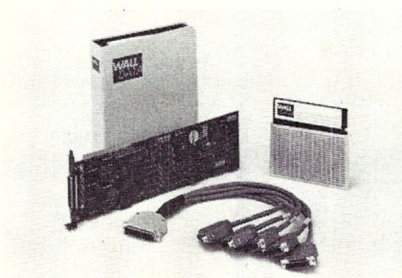
Cost is $295 plus shipping ($2.50 UPS ground) and can be ordered direct from Data Base Decisions, 14 Bonnie Lane, Atlanta GA 30328; (800)722-7006 or (404)256-3860.



S/370-S/3X Interface Board

The Wall Data Datagate/PC is a PC-compatible plug-in card that allows up to 5 PCs or ASCII devices to connect to S/370 or S/3X systems without affecting the processing capabilities of the host PC. It has its own microprocessor, memory, and communications controllers. An operating system is supplied, on disk, which supports multiple, concurrent protocols and communications tasks. A dialback security program protects host programs and data bases. Attachment to the host or front-end processor can be local (via modem eliminator) or remote using either point-to-point or multi-point data lines.

Cost is $2,495. Contact: Wall Data Incorporated, 17769 N.E. 78th Place, Redmond WA 98052;(206)883-4777.



Low Cost GPIB Interface

The Real Time Devices GP100 is a GPIB interface based on the TMS9914A GPIB controller chip that can support up to 15 devices. It can be used as a talker, listener, or controller. The board supports DMA and interrupt. It includes an assembly language interface to Basic, Forth, Fortran or Pascal, with over 40 commands, and manual that provides a GPIB tutorial as well as installation instructions

Cost is $299. For ordering information contact:Real Time Devices, Inc., Box 906, State College PA 16804, (814)234-8087.

S-100 Bus Products

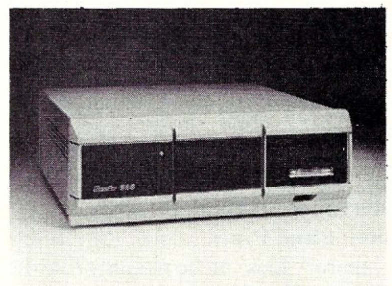
286 Multi-User System

The CompuPro 286/80 is an 80286-based S-100 system with 80Mb hard and 800Kb floppy drives, as well as built-in tape backup unit. It has 768Kb of main memory, expandable to 1Mb, and a 512Kb cache buffer for improved performance. The 16-slot mainframe has a rear panel with slots for up to 18 serial connectors (9 serial ports included with standard configuration).

Included is a proprietary implementation of Concurrent DOS, a real-time, multi-user multi-tasking operating system with record locking and password protection. It contains support for networking and multiple 8 and 16-bit slave processors

Options include a 10Mhz 80286 microprocessor operating with no wait states, networking, slave processors, graphics, additional mass storage and I/O ports.

Suggested list price is $12,500. For location of nearest dealer contact: Viasyn Corp., 26538 Danti Court, Hayward CA 94545; (415)786-0909.

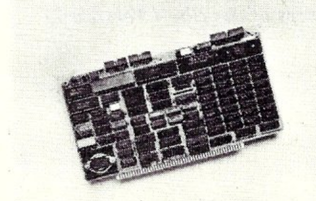


Network Processor

The InterContinental Microsystems (ICM) CPS-16F is an S-100 network processor board compatible with CP/M-80/-86 and PC/MS-DOS. It contains a 20Mhz NEC V30 microprocessor capable of running both 8088/8086 and 8080/Z80 programs. The CPS-16F operates with any master processor that complies with the IEEE-696.1/D2 S-100 standard.

When combined with ICM's MS-1000 software and M/STER PC-compatible terminal, the CPS-16F allows the running of PC-DOS monochrome programs and both TurboDOS and PC-DOS on the same system. The board contains 1Mb of RAM with no wait states, a real-time clock, two serial and two parallel ports.

For more information contact: ICM, 4015 Leaverton Court, Anaheim CA 92807; (714)630-3714.



U.S. Postal Service
STATEMENT OF OWNERSHIP, MANAGEMENT AND CIRCULATION
Required by 39 U.S.C. 3685)

1A. TITLE OF PUBLICATION: MICRO/SYSTEMS JOURNAL
1B. PUBLICATION NO.: 8 7 5 0 9 4 8 2
2. DATE OF FILING: revised form: 11/14/86
3. FREQUENCY OF ISSUE: bi-monthly
3A. NO. OF ISSUES PUBLISHED ANNUALLY: 6
3B. ANNUAL SUBSCRIPTION PRICE: $20.00

4. COMPLETE MAILING ADDRESS OF KNOWN OFFICE OF PUBLICATION *(Street, City, County, State and ZIP+4 Code) (Not printers)*
501 Galveston Dr., Redwood City, CA 94063

5. COMPLETE MAILING ADDRESS OF THE HEADQUARTERS OF GENERAL BUSINESS OFFICES OF THE PUBLISHER *(Not printer)*
same as above

6. FULL NAMES AND COMPLETE MAILING ADDRESS OF PUBLISHER, EDITOR, AND MANAGING EDITOR *(This item MUST NOT be blank)*
PUBLISHER *(Name and Complete Mailing Address)*
Laird Foshay, 501 Galveston Dr., Redwood City, CA 94063
EDITOR *(Name and Complete Mailing Address)*
Sol Libes, P. O. Box 1192, Mountainside, NJ 07092
MANAGING EDITOR *(Name and Complete Mailing Address)*
Lennie Libes, P.O.Box 1192, Mountainside, NJ 07092

7. OWNER *(If owned by a corporation, its name and address must be stated and also immediately thereunder the names and addresses of stockholders owning or holding 1 percent or more of total amount of stock. If not owned by a corporation, the names and addresses of the individual owners must be given. If owned by a partnership or other unincorporated firm, its name and address, as well as that of each individual must be given. If the publication is published by a nonprofit organization, its name and address must be stated.) (Item must be completed.)*

FULL NAME	COMPLETE MAILING ADDRESS
M&T Publishing	501 Galveston Dr., Redwood City, CA 94063

8. KNOWN BONDHOLDERS, MORTGAGEES, AND OTHER SECURITY HOLDERS OWNING OR HOLDING 1 PERCENT OR MORE OF TOTAL AMOUNT OF BONDS, MORTGAGES OR OTHER SECURITIES *(If there are none, so state)*

FULL NAME	COMPLETE MAILING ADDRESS
Markt & Technik	Hans-Pinsel-Strasse 2, 8013 Haar bei Munich West Germany

9. FOR COMPLETION BY NONPROFIT ORGANIZATIONS AUTHORIZED TO MAIL AT SPECIAL RATES *(Section 423.12 DMM only)*
The purpose, function, and nonprofit status of this organization and the exempt status for Federal income tax purposes *(Check one)*
☐ (1) HAS NOT CHANGED DURING PRECEDING 12 MONTHS
☐ (2) HAS CHANGED DURING PRECEDING 12 MONTHS
(If changed, publisher must submit explanation of change with this statement.)

10. EXTENT AND NATURE OF CIRCULATION *(See instructions on reverse side)*	AVERAGE NO. COPIES EACH ISSUE DURING PRECEDING 12 MONTHS	ACTUAL NO. COPIES OF SINGLE ISSUE PUBLISHED NEAREST TO FILING DATE
A. TOTAL NO. COPIES *(Net Press Run)*	12,600	12,400
B. PAID AND/OR REQUESTED CIRCULATION 1. Sales through dealers and carriers, street vendors and counter sales	800	700
2. Mail Subscription *(Paid and/or requested)*	11,200	10,800
C. TOTAL PAID AND/OR REQUESTED CIRCULATION *(Sum of 10B1 and 10B2)*	12,000	11,500
D. FREE DISTRIBUTION BY MAIL, CARRIER OR OTHER MEANS SAMPLES, COMPLIMENTARY, AND OTHER FREE COPIES	----	-----
E. TOTAL DISTRIBUTION *(Sum of C and D)*	12,000	11,500
F. COPIES NOT DISTRIBUTED 1. Office use, left over, unaccounted, spoiled after printing	600	900
2. Return from News Agents	--------	-------
G. TOTAL *(Sum of E, F1 and 2—should equal net press run shown in A)*	12,600	12,400

11. I certify that the statements made by me above are correct and complete — SIGNATURE AND TITLE OF EDITOR, PUBLISHER, BUSINESS MANAGER, OR OWNER

PS Form 3526, July 1984 *(See instruction on reverse)*

CP/M

•Dr. Dobb's Z80 Toolbook
Item #022 $25.00
This book and its companion disk contain everything you need to write your own Z80 assembly language programs. You'll find a method of designing programs and coding them in assembly language, and a complete integrated toolkit of subroutines.
Z80 Toolbook with software on disk *Item #022A $40.00*
Formats: 8" SS/SD, Apple, Osborne, Kaypro

Micro/Systems Journal Library

•Programmer's Guide to CP/M *Item #219 $14.95*
Edited by Sol Libes, this is a collection of all CP/M articles in *Micro/Systems* magazine from January 1980 through February 1982.

BASIC Programmers

•Basic Booster Library *Item #214 $29.95*
This collection of time-saving screen and menu utilities can increase the power and speed of interactive BASIC programs, and simplify the coding required for application screens.

•Turbo Pascal for BASIC Programmers
Item #206A $14.95
Learn why Pascal is easier to write, read, test and debug than BASIC. You'll find a library of useful programs, appendices including the ASCII character codes, a Turbo Pascal dictionary, and other helpful aides. A disk is also available.
•Companion disk *Item #206B $29.95*

Toolbooks from Dr. Dobb's

•Dr. Dobb's Toolbook of 68000 Programming
Item #040 $29.95
This Toolbook contains the b 68000 articles reprinted from *D Dobb's Journal,* along with new 68000 programming tools, applications and examples. Softw in the book is also available on d with full source code.
68000 Toolbook with software on disk *Item #041 $49.95*
Formats: CP/M 8", Osborne, Macintosh, Ami Atari 520st, MS-DOS

•Dr. Dobb's Toolbook of C *Item #005 $29.95*
This authoritative reference contains over 700 pages of mate by C experts, including the best C articles from *Dr. Dobb's Journal.* You'll find hundreds of pages of valuable C source c including a complete compiler, an assembler, and text processing utilities, all also available on disk.

•Dr. Dobb's Toolbook of Forth *Item #030 $22.95*
This collection of useful Forth programs and tutorials contai expanded and revised versions of *Dr. Dobb's* best Forth artic along with new material. The screens in the book are also available on disk with full source code.

Toolbook of Forth with software on disk
Item #031 $39.95
Formats: MS/PC-DOS, Apple II, Macintosh, or CP/M. For CP/M disks specify Osborne or 8" SS/SD.

MS-DOS Utilities & Books

•PC Tools *Item #216 $39.95*
PC Tools provides a complete collection of RAM resident system utility programs for IBM PC's and compatibles. It includes many of the same features found in DOS, along with additional features for recovering accidently deleted files, viewing and editing file contents, and more.

•DS Backup *Item # 215 $69.95*
This high-performance program has all the features you need to make your backup routine fast and easy. The package contains a disk and manual and is not copy protected.

•Taming MS-DOS *Item #060 $19.95*
This advanced user's guide will show you how to enhance and extend the power of MS-DOS so that you can work more efficiently, you'll find nearly 50 batch files and ready-to-use programs with source code so you can customize DOS to fit your needs.

•TallScreen *Item #217 $49.95*
This resident screen and keyboard enhancement utility extend the capabilities of DOS to bring you more flexibility and power. You can scroll the screen to see previous DOS output, store multiple DOS commands, use a full screen editor at the DOS command line, and more.

Turbo Pascal Books & Software Libraries

'urbo-Advantage *Item #070 $49.95*
ave time developing your Turbo Pascal programs with ıis library of 220 routines for Turbo Pascal. Arithmetic perations, bit manipulations, check routines, data ɔmpression, menu functions, MS-DOS support, and atistical procedures are just a few of the routines included. he package contains source code and documentation. or MS-DOS systems.

•Turbo Complex *Item #071 $89.95*
This library of Turbo Pascal procedures and functions allows you to work easily with complex numbers, vectors and matrices. Some *Turbo Complex* routines are most effectively used with *Turbo Advantage* routines (above). The package includes full source code and documentation. For MS-DOS systems.

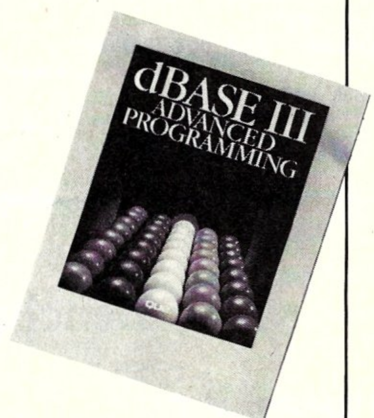
'urbo Display *Item #072 $69.95*
'urbo Display is a screen generator for Turbo Pascal. You'll nd a screen oriented text editor along with 30 Turbo Pascal ·ocedures and functions. The package includes full source ›de and documentation. *Turbo Display* is most effectively ıed with the *Turbo Advantage* routines, above (item #070). ɔr MS-DOS systems.

•Stat Toolbox for Turbo Pascal *Item #050 $69.95*
The *Stat Toolbox* contains two complete packages: a Turbo Pascal library of statisical routines including statistical distribution functions, random number generation, basic descriptive statistics, and more; and fully-functioning statistical programs with two data management systems. Source code is included.

Dbase III Programming Guides

'base III Advanced Programming
tem #204 $22.95
.earn how to speed up Dbase programs, and ıow programs written in assembly, BASIC, 'ascal or C can be adapted to run with Dbase pplications. This book will show you how ɔ write efficient code, how a particular coding ɜchnique works, and when to use it.

•Advanced Programming Guide to Dbase III *Item #213 $28.95*
This is the most in-depth book on Dbase available, containing over 600 pages of information on debugging techiniques, subroutines, structured programming, terminal codes, reserved words, and more.



To Order:

order any of Dr. Dobb's MicroSystems Journal ›ducts, return the order m, or

Call Toll-Free
1-800-528-6050
EXT 4001

ɪ refer to product item mber, title, and disk mat.

›r customer ›rvice questions:

Call M&T
Publishing, Inc.
(415) 366-3600
EXT 216

Order Form

Yes! Please send me the following book(s), software with payment by:
☐ Visa ☐ Mastercard ☐ American Express ☐ Check

Quantity	Item#	Description	Disk Format	Unit Price	Total Price

Name______________________
Address______________________
(Please use street address, not P.O. Box)
City ____________ State ______ Zip ________
Card#______________ Exp. Date________
Signature______________________

Sub-Total	
* Sales Tax	
** Shipping	
Total Order	

*Calif. residents must add applicable sales tax to total

**Add $2.25 per item shipping

3032

THE SOFTWARE DIRECTORY

When contacting software publishers please mention that you read about them in Micro/Systems Journal.

Program Name: RTC
Requirements: PC-DOS or VAX-VMS
Description: Translates RATFOR code to C code
Price: PC-DOS Version, $450; VAX-VMS 9-track Version, $1950, plus $5 S&H.
Publisher: Cobalt Blue
1683 Milroy, Suite 101
San Jose CA 95124
(408)723-0474

Program Name: Invisible Optimizer
Requirements: PC/XT/AT or compatible
Description: A disk buffering product to enhance system speed. Provides automatic disk cacheing in unused RAM area. A linkable version is also available.
Price: $95
Publisher: Stellation Two, Inc.
26 W. Mission St., Suite 3
Box 2342
Santa Barbara CA 93120
(805)569-3132

Program Name: Periscope II-X
Requirements: PC/XT/AT or compatible running DOS 2.0 or later.
Description: A professional debugging system. This new version has command editing, faster monitor breakpoints and support for 8087/80287, Aztec and DeSmet C symbols, and 286 turbo boards. New features also include new traceback display formats, ability to load multiple symbol tables, echo output to disk, defining of up to 4 data windows, enhanced expression evaluation and symbol support. Comes with 30-day money-back guarantee.
Price: $115
Publisher: Data Base Decisions
14 Bonnie Lane
Atlanta GA 30328
(404)256-3860

Program Name: MEGABACK
Requirements: CP/M2.2
Description: Backup for copying hard-disk to multiple floppy disks. Accommodates all media errors and virtually all user errors, and tells how to correct the error before proceeding. Also, has full prompting and password protection. Files can be backed up either individually or as a group. A unique sequence number, checksum and printable directory is stored on each backup disk. Includes a 30-day money-back guarantee.
Price: $95
Publisher: Logic Associates
1433 W. Thome
Chicago IL 60660
(312)274-0531

Program Name: TECHTYPE
Requirements: MS-DOS, CP/M-86 or CP/M-80
Description: Text formatter for scientific, engineering, math, and multi-lingual document production. Allows unlimited sub/super-scripting and can mix up to ten fonts. User can control format, pitch and emphasis. User can access CRT attributes, all 256 IBM characters and with an IBM CGA display all fonts and sub/super-scripts are displayed. Drivers for dot-matrix, and daisy-wheel printers are provided.
Price: $300
Publisher: Green Mountain Radio
Research Company
240 Staniford RD
Burlington VT 05401
(802)862-0997

Benchmarking the Exec-PC Sprint

by Charles H. Strom

The Exec-PC ready-to-install version of the PC-Sprint accelerator board is supplied with possibly the most complete documentation we have ever seen in the personal computer market. There is nothing left to the imagination, and all explanations are in clear, monosyllabic English.

We installed the board in a Taiwanese XT clone with 640Kb of ram on the system board, two floppy drives and a CMI 6426 hard disk with a DTC controller. Since Exec-PC supplied the short ribbon cable, we made use of it, affixing the Sprint board to the rear of the chassis with the included mounting tape.

Operation with the board installed is identical to the stock system except that we occasionally experienced cold boot problems. They are easily cured by activating the new reset switch. Once booted, we have found no incompatibilities whatsoever.

In the May/June, 1986 issue of M/SJ, Sol and Don Libes published an interesting table of benchmarks based on the programs available on PC-Blue's Volume 135, released to the public domain by PC magazine. We replicated several of these benchmarks as well as running Norton's SI parameter and a program called SIEVTIME which calculates the time required to find 18999 primes using the Sieve of Eratosthenes. (This latter program is courtesy of Paul Homchick and is available on GEnie's IBM RoundTable.) Here are our results:

Benchmark	Stock 8088 4.77Mhz	PC-Sprint V20, 7.37Mhz
1	:29	:17
5	:48	:29
6	:38	:23
7	1:45	1:02
8	1:31	:55
9	2:07	1:16
10	:09	:05
11	1:04	:43
SIEVTIME	2:02	1:14
NORTON SI	1.00	2.8

Tests:
1. Prime Number Calculation - compiled Basic
5. Basic Program - Integer addition #1
6. Basic Program - Integer addition #2
7. Basic Program - Floating point arithmetic
8. Basic Program - String Calculation
9. Basic Program - Data Look Up
10. Basic Program - Empty Loop
11. Basic Program - File Update

Micro/Systems Journal™ Back Issues

The following back issues are available at $5.00 for 1 issue, $4.50 each for 2–5 issues and $4.00 for 6 or more issues. Send orders with payment to M&T Publishing, 501 Galveston Drive, Redwood City, CA 94063. Credit card orders may be placed by calling 415-366-3600, ext. 216.

MARCH/APRIL 1985 (Vol. 1, No. 1): Bringing up CP/M Plus, Assembly Language Extensions for MS-Basic, New Tricks for CP/M2.2, Building an IBM/PC or XT Clone, Extended Single Density Storage, Variable Size Arrays in C; REVIEWS: dBase-III and 16-Bit Lisp & ProLog-Part I.

MAY/JUNE 1985 (Vol. 1, No. 2): Build an S-100 to PC-Bus Converter, interfacing to MS-DOS Part-I, Loadable Drivers fo CP/M2.2, Roll Your Own PC-Clone, Bringing up ZCPR-3, C & Godbout Disk-1 Controller, Writing Translation Programs in C and Turbo Pascal; REVIEWS: 16-Bit Lisp & Prolog-Part II.

JULY/AUGUST 1985 (Vol. 1, No. 3): Structured Programming With Microsoft M80 Assembler, Local Variables in Forth, Interfacing to MS-DOS Part-II, Data Translation with Turbo Pascal, Implementing Sets with Bit Operations in C, A Unix Mail List System; REVIEWS: Scientific & Technical Word Processors-Part I, Macrotech MI-286 S-100 CPU Card, Slicer System, Concurrent PC-DOS, Coherent Operating System.

SEPTEMBER/OCTOBER 1985 (Vol. 1, No. 4): Who Prints on Printer with CCP/M Interrupt Borrowing with Turbo Pascal, Upgrading CompuPro I/O Boards, Program Interfacing to MS-DOS Part-III, SIG/M & PC/Blue Public Domain Software Distribution Points, TurboDos INSTALL program; REVIEWS: CompuPro 10+, TEX.

NOVEMBER/DECEMBER 1985 (Vol. 1, No. 5): TurboCharge Your 8086/8088 Computer, Faster Floating Point Math with C, Bringing up CP/M-86K, Program Interfacing To MS-DOS Part-IV, advanced Machine Level Interface Techniques For Turbo Pascal, Context-Independent Macros for C, Build Your Own PC Program Library Using Public Domain Software, dBase-II Speed Techniques-Part I, Tuning Unix Program; REVIEWS: Peak 68K8-CP, Eureka For CP/M-80, Scientific Workstations.

JANUARY/FEBRUARY 1986 (Vol. 2, No. 1): Implement PC-DOS on Non-IBM Compatible Computers, Part-I; TurboCharge Your 8086/8088 Computer, Part-II; Assembling An AT Clone; Build An S-100 EPROM Emulator, Transferring Files Between CP/M and MS-DOS Systems; Program Interfacing to MS-DOS Part V; dBase-II Speed Techniques; Unix Public Domain Software; REVIEWS: Scientific Word Processors-Part II; Turbo-DOS PC; Turbo Pascal Support & Enhancement.

MARCH/APRIL 1986 (Vol. 2, No. 2): Implementing PC-DOS On Non-IBM Compatible System Part-II (Conclusion), Program Interfacing TO MS-DOS Part-VI, Rolling Your Own PC/XT/AT Clone, Converting From CP/M To MS-DOS On The PC, A C Subroutine For Parsing Command-Line Arguments, Arithmetic Aberrations With dBase-III and the dBase Developer's Release, The Marriage of TEX and Postscript; REVIEWS: C Interpreters, C Source-Level Debuggers, Lomas Data Products' S-100 PC, CompuPro S-100 PC Video Board.

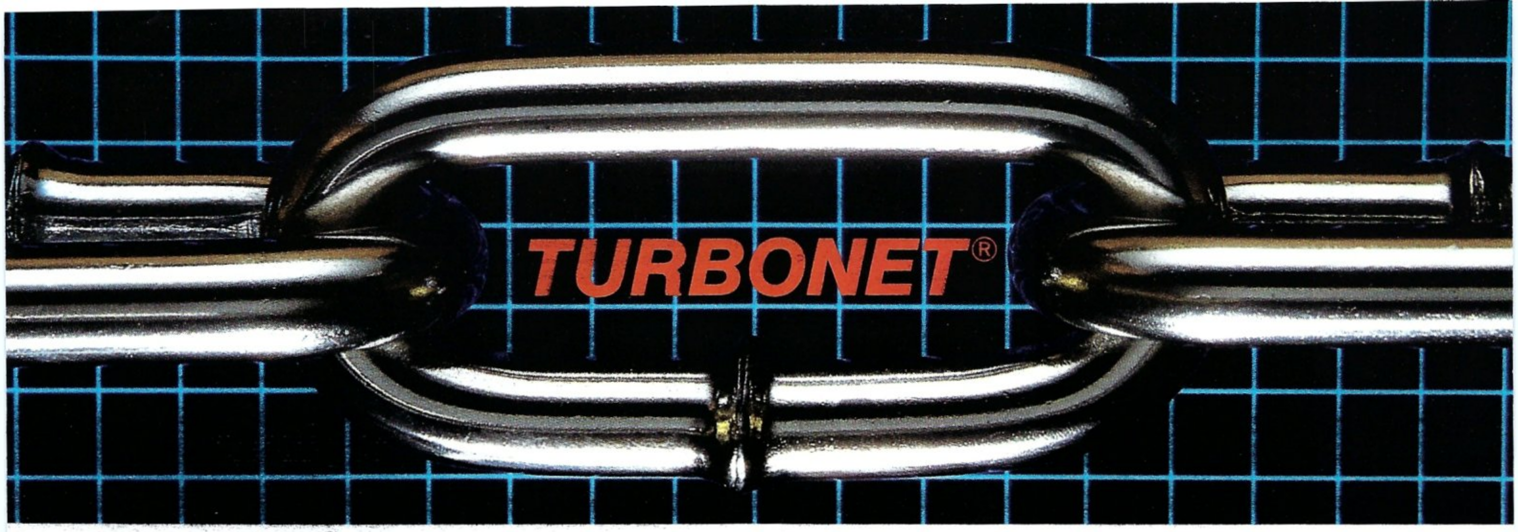
MAY/JUNE 1986 (Vol. 2, No. 3): Build An S-100 HD64180 CPU Card, Configuring MS-DOS, Writing MS-DOS Device Drivers (Part 2), Building An AT Clone, Adding A RAM Disk To A CP/M System, Obfuscated C Code Contest, Sorting With Turbo Pascal, M[illegible] dBase Speed Techniques, Using an RCPM System; REVIEWS: Concurrent DOS, MEX-PC, ConIX, FirsTime.

JULY/AUGUST 1986 (Vol 2., No. 4): Power to the PC, Creating A Copy Protected Program, Build Smart Keyboard Interface, Writing An MS-Dos Memory Disk Device Driver, Declare & Define C Variables in One File, More Loadable BIOS Drivers For CP/M; REVIEWS: Phoenix' PFIX-Plus Debugger, Holliston Challenger XT186, Magnum Digital PRO-180 & FD-100, XPIP, PMATE, Ryan-McFarland Fortran, WAT-COM, Watfor-77, MicroGlyph SciPlot, Softech Matrix Calculator, Personal TEX.

SEPTEMBER/OCTOBER 1986 (Vol. 2, No. 5): Control Systems Made Simple; Enhancing CP/M-80; A First Look At the 80386; Interfacing Using the SCSI Bus; Program Interfacing To MS-DOS—Part IX; Using SYMDEB With NMI Breakpoints, SYSLIB, Z3LIB & VLIB; REVIEWS: PC-Pro; CompuMagic Utility Package; Plu*Perfect Systems' DataStumper.

Micro/Systems Journal accepts Classified Ads. The charge is $6/line (3 lines minimum); 40 characters max./line. Three times frequency $15/line; six times $25; non-profit clubs $2/line. Logos, special type, etc. are extra charge. Check must accompany ad copy. Send to M&T Publishing, Inc., 501 Galveston Dr., Redwood City, CA 94063

Advertiser Index



One Strong Link Can Strengthen Your Whole System

TurboNET® is Teletek's new local area network that permits reliable high speed data transfers among computers of differing bus architectures. TurboNET is designed to be used in networks consisting of multiple S-100 based and PC based systems. It will allow up to 4000 users, including up to 255 IBM PCs or compatibles, to share a single network and all attached peripherals. The network can be organized in any number of different ways mixing Teletek's 8 and 16-bit multiuser systems and PCs in any combination.

Teletek's Networking Family consists of:

TurboNET PC: IBM-PC Network Interface Board

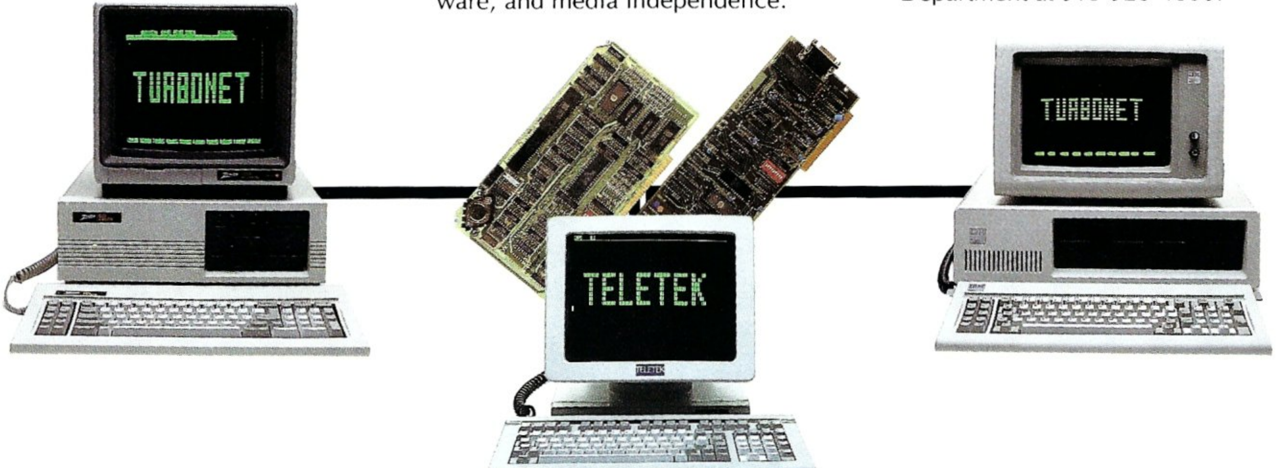
Teletek's TurboNET PC board offers IBM-PC Compatibility, CSMA industry standard protocols, 2 megabits/sec network speed, on-board CPU and communications management firmware, and media independence.

TurboNET S-100: S-100 Network Interface Board

Teletek's TurboNET S-100 board offers IEEE 696 Compatibility, CSMA industry standard protocols, 2 megabits/sec network speed, on-board CPU and communications management firmware, and media independence.

The benefits are obvious: The cost savings of shared peripherals, almost unlimited system expansion capability, and the use of existing PC workstations with the ability to run the myriad of application software written for MS-DOS and PC-DOS. This coupled with Teletek's 8 and 16-bit multiuser systems running application software written for CP/M and MP/M allows the system the ability to access almost any software library.

For more information on Teletek's TurboNET S-100 and TurboNET PC boards or on any of our full line of S-100 products, please call our Sales Department at 916-920-4600.



TELETEK

4600 Pell Drive, Sacramento, CA 95838
(916) 920-4600 Telex #4991834 Answer back — Teletek

The Peak of Performance
SCALE THE HEIGHTS OF PRODUCTIVITY
Sure, you've proven that in your hands a computer is a productive tool. But if you haven't teamed up with a SemiDisk you have heights yet to climb!
IT'S NO MERE RAMDISK
SemiDisk has been leading the way for Disk Emulators since their inception. If you've seen RAMdisks you know what it's like to load programs in an instant, and read or write files without delay. Unlike alternatives, the SemiDisk offers up to 8 megabytes of instant-access storage while leaving your computer's main memory free for what it does best - computing!
KEEP A GRIP ON DATA
Go ahead, turn off your computer. Take a vacation. With the battery backup option, your valuable data will be there in the morning even if you aren't. You'll sleep better knowing not even a 5 hour blackout will sabotage your files.
NEW LOWER SEMIDISK PRICES THAT WON'T SNOW YOU UNDER
512K 2Mbyte
IBM PC, XT, AT $495 $995
Epson QX-10 $595 $995
S-100, SemiDisk II $799 $1295
S-100, SemiDisk I $595 ——
TRS-80 II, 12, 16 $695 $1295
Battery Backup Unit $130 $130
Software drivers available for CP/M 80, MS-DOS, ZDOS, TurboDOS, and VALDOCS 2.
SEMIDISK
SemiDisk Systems, Inc.
P.O. Box GG, Beaverton, Oregon 97075
503-626-3104
Call 503-646-5510 for CBBS/NW, and 503-649-8327 for CBBS/Aloha, all SemiDisk equipped computer bulletin boards, (300/1200/2400 baud) SemiDisk, SemiSpool trademarks of SemiDisk Systems.