Display until JANUARY 20, 1988

Extra All-IBM Edition

BYTE

THE SMALL SYSTEMS JOURNAL®

Inside the IBM PCs

Fourth Annual Special Issue

Assessing the PS/2s
Micro Channel vs. NuBus
OS/2 Programming

$3.95 US
$4.95 CN
£3.65 UK
A McGraw-Hill Publication
0360-5280

Turbo Prolog: The Natural Language of Artificial Intelligence

Whether you're a first-time programmer or an experienced one, Turbo Prolog's natural implementation of Artificial Intelligence soon shows you how to build expert systems, natural language interfaces, customized knowledge bases and smart information management systems.

Turbo Prolog and Turbo C work hand-in-hand

Turbo Prolog® interfaces perfectly with Turbo C® because they're both designed to work with each other.

The Turbo Prolog/Turbo C combination means that you can now build powerful commercial applications using two of the most powerful languages available.

Turbo Prolog's development system includes:

- ☑ A complete Prolog compiler that is a variation of the Clocksin and Mellish Edinburgh standard Prolog.
- ☑ A full-screen interactive editor.
- ☑ Support for both graphic and text windows.
- ☑ All the tools that let you build your own expert systems and **AI** applications with unprecedented ease.

†$39.95
*$69.95
**$99.95

> "An affordable, fast, and easy-to-use language that will delight the newcomer . . . You experienced Prolog hackers will likewise be delighted, if not astonished, by the features and performance of the Turbo Prolog development environment.
>
> Turbo Prolog offers generally the fastest and most approachable implementation of that language."
>
> *Darryl Rubin, AI Expert*

How Turbo Prolog's new Toolbox adds 80 powerful tools and 8000 lines of source code

In keeping with Borland tradition, we've quickly added the new Turbo Prolog Toolbox™ to Turbo Prolog.

With 80 tools and 8000 lines of source code that can easily be incorporated into your own programs—and 40 sample programs that show you how to put these AI tools to work—the Turbo Prolog Toolbox is a highly intelligent, high-performance addition. ***Only $99.95!***

Turbo Prolog Toolbox features include:

- ☑ Business graphics generation: boxes, circles, ellipses, bar charts, pie charts, scaled graphics
- ☑ Complete communications package: supports XMODEM protocol
- ☑ File transfers from Reflex,* dBASE III,* 1-2-3,* Symphony*
- ☑ A unique parser generator: construct your own compiler or query language
- ☑ Sophisticated user-interface design tools
- ☑ Contains 40 example programs
- ☑ Easy-to-use screen editor: design your screen layout and I/O
- ☑ Calculated fields definition
- ☑ Over 8,000 lines of source code you can incorporate into your own programs

Tur
The most pow
compi

Our new Turbo C generates fast, tight, production-quality code at compilation speeds of more than 13,000 lines a minute!

It's the full-featured optimizing compiler everyone has been waiting for.

Switching to Turbo C, or starting with Turbo C, you win both ways

If you're already programming in C, switching to Turbo C will make you feel like you're riding a rocket instead of pedaling a bike.

If you're never programmed in C, starting with Turbo C gives you an instant edge. It's easy to learn, easy to use, and the most efficient C compiler at any price.
Only $99.95!

> "Turbo C does look like What We've All Been Waiting For: a full-featured compiler that produces excellent code in an unbelievable hurry . . . moves into a class all its own among full-featured C compilers . . . Turbo C is indeed for the serious developer . . . One heck of a buy—at any price."
>
> *Michael Abrash, Programmer's Journal*

bo C: **NEW!**
erful optimizing ler ever

Sieve benchmark

	Turbo C	Microsoft® C
Compile time	*2.4*	13.51
Compile and link time	*4.1*	18.13
Execution time	*3.95*	5.93
Object code size	*239*	249
Execution size	*5748*	7136
Price	*$99.95*	$450.00

Benchmark run on an IBM PS/2 Model 60 using Turbo C version 1.0 and the Turbo Linker version 1.0; Microsoft C version 4.0 and the MS overlay linker version 3.51

Technical Specifications

- Compiler: One-pass optimizing compiler generating linkable object modules. Included is Borland's high-performance Turbo Linker.™ The object module is compatible with the PC-DOS linker. Supports tiny, small, compact, medium, large, and huge memory model libraries. Can mix models with near and far pointers. Includes floating point emulator (utilizes 8087/80287 if installed).
- Interactive Editor: The system includes a powerful, interactive full-screen text editor. If the compiler detects an error, the editor automatically positions the cursor appropriately in the source code.
- Development Environment: A powerful "Make" is included so that managing Turbo C program development is highly efficient. Also includes pull-down menus and windows.
- Links with relocatable object modules created using Borland's Turbo Prolog into a single program.
- Inline assembly code.
- Loop optimizations.
- Register variables.
- ANSI C compatible.
- Start-up routine source code included.
- Both command line and integrated environment versions included.
- License to the source code for Runtime Library available.

Join more than 100,000 Turbo C enthusiasts. Get your copy of Turbo C today!

Minimum system requirements: All products run on IBM PC, XT, AT, PS/2, portable and true compatibles. PC-DOS (MS-DOS) 2.0 or later. 384K RAM minimum. Basic Telecom and Editor Toolboxes require 640K.

Borland International
4585 Scotts Valley Drive, Scotts Valley, CA 95066
Telephone: (408) 438-8400 Telex: 172373

Why more than 600,000 programmers worldwide are using Turbo Pascal today

The irresistible force behind Turbo Pascal's worldwide success is Borland's advanced technology. We created a compiler so fast, that Turbo Pascal® is now the worldwide standard. And there are more tools for Turbo Pascal than for any other development environment in the world.

You'll get everything you need from Turbo Pascal and its 5 Toolboxes

Turbo Pascal and Family are all you'll ever need to perfect programming in Pascal.

If you've never programmed in Pascal, you'll probably want to start with Turbo Pascal Tutor® 2.0,† and as your expertise quickly grows, add Toolboxes like our

- Database Toolbox®*
- Editor Toolbox®*
- Graphix Toolbox®*
- GameWorks®*

and our newest,

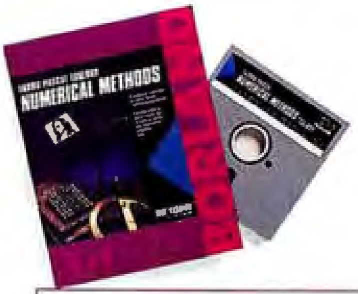
- Numerical Methods Toolbox™**

And because Turbo Pascal is the established worldwide standard, 3rd party, independent non-Borland developers also offer an incredible array of programs for Turbo Pascal.

“Borland International's Turbo Pascal took the programming world by storm. A great compiler combined with a good editor at an astounding price, the package quickly came to be called, simply, Turbo—and has sold more than 500,000 copies.

Stephen Randy Davis, PC Magazine

Language deal of the century. *PC Magazine*”



For Scientists and Engineers: Turbo Pascal Numerical Methods Toolbox

The Numerical Methods Toolbox is a complete collection of Turbo Pascal routines and programs. Add it to your development system and you have the most comprehensive and powerful numerical analysis capabilities—at your fingertips!

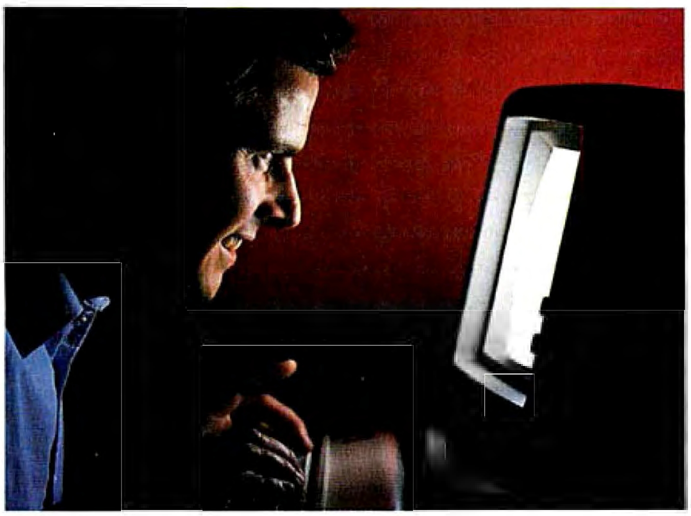
The Numerical Methods Toolbox is a state-of-the-art mathematical toolbox with these ten powerful features:

- Zeros of a function
- Interpolation
- Differentiation
- Integration
- Matrix Inversion
- Matrix Eigenvalues
- Differential Equations
- Least Squares
- Fourier Transforms
- Graphics

Each module comes with procedures that can be easily adapted to your own program. The Toolbox also comes complete with source code. So you have total control of your application.

Only $99.95!

Turbo C, Turbo Basic, Turbo Pascal and Turbo Prolog: technical excellence



" Borland International's Turbo Pascal, Turbo Basic and Turbo Prolog automatically identify themselves, by virtue of their 'Turbo' forenames, as superior language products with a common programming environment. The appellation also means to many PC users a 'must have' language. To us Turbo C looks like a coup for Borland.

Garry Ray, PC Week "

BI-1131B

Turbo Basic introduces its powerful new Telecom, Editor and Database Toolboxes

NEW!

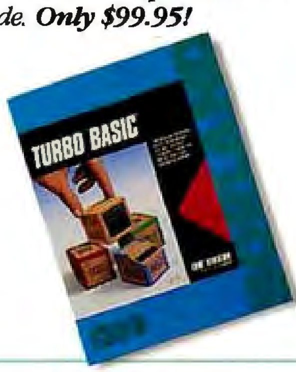
Turbo Basic® is the breakthrough you've been waiting for. The same power we brought to Pascal with Turbo Pascal has now been applied to BASIC with Turbo Basic.

Compatible with BASICA, Turbo Basic is the high-performance, high-speed BASIC you'd expect from Borland.

Basically, Turbo Basic is all you need

It's a complete development environment which includes an incredibly fast compiler, an interactive editor and a trace debugging system. It outperforms all its rivals, and because it's compatible with BASICA, you probably already know how to use it.

Includes a free MicroCalc™ spreadsheet complete with source code. ***Only $99.95!***



A technical look at Turbo Basic

- ☑ Full recursion supported
- ☑ Standard IEEE floating-point format
- ☑ Floating-point support, with full 8087 (math co-processor) integration. Software emulation if no 8087 present
- ☑ Program size limited only by available memory (no 64K limitation)
- ☑ VGA, CGA, and EGA support
- ☑ Access to local, static, and global variables
- ☑ Full integration of the compiler, editor, and executable program, with separate windows for editing, messages, tracing, and execution
- ☑ Compile, run-time, and I/O errors place you in the source code where error occurred
- ☑ New long integer (32-bit) data type
- ☑ Full 80-bit precision
- ☑ Pull-down menus
- ☑ Full window management

> "Borland has created the most powerful version of BASIC ever."
>
> *Ethan Winer, PC Magazine*

NEW!

Telecom Toolbox is a complete communications package which takes advantage of the built-in communications capabilities of BASIC—use as is or modify.

- Pull-down menus and windows
- XMODEM support
- VT 100 terminal emulation
- Captures text to disk or printer
- PhoneBook file
- 300, 1200, 2400 baud support
- Supports script files
- Fast screen I/O
- Supports most of XTalk's command set
- Manual dial and redial options

Use Telecom Toolbox to embed communications capabilities into your own programs and/or build your own communications package. Source code included for all Toolbox code and sample programs. ***Only $99.95!***

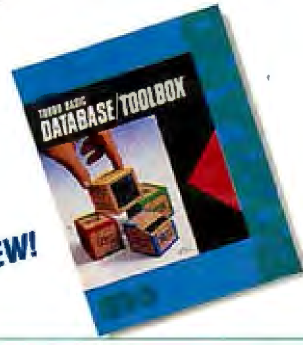
For the dealer nearest you or to order by phone call

(800) 255-8008

in CA (800) 742-1133 in Canada (800) 237-1136

BORLAND
INTERNATIONAL

Circle 37 on Reader Service Card (Dealers: 38)



NEW!

Database Toolbox means that you don't have to reinvent the wheel each time you write new Turbo Basic database programs.

- ☑ "Trainer" shows you how B+ trees work. (Simply key in sample records and you'll see your index being built.)
- ☑ Turbo Access instantly locates, inserts or deletes records in a database—using B+ trees.
- ☑ Turbo Sort sorts data on single items or on multiple keys and features virtual memory management for sorting large data files.

Source code included.

Only $99.95!

NEW!

Editor Toolbox is all you need to build your own text editor or word processor. Includes source code for two sample editors.

First Editor is a complete editor ready to include in your programs, complete with windows, block commands and memory-mapped screen routines.

MicroStar™ is a full-blown text editor with a complete pull-down menu user interface, and gives you

- Wordwrap
- Undo last change
- Auto-Indent
- Find and Find/Replace with options
- Set left/right margins
- Block mark, move and copy
- Tab, insert, overstrike modes, line center etc.

Includes source code.

Only $99.95!

BI-1131B

MORE POINTS OF VIEW
The ATI EGA WONDER — the world's fastest selling EGA card — now has MORE POINTS OF VIEW.
Lower Cost. Simple Operation.
Higher Resolutions. Any Monitor.
Automatic Software Selection.
Our buyers demand the best.
ATi ®
TECHNOLOGIES INC.
Technology you can Trust.
EGAWONDER™
Any Software
• 800x560 (1)
• 752x410 (1)
• 640x480 VGA (1)
• 640x350 EGA
• 320x200 CGA
• 720x350 MDA
• 720x350 Herc.
• 132 Columns
Any Monitor
• MultiSync
• EGA Color
• RGB Color
• 25 kHz Color
• TTL Monochrome
• Composite Mono
• PC Portable
• Compaq Portable (2)
• Polaroid Palette
Any Time
• Windows
• GEM
• AutoCAD
• 1-2-3
• Symphony
• PageMaker
• Ventura
• VTerm
• SmarTerm
• WordPerfect
• WordStar & more
Now available from all major computer stores. Call us today at (416) 756-0711 more information.
ATI Technologies Inc. 3761 Victoria Park Ave., Scarborough, Ontario, Canada M1W 3S2 Tel: (416) 756-0711 Fax: (416) 756-0720 Tlx: 06-966640 (ATI TOR)
Circle 22 on Reader Service Card

VOLUME 12, NUMBER 12, 1987

BYTE

Inside the IBM PCs

COVER ILLUSTRATED BY ROBERT TINNEY

BYTE (ISSN 0360-5280) is published monthly with additional issues in June and October by McGraw-Hill Inc. Founder: James H. McGraw (1860–1948). Executive, editorial, and advertising offices: One Phoenix Mill Lane, Peterborough, NH 03458, phone (603) 924-9281. Office hours: Monday through Thursday 8:30 AM–4:30 PM, Friday 8:30 AM–1:00 PM, Eastern Time. Address subscriptions to BYTE Subscriptions, P.O. Box 6821, Piscataway, NJ 08855. Postmaster: send address changes, USPS Form 3579, undeliverable copies, and fulfillment questions to BYTE Subscriptions, P.O. Box 6821, Piscataway, NJ 08855. Second-class postage paid at Peterborough, NH 03458 and additional mailing offices. Postage paid at Winnipeg, Manitoba. Registration number 9321. Subscriptions are $22 for one year, $40 for two years, and $58 for three years in the U.S. and its possessions. In Canada and Mexico, $25 for one year, $45 for two years, $65 for three years. $69 for one year air delivery to Europe. 31,000 yen for one year air delivery to Japan, 15,600 yen for one year surface delivery to Japan, $37 surface delivery elsewhere. Air delivery to selected areas at additional rates upon request. Single copy price is $3.50 in the U.S. and its possessions, $4.25 in Canada and Mexico, $4.50 in Europe, and $5 elsewhere. Foreign subscriptions and sales should be remitted in U.S. funds drawn on a U.S. bank. Please allow six to eight weeks for delivery of first issue. Printed in the United States of America.

Address editorial correspondence to: Editor, BYTE, One Phoenix Mill Lane, Peterborough, NH 03458. Unacceptable manuscripts will be returned if accompanied by sufficient postage. Not responsible for lost manuscripts or photos. Opinions expressed by the authors are not necessarily those of BYTE.

Subscription questions or problems should be addessed to: BYTE Subscriber Service, P.O. Box 6821, Piscataway, NJ 08855.

EDITORIAL AND BUSINESS OFFICE:
One Phoenix Mill Lane, Peterborough, New Hampshire 03458, (603) 924-9281.
West Coast Branch Offices: 425 Battery St., San Francisco, CA 94111, (415) 954-9718; 3001 Red Hill Ave., Building #1, Suite 222, Costa Mesa, CA 92626, (714) 557-6292.
New York Branch Editorial Office: 1221 Avenue of the Americas, New York, NY 10020, (212) 512-3175.
BYTEnet: (617) 861-9764 (set modem at 6-1-N or 7-1-E; 300 or 1200 baud). **Fax:** (603) 924-7507. **Telex:** (603) 924-7861.
SUBSCRIPTION CUSTOMER SERVICE: Non-U.S. (201) 837-1315; inside U.S. (outside N.J.) 1-800-423-8272; (inside N.J.) 1-800-367-0218.

Officers of McGraw-Hill Information Systems Company: President: Richard B. Miller. Executive Vice Presidents: Frederick P. Jannott, Construction Information Group; Russell C. White, Computers and Communications Information Group; J. Thomas Ryan, Marketing and International. Senior Vice Presidents–Publishers: Laurence Altman, Electronics; David J. McGrath, Engineering News-Record. Group Vice Presidents: J. Burt Totaro, BYTE; Frank A. Shinal, Dodge; Peter B. McCuen, Communications Information. Vice Presidents: Robert D. Daleo, Controller; Fred O. Jensen, Planning and Development; Michael J. Koeller, Human Resources; Talat M. Sadiq, Systems Planning and Technology.

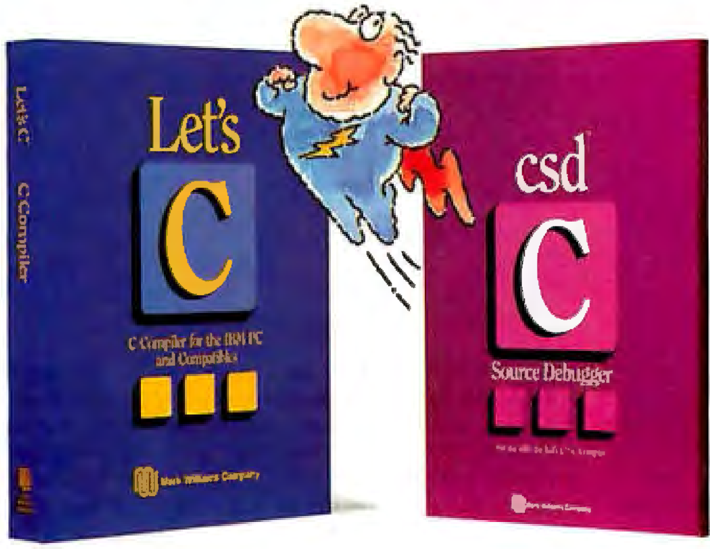
Officers of McGraw-Hill Inc.: Harold W. McGraw Jr., Chairman; Joseph L. Dionne, President and Chief Executive Officer; Robert N. Landes, Executive Vice President, General Counsel and Secretary; Walter D. Serwatka, Executive Vice President and Chief Financial Officer; Shel F. Asen, Senior Vice President, Manufacturing; Robert J. Bahash, Senior Vice President, Finance and Manufacturing; Frank D. Penglase, Senior Vice President, Treasury Operations; Ralph R. Schulz, Senior Vice President, Editorial; George R. Elsinger, Vice President, Circulation.

PRESENTING THE DIFFERENCE BETWEEN FAST COMPILING AND FAST PROGRAMMING.

For compiling speed, you can't do better than Let's C. But to really speed up programming you can't do without the powerful source level debugger, *csd*.

If you want the power, portability and flexibility of C, start with the complete compiler, Let's C. For utilities, editor, compiling speed and fast, dense code, Let's C has it all.

But to get your programs up and running you need more. Because even the fastest compiler can't outrun bugs. You need the revolutionary C Source Debugger, *csd*.



CUT DEVELOPMENT TIME IN HALF WITH *csd*

csd lets you bypass the time consuming frustrations of debugging–like long dumps and clunky assembler. With *csd*, you actually debug in C. You learn faster because you watch your program run in C. You finish faster because *csd* combines the speed of a compiler with the interactive advantages of an interpreter. The end result? Development time is sliced in half.

LIMITED TIME OFFER FREE *csd* WITH LET'S C!

REVIEWERS ARE RAVING ABOUT LET'S C AND *csd*.

"Let's C is an inexpensive, high-quality programming package... with all the tools you will need to create applications."
–William G. Wong, *BYTE*, August 1986.

"The performance and documentation of the $75 Let's C compiler rival those of C compilers for the PC currently being sold for $500...highly recommended..."
–Marty Franz, *PC TECH JOURNAL*, August 1986.

"csd *is close to the ideal debugging environment...a definite aid to learning C and an indispensable tool for program development."*
–William G. Wong, *BYTE*, August 1986.

"This is a powerful and sophisticated debugger built on a well-designed, 'serious' compiler."
–Jonathon Sachs, *Micro/Systems Journal*, April, 1986

LET'S C AND *csd* FEATURES

NEW VERSION 4.0!

Let's C:

- Now compiles twice as fast
- Integrated edit-compile cycle: editor automatically points to errors
- Includes both small and large memory model
- Integrated environment or command line interface
- 8087 sensing and support
- Documentation features new lexicon format
- MS-DOS object compatible
- New make utility
- Fast compact code plus register variables
- Full Kernighan & Ritchie C and extensions
- Full UNIX compatibility and complete libraries
- Many powerful utilities including make, assembler, archiver, cc one-step compiling, egrep, pr, tail, wc
- MicroEMACS full screen editor with source included
- Supported by dozens of third party libraries
- For the IBM-PC and Compatibles
- Not copy protected

Sieve Benchmark
(Compile time in seconds)
Let's C: **2.8** (On 512K 6Mhz IBM-AT)
Turbo C: **3.89** (As advertised)

csd:

- Large and small memory model
- Debug in C source code, not assembler
- Monitor variables while tracing program
- Does not change program speed or size
- Provides separate source, evaluation, program and history windows
- On-line help screens
- Can interactively evaluate any C expression
- Can execute any C function in your program
- Trace back function
- Ability to set trace points
- Not copy protected

START TO FINISH, THERE'S NO BETTER ENVIRONMENT.

Get started with the right C compiler and you'll have everything you need for development–including source level debugging. On top of it all, Let's C and *csd* are today's best values in professional C programming tools. And most reliable: Mark Williams C compilers have been sold with DEC, Intel and Wang computers since 1981.

60 DAY MONEY BACK GUARANTEE

Mark Williams gives you a full 60 days to find out just how good Let's C and *csd* really are–or your money back.

So if you want more than a fast compiler–if you want your programs up and running fast, ask for Let's C and *csd*. You'll find them at your software dealer's, in the software department of your favorite bookstore, through the Express Program at over 5500 Radio Shacks or you can order now by calling 1-800-MWC-1700.*

*In Illinois call, 1-312-472-6659.

Mark Williams Company
1430 West Wrightwood, Chicago, Illinois 60614

MARK WILLIAMS LET'S C AND *csd*. ONLY $75 EACH.

Circle 128 on Reader Service Card

Editorial

The Meaning of SAA

First-of-the-year rumors about IBM's Systems Application Architecture (SAA) hinted at a major new look for IBM software, possibly with the introduction of what turned out to be the Personal System/2 machines. The new look failed to appear in any substantive way, however, as only glimpses of a prototype Presentation Manager surfaced during the PS/2 debut.

Finally, in early May, IBM released a publication called *Systems Applications Architecture: An Overview*. This booklet offers a peek at what SAA may offer. Unfortunately, the peek is through a fog of banal generalities that really don't say what SAA is or what it is supposed to be. Instead, the booklet offers a vision of what SAA might become—when IBM gets around to developing it.

After the PS/2 introduction, most people focused on the pros and cons of OS/2, and SAA receded into the shadows. I have an aversion to things that lurk in shadows, so let's look at SAA more closely.

What Is SAA?

The short definition of SAA is a common user interface, programmatic interface, and communications interface for three IBM computer lines—mainframe computers, minicomputers, and personal computers. The long definition adds only a little detail to each of the three interface categories.

Based on this definition, SAA is a very ambitious idea. Basically, IBM wants programs running on three different hardware platforms under three different operating systems to look the same to users, programmers, and communications channels. The accomplishment of this will require raising software up to yet another level of abstraction far removed from the computers the software runs on. Moving to another level of abstraction usually slows down programs and requires more memory.

What will this software look like? For users, it presumably will look something like the Presentation Manager, which in turn looks like its parent, Microsoft Windows. For programmers, SAA programs will use standard languages like C, FORTRAN, and COBOL, plus "application generators" and a "procedures language," probably a derivative of REXX (REXX uses a syntax similar to English and is designed for nonprogrammers); all these languages will call a standard set of operating-system and user-interface services.

The communications interface is more complex, incorporating protocol support for the 3270 Data Stream, Document Content Architecture, Intelligent Printer Data Stream, SNA Distribution Services, and many other network/session services and data link control systems.

Future versions of Lotus 1-2-3 and dBASE will therefore look the same, whether you are using them on a PC or a 370 terminal. Plus, you'll be able to send a file created by an application on your PC to a mainframe, where it can be read by the mainframe version of the same application.

Sound Familiar?

The grand scheme of SAA has much appeal—programs that you can feel comfortable with when you change computers, portability across machines, and universal data. But I get skeptical when someone says to me, "Let's all do it this way; it's cool, trust me." SAA seems to be IBM's way of making us all conform to a single way of producing software.

Of course, this idea is not new nor foreign to the personal computer—the Macintosh imposes a similar "do it my way" philosophy. You can get away with imposing a set of standard interfaces if the standards are good enough to be the only game in town.

The best way to impose this kind of software conformity is to control the programming environment. Instead of providing operating systems that simply perform file and memory management, build environments with several hundred system calls that control everything from screen management to communications. Come to think of it, that sounds just like OS/2 Presentation Manager.

We may be dealing here with a simple trade-off situation; you trade off your bit-level control of the computer in return for sophisticated, ready-made routines that make creating software easier; and all the software looks alike.

Are we going to get any future VisiCalcs out of such a closely controlled environment?

Get There

In rural New England, it is often a travel reality that "you can't get there from here." An obvious first question about SAA is, how do we get there? What are the intermediate destinations on the route?

Unfortunately, IBM doesn't know yet. It promises that the conventions of SAA "will be published in 1987." As I write this in early September, only one document on SAA has been published. And this booklet is short on specifics. Furthermore, SAA will be a moving target right from the beginning. The overview booklet states, "In its initial form, SAA and its current IBM product implementations offer a starting point. It is IBM's intent *that both definitions and implementations increase over time in an evolutionary process*" (emphasis mine).

Apparently, then, SAA will be whatever IBM says it is at any given time.

From Here

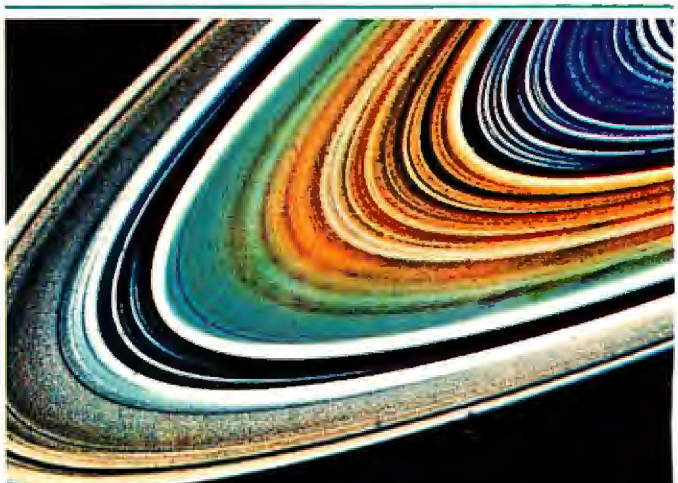
Two things are clear about SAA. First, it will be a long time in coming. Second, if you don't work for IBM—or have to contend with the three-tiered computer arrangement SAA is really for—you can probably ignore it. Someday, we may have a "universal" computer that runs software that we all use and that must therefore always appear the same. But that day is still years, maybe decades, away. In the meantime, we've all got to make a living, and using or building a slightly better/faster/cheaper spreadsheet/word processor/database is still the best way to get to the bank from here.

—G. Michael Vose
Senior Technical Editor

CONNECT MAINFRAME TAPE DRIVES TO YOUR PC!

Whether your tape data comes from millions of miles away by satellite or from just down the hall in accounting, our "TAPE CONNECTION" can read and write ½" 9-track magnetic tapes using your PC! Over 500 million reels of magnetic tape are in use by most mainframe and minicomputer systems. For more than 25 years, ½" 9-track tape has been the standard worldwide for storing and retrieving large data files. Why not let your PC and our tape system assist you in using the vast resources of the tape data world?

Supporting 800 NRZI, 1600 PE and 6250 GCR, our file transfer software processes labeled or unlabeled tapes from most computer systems, including IBM OS/DOS, DEC/VAX, UNISYS, Honeywell, Burroughs, NCR, and HP. Large multivolume tape reels can be transferred to disk at rates up to 5 MB/minute! Backup and restore capabilities work well with Novell networks for IBM XT/AT. With our Tape Data Extraction software, you can read packed fields, select and extract specific records and fields, such as payroll or personnel data, tax structure statistics, department records, as well as import data into LOTUS 123. You'll find so many applications for your particular business. You could even use your PC to read and display NASA's Voyager II mission data tapes, such as this highly enhanced image of Saturn's rings recorded from a distance of 8.9 million kilometers (from the tape library shown above).



We have years of experience with IBM mainframes using magnetic tape, so we're qualified to assist you in implementing and supporting your application. Since 1981, we have supplied thousands of conversion systems throughout the world, including most of the *Fortune 500* companies. Our customer support personnel are available to answer your questions, free of charge. Our high volume allows us to offer low prices on Anritsu, Cipher, Kennedy and Qualstar equipment. Systems come complete and ready to use with controller card, cables, software and drive. Ranging from $2995 to $8995, we have a system for you, so call us today!

Dealer and volume discounts available.

See us at Comdex Las Vegas
Booth No. 33 WIGI—163 West Hall

FLAGSTAFF ENGINEERING

1120 Kaibab • Flagstaff, Arizona 86001 • 602-779-3341
Compusol-Europe • 12 Rue Rosenwald • 75015 Paris
Tel 530.07.37 • Telex 205431F

Circle 83 on Reader Service Card (Dealers: 84)

A Lot Of Promises.

Take a close look at these two machines. At 10 MHz, operating at one wait state, you might believe IBM's® Personal System/2™ Model 50 is one of the fastest 80286 computers available. Fact is, an InfoWorld benchmark test ranks the AST Premium/286's™ CPU performance number one.

You might also think IBM's system is the first to take advantage of powerful multitasking operating system software. And you'd be wrong again. When we introduced the AST Premium/286 a year ago with advanced FASTslot™ architecture, we designed a home for Microsoft's® MS OS/2.™ In fact, it delivers all zero wait-state memory for MS OS/2.

Of course, MS OS/2 may not be available for a while. Which is okay, if you have an AST Premium/286. Built into every system is AST's Enhanced Expanded Memory (EEMS), allowing EEMS software such as Windows™ 2.0 and DESQview™ to multitask existing applications... today.

So, hold on to any of your existing off-the-shelf application software. As long as it's AT®-compatible, it will run on the AST Premium/286.

What can the competition offer you today? Promises for the future. We can't wait that long. And neither should you.

If you want more than promises, make a commitment

Benchmark Test Results
For Selected Performance Computers

CPU measures main processor performance relative to the 6-MHz (Model 099) IBM PC AT. Hard disk performance is tested for sequential and random data access.

SYSTEM (80286-BASED PCS) (Clock speed in MHz/No. of wait states)	**CPU**	**Hard Disk** (sequential)	**Hard Disk** (random)
AST Premium/286 (10/0)	***2.25***	***1.41***	***2.12***
IBM PC AT (6/1)	1.00	1.00	1.00
IBM PC XT-™286 (6/0)	1.32	1.33	1.03
IBM PC AT (8/1)	1.37	1.17	1.40
IBM PS/2 Model 50 (10/1)	1.71	1.70*	1.19*
IBM PS/2 Model 60 (10/1)	1.72	2.02	1.67

*With RAM cache: seq. 1.92, ran. 1.03
Source: InfoWorld Hardware Benchmark System, as published in InfoWorld May 11, 1987

Circle 116 on Reader Service Card

What's New

VGA-Compatible Monitor

Mitsubishi's XC-1429C is a VGA-compatible monitor that has a 14-inch (diagonal) screen. With .28-millimeter dot pitch and a fixed horizontal scanning frequency of 31.5 kilohertz, it features a resolution of 640 by 480 pixels, a vertical scanning frequency of 60 or 70 hertz, and the ability to display an infinite number of colors.

The XC-1429C also includes a precision in-line gun, a 120-volt power supply, and a 15-pin shrink connector cable. An optional tilt-and-swivel base is available.
Price: $685.
Contact: Mitsubishi Electronics America Inc., Computer Peripherals Division, 991 Knox St., Torrance, CA 90502, (213) 515-3993.
Inquiry 751.

Mitsubishi's new monitor is VGA-compatible.

Publish with Byline

Ashton-Tate's Byline is a desktop-publishing program designed for business users with no special knowledge of graphic arts and typography. It features a WYSIWYG (what you see is what you get) display and a dBASE merge function that lets you import dBASE III Plus databases into prestyled forms.

In addition, Byline enables you to directly import and export MultiMate, WordPerfect, and WordStar files. You can also import files from Lotus 1-2-3, Symphony, and several paint programs.

Five fonts, with adjustable type sizes between 8 and 144 points, are provided. Also included are lines and boxes, autoleaders and repeating characters, automatic kerning, multiple left and right master pages, and automatic text flow.

Byline runs on IBM PCs and compatibles with 384K bytes of RAM and an EGA, CGA, Hercules, or Hercules Plus graphics card.
Price: $295.
Contact: Ashton-Tate, 20101 Hamilton Ave., Torrance, CA 90502-1319, (213) 329-8000.
Inquiry 752.

Multilingual, Scientific Word Processor

ChiWriter "The Scholar's Edition," based on Horstmann Software Design's scientific word-processing program, is a multilingual, multifont word-processing program that lets you write left to right or right to left.

ChiWriter "The Scholar's Edition" includes standard Roman, foreign (all western European and Scandinavian languages), classical-koine-mode Greek, and biblical-modern Hebrew alphabets. You can also design your own fonts.

The program includes footnotes, composite characters, user-specified document-saving intervals, columns, macros, overlapping super- and subscript levels, automatic pagination, headers and footers, and microspacing capabilities.

ChiWriter "The Scholar's Edition" runs on IBM PCs and compatibles with MS-DOS 2.0 or higher, 256K bytes of RAM, and a CGA. It supports some 9-pin printers. Support for Hercules monochrome, the EGA, 24-pin printers, and Hewlett-Packard LaserJet printers is optional.
Price: $99.95; Hercules monochrome, EGA, and 24-pin-printer support, $19.95; HP LaserJet support, $49.95.
Contact: Paraclete Computer & Software, 1000 East 14th St., Suite 187, Plano, TX 75074, (214) 578-8185.
Inquiry 753.

QuadStar LAN Features Easy Setup

Quadram's QuadStar is a local-area network package that uses phone-type cabling with modular connectors. It can accommodate up to 50 microcomputers and can provide gateways to minicomputers and mainframes.

On the hardware end of the system are three available boards that plug into IBM PCs, XTs, ATs, and compatibles. The QS-PH6 Personal Hub plugs into a full slot in the network-manager computer. It provides five external ports for connecting to file servers and users, and one internal adapter port. It also provides an 8K-byte RAM buffer and 8K bytes of diagnostic tests.

The QS-188 Intelligent Adapter Board is for use with file servers, which provide resources such as software, printers, and data that is shared by network users. It plugs into a full-size slot and includes an Intel 80188 microprocessor running at 8 megahertz that offloads network software processing from the host CPU. It also provides a 64K-byte RAM buffer.

The QS-100 Adapter Board is a half-size expansion board for network users. It provides an 8237 microprocessor that eliminates the need for sharing direct-memory-access channels.

Tapestry is QuadStar's NetBIOS-compatible operating system. It uses icons and resides on the network manager or a file server.

QuadStar conforms to the

continued

IEEE 802.3 1Base5 physical interface standard and transfers data at 1 megabit per second using the CSMA/CD baseband. The maximum distance between a user and the network manager is 800 feet, using 24-gauge telephone wire.

The network manager and file server require hard disk drives and 640K bytes of RAM with at least 160K bytes of overhead. Network users require 512K bytes of RAM with a 95K-byte overhead.

Quadram's QuadStarter Kit is for a network linking two microcomputers, expandable to six. It includes Tapestry, the QS-PH6 Personal Hub board for the network manager, the QS-100 board for a user, and 50 feet of shielded twisted-pair cable.
Price: QuadStarter Kit, $1095; QS-100, $375; QS-188, $495; QS-PH6, $575.
Contact: Quadram, One Quad Way, Norcross, GA 30093-2919, (404) 923-6666.
Inquiry 754.

Multiconfiguration NCR System

Using split-card architecture, NCR's PC810 is an IBM PC AT-compatible system that comes in 35 different configurations. The main processor and main memory are on one expansion board, while a second board provides the video adapter, disk drive controllers, extended memory, and serial and parallel ports. The boards are connected by the eight-slot system bus.

The heart of the basic PC810 configuration includes an 80286-10 microprocessor running at 6 or 10 MHz, 640K bytes of RAM, a 5¼-inch 1.2-megabyte floppy disk drive, a real-time clock/timer with battery, and a keyboard with 30 function keys and separate cursor pad. The PC810 also has six expansion slots for additional boards and an 80287 math coprocessor socket.

NCR's AT-compatible comes in 35 different configurations.

Among the options available are 20-, 30-, 44-, and 70-megabyte hard disk drives and half-height 720K-byte or 1.44-megabyte 3½-inch floppy disk drives. You can choose EGA, CGA, monochrome, or no graphics support on the second board. Memory is expandable to 1 megabyte on the processor board; you can have up to 16 megabytes total RAM.

NCR DOS 3.2 and diagnostic software is included. The PC810 measures 6.1 by 21.2 by 16.5 inches and weighs about 38 pounds.
Price: Basic configuration with no graphics adapter, $2950; with 20-megabyte hard disk drive, $3325; with 30-megabyte hard disk drive, $4220; with CGA and monochrome graphics, $3200; with CGA and monochrome graphics and 20-megabyte hard disk drive, $3525; with CGA and monochrome graphics and 30-megabyte hard disk drive, $4420; with EGA, $3400; with EGA and 44-megabyte hard disk drive, $4820.
Contact: NCR Corp., Personal Computer Division, Dayton, OH 45479, (513) 445-2078.
Inquiry 755.

Hard Disk Expander Doubles Capacity

Konan's KXP-230 is a half-slot hard disk controller that the company claims doubles your hard disk capacity without using run-length-limited (RLL) formatting. Data is stored on the disk in standard modified-frequency-modulation (MFM) format, then compressed and compacted. You can use the KXP-230 with any ST-506/412-compatible hard disk drive up to 302 megabytes in size.

When you install the KXP-230, an expanded disk called an EDISK is created automatically. The hard disk is partitioned, with one partition containing MS-DOS and the KXP-230 software; the EDISK partition takes up the rest of the disk space. As an example, Konan says a reformatted 20-megabyte hard disk will have a 1-megabyte DOS partition and an EDISK with up to 38 megabytes.

A disk cache, disk error correction, and fragmentation-control capabilities are also provided. The fragmentation control lets you add data to a file in contiguous tracks, rather than in the next available disk location.
Price: KXP-230Z (for IBM PC or PC XT), $249; KXP-230ZA (for IBM PC AT), $299; KXP-230ZT (for Tandy 1000), $249.
Contact: Konan Corp., 4720 South Ash Ave., Tempe, AZ 85282, (602) 345-1300.
Inquiry 756.

Baler 3.23

Baler 3.23 runs 38 percent faster than earlier versions, according to the manufacturer. Baler 3.23 converts Lotus 1-2-3 worksheets into stand-alone programs that support the operational commands for Lotus, including macros, arrows, and user inputs, but not the commands for developing a worksheet. The "baled" (executable) programs, written in BASIC, run without Lotus. The worksheet formulas in baled programs can't be changed.

Baler 3.23 runs on the IBM PC, XT, AT, and compatibles with 256K bytes of RAM, MS-DOS or PC-DOS 2.0 or higher, and one 1.2-megabyte floppy disk drive. (A hard disk drive is recommended.) Baler also requires Lotus 1-2-3 (versions 1A, 2, or 2.01) or a work-alike spreadsheet program and either Microsoft's QuickBASIC Compiler or IBM's BASIC Compiler version 2.0.
Price: $495; Microsoft's QuickBASIC Compiler, $99.
Contact: Brubaker Software, 8825 North County Line Rd. E, Lafayette, IN 47905, (317) 564-2584.
Inquiry 757.

Simon and Schuster Speed Reading

Speed Reading Tutor IV provides reading lessons that last about an hour each and are designed to be paced two or three days apart, Simon and Schuster reports.

The first lesson tests your initial speed and comprehension. The program then customizes the next eight lessons, depending on your current

continued

ability. Each lesson begins with a practice reading. You can enter the reading speed you want to attain, and the program produces an audible prompt telling you when to go on to the next page. A reading timer determines your reading speed.

Following each practice reading, you take a comprehension quiz. A graph of your speed and comprehension scores follows. A lesson-by-lesson progress chart is also provided.

Speed Reading Tutor IV also includes Eyerobics and T-scope exercises to assist you in improving your peripheral vision. The Eyerobics exercise flashes different or identical patterns on the screen, which become more intricate and farther apart when you respond correctly. During the T-scope exercise, numbers, phrases, and stories flash by one line at a time, gradually increasing in length and number of lines.

Speed Reading Tutor IV runs on the IBM PC, XT, AT, and compatibles with PC-DOS 1.1 or higher, 128K bytes of RAM, an 80-column color or monochrome monitor, and two floppy disk drives or one floppy disk drive and one hard disk drive.
Price: $39.95.
Contact: Simon and Schuster Software, One Gulf & Western Plaza, New York, NY 10023, (212) 373-8882.
Inquiry 758.

Digitizer Measures It All

You can measure areas, distances, perimeters, lengths of curving lines, angles, slopes, and other two-dimensional figures with the Sigma-Scan 3.0 Measure-System. It includes a 12- by 12-inch opaque digitizing tablet and software.

The program also enables you to perform point, stream, and incremental *x,y* digitizing and object tallying. You can collect multiple mea-

Sigma-Scan measures any type of two-dimensional figure.

surements simultaneously and set your own measurement standards.

Measurements are automatically loaded into an internal 25-column by 65,000-row worksheet as they are collected. You can then perform size, sum, mean, min, max, standard deviation, standard error, confidence intervals, t-tests, linear regressions, screen plots, and other statistical functions. You can construct nonlinear piecewise calibration tables, macros, and data transforms. You can also output data in ASCII files.

The Sigma-Scan 3.0 Measurement System runs on the IBM PC, XT, AT, and compatibles with MS-DOS or PC-DOS 2.0 or higher, 512K bytes of RAM (640K bytes recommended), and two floppy disk drives or one floppy disk drive and one hard disk drive. It also runs on IBM PS/2s. It supports CGA, EGA, Hercules, and VGA cards. The software comes on either 3½- or 5¼-inch floppy disks and is not copy-protected. The software is also available separately and supports most digitizers.
Price: $1195; software only, $495.
Contact: Jandel Scientific, 2656 Bridgeway, Sausalito, CA 94965, (415) 331-3022.
Inquiry 759.

Charley is a PC Genlock

Video Charley provides genlock to an external video source to let you overlay graphics onto video. The FCC-legal NTSC output includes 64-color EGA support, programmable key colors, a selectable blanking source, and two 16-level dissolve registers. The daughterboard supports IBM standard software, including paint and graphics programs. You install it on existing EGA cards by using the 20-pin features connector. Video Charley works with the IBM PC, XT, AT, and compatibles.
Price: $749.95.
Contact: Progressive Image Technology, 322 East Bidwell St., Folsom, CA 95630, (916) 985-7501.
Inquiry 760.

C-Z80/64180 Cross Compiler

Archimedes Software has announced the C-Z80/64180 C cross compiler for the Zilog Z80 and Hitachi 64180 microprocessor families. The program includes a C compiler, C library functions, macro assembler, linker, and librarian.

The compiler implements the ANSI-standard C enhancements, including function prototyping. It also supports the Kernighan and Ritchie C definition. Code generation for either microprocessor is switch-selectable, and you can access the 64180's full megabyte of address space. The compiler supports IEEE 32-bit single-precision floating-point library functions.

The C-Z80/64180 C cross compiler also supports trigonometric, exponential, and logarithmic math functions; four memory models; relocatable libraries; link and relocate functions; and output options for Motorola S-format, Tektronix standard hexadecimal, and Symbolic for symbolic emulator debugging.

The compiler runs on the IBM PC and compatibles with MS-DOS 2.1 or higher and 512K bytes of RAM.
Price: $995.
Contact: Archimedes Software Inc., 1728 Union St., San Francisco, CA 94123, (415) 771-3303.
Inquiry 761.

Keep Your PC Cool

Coldblue consists of two side-by-side fans attached to a plenum chamber. It ventilates your PC's card area at more than 25 cubic feet of air per minute, according to Mandrill. The company also reports that Coldblue lowers the internal operating temperature of an IBM PC by more than 20 degrees.

Coldblue mounts inside IBM PCs and XTs between the chassis and cover in front of your boards. A pin connects to the leftmost opening in the disk drive connector, letting the fan draw power from the computer. Grounding is provided on your system's speaker screw. Coldblue features a blue-light indicator visible through the front air inlet in the computer's cover.
Price: $185.
Contact: Mandrill Corp., P.O. Box 33848, San Antonio, TX 78265, (512) 341-6155.
Inquiry 762.

continued

A Faster Word

Microsoft Word 4.0 is faster, has document-retrieval capabilities, macros, and an improved user interface.

Speed increases are in scrolling, file load and save, cursor movement, and pagination features. A new switch toggles between text and graphics mode. Version 4.0's WYSIWYG graphics mode takes advantage of high-resolution graphics boards and monitors, such as the new VGA graphics modes of the IBM PS/2 machines.

Version 4.0's macro language offers conditionals, looping, and prompts. The document-management and -retrieval feature lets you assign summary sheets to each document, set up directories of documents for sorting, and choose words within documents to use as search criteria.

The user interface now has no alpha command; you use the Escape key to move in and out of editing. The cursor keys move you around in the menus and property sheets, and you can remap all 40 function-key combinations. You can also choose to display line and column counter numbers on-screen, and you can have prompt lines appear to explain menu commands. You have a choice of having border lines show or adding up to two lines on-screen.

Other tools added to Word 4.0 include redlining, a spreadsheet link, style sheets, a 130,000-word spelling checker, and the ability to draw lines and paragraph borders.

Microsoft also announced a network version of Word 4.0.

Word 4.0 runs on the IBM PC, XT, AT, and PS/2s with at least 320K bytes of RAM and two floppy disk drives or one floppy disk drive and one hard disk drive.
Price: $450.
Contact: Microsoft Corp., 16011 Northeast 36th Way, P.O. Box 97017, Redmond, WA 98073-9717, (206) 882-8080.
Inquiry 763.

The document-management and -retrieval feature in Word 4.0.

Neural-Network Simulation

SYSPRO stands for system simulation program, and it lets you simulate hierarchical dynamic systems, as well as model and simulate neural networks.

SYSPRO simulates systems that you define with four types of data in input files. Simulation run-time instructions tell the program how long to run, how often to plot points, which points to plot, and other simulation-management instructions. The model parameter data establishes values for the system models for each run. The initial state of the system defines the starting point for the system trajectory, and the real-time input data causes the system state to change. You can save a complete record of the system state at the end of the run and at specified intervals.

SYSPRO includes the simulation executive (object code) and the Evolve module, written in FORTRAN, which calls the root program of your model system. The Datagen module lets you generate input-data files containing signals in additive noise. JPlot is a BASIC program that lets you display graphics of simulation variables on-screen or produce printer plots. Batchkit is a collection of DOS batch files for compiling and linking your models into SYSPRO and for running laboratory experiments.

SYSPRO Plus also includes an executable load module, FORTRAN source files, and listings for BPNET, a back-propagation network model that can perform the nonlocal component of the generalized delta rule. This is the top level of a hierarchical system made up of Neurons. NETINP, an interface program, translates your network structure specifications and real-time inputs to the system. Neuron is a processing element model that performs the state transition function for a neuron with 40 learnable synapses, five excitatory synapses, five inhibitory synapses, and a learnable threshold level.

SYSPRO and SYSPRO Plus run on IBM PCs and compatibles with MS-DOS or PC-DOS 2.1 or higher (256K bytes of RAM on an IBM PC or XT, 512K RAM on an AT), GWBASIC, a FORTRAN compiler, text editor, and math coprocessor. An EGA is recommended.
Price: SYSPRO, $995; SYSPRO Plus, $1250.
Contact: Martingale Research Corp., 100 Allentown Pkwy., Suite 211, Allen, TX 75002, (214) 422-4570.
Inquiry 764.

Update with 3½-inch Drives

Astro's EXT1 3½-inch floppy disk drives provide either 720K bytes or 1.44 megabytes of storage space with an average access time of 94 milliseconds. Six configurations are available.

The EXT1-PS is designed for the IBM PC, XT, and compatibles that have a 37-pin D-connector on the back. It comes with a self-power adapter, interface cables, and utility software. The EXT/AT-PS is designed for IBM PC ATs and compatibles without external 37-pin D-connectors. It interfaces to your computer via a half-slot adapter card. The EXT/AT is similar to the EXT/AT-PS, except it gets its power from your computer. The EXTJr. is designed for PCjrs and also uses an adapter card. Two internal drives are available.
Price: EXT1-PS, 720K-byte model, $365; 1.44-megabyte model, $430; EXT/AT-PS, 720K-byte model, $395; 1.44-megabyte model, $460.
Contact: Astro Systems Inc., 807 Aldo Ave., #106, Santa Clara, CA 95054, (408) 727-5599 or 727-7626.
Inquiry 765.

Data Acquisition

The Pro-Data data-acquisition program is an extension of PC-DOS that can run in the background and provide real-time data to spreadsheet programs. It lets you acquire analog and digital data.

Pro-Data maintains files

continued

containing current analog input values, previous minute averages, previous hour averages, totals from analog inputs, and pulse counts from digital inputs. It also keeps a historical log of minute and hourly averages for the previous 120 samples.

Pro-Data supports up to 10 I/O device drivers. You can have up to 128 analog or frequency inputs, 128 digital inputs, 128 analog outputs, and 128 digital outputs.

The memory-resident program runs on the IBM PC, XT, AT, and compatibles with PC-DOS 2.1 and 640K bytes of RAM. A floppy disk drive and a hard disk drive are recommended.

Price: $750.

Contact: Industrial Interfaces Inc., 915 Whitaker St., Pasadena, TX 77506-2342, (713) 473-5112.

Inquiry 766.

Monitoring the Weather

If those long hours in front of your PC keep you indoors more often than you'd like, Technology Marketing has a system that will let you vicariously experience the outdoors. PC WeatherPro enables you to monitor and store measurements of barometric pressure, inside and outside temperature, rainfall, wind speed, and wind-chill temperature. The system runs in the background and lets you set alarms that will sound if critical changes occur in temperature or wind speed.

The system includes a half-slot card, solid-state barometer, electronic rain collector, anemometer and wind vane, AC power adapter, two temperature probes, and connecting cables. Software included provides both graphic and digital readings of weather measurements and calculations.

PC WeatherPro also lets you produce barometric plots, and individual screens are provided for wind, rain, and temperature detail. Data is logged with time-and-date stamps.

PC WeatherPro monitors, stores, and analyzes meteorologic data.

The system runs on all IBM PCs and compatibles; the software takes up 64K bytes of RAM.

Price: $575.

Contact: Technology Marketing Group Ltd., 4000 Kruse Way Place, Building 2, Suite 120, Lake Oswego, OR 97035, (503) 635-3966.

Inquiry 767.

Golden Retriever Pup

Golden Retriever Pup uses pattern recognition to scan files and locate information in much the same way as the original Golden Retriever, but Pup searches floppy disks only. It does not distinguish between uppercase and lowercase letters, and it can match by comparing the spelling and ordering of words within a phrase. It doesn't interpret meanings and doesn't recognize synonyms.

During the search, the program uses a proprietary search algorithm and calculates a score from 0 to 100 for all the potential text patterns in the files, with 100 being an exact match. You define a minimum score, and if a pattern equals or exceeds the score, Pup reports the match and saves the pattern. Another feature is the ability to search only the files you want searched. The program offers wild-card–character, file-date, and subdirectory options.

On the IBM PC XT, Golden Retriever Pup can search through 2000 characters per second, according to S K Data.

The program requires an IBM PC, XT, AT, or compatible with MS-DOS or PC-DOS 2.0 or higher. It is menu-driven and not copy-protected.

Price: $99.

Contact: S K Data Inc., P.O. Box 413, Burlington, MA 01803, (617) 229-8909.

Inquiry 768.

A Braille Interface Terminal

The Braille Interface Terminal consists of a 20-cell braille display that serves as a window to the computer screen, a braille keyboard, a joystick, a full-slot card, and software that provides commands to maneuver the window around the screen.

The Braille Interface Terminal runs on the IBM PC, XT, AT, and compatibles. You can enter data and commands into the computer from either the braille keyboard or the PC's keyboard. With the joystick, you can maneuver the window, moving down and across lines and columns. You can also maneuver the window with the keyboard, either tracking the cursor as it moves around the screen or monitoring a 20-character window while entering data.

Review commands enable you to move the window to any screen location or jump to the beginning or end of a line, or to the top or bottom of the screen. A Find function is also provided. You can send control commands to the computer from the braille keyboard.

Other features include WordPerfect commands and macros that let you perform most functions from the braille keyboard, and a menu-manager program that assists you in managing your hard disk drive from a braille menu.

Price: $4195.

Contact: Telesensory Systems Inc., 455 North Bernardo Ave., Mountain View, CA 94043, (800) 227-8418 or (415) 960-0920; in California, (800) 874-9009.

Inquiry 769.

The Sensual Keyboard

Northgate takes behaviorist philosophy to the typist. According to company president Arthur B. Lazere, Northgate's C/T (which stands for click/tactile) keyboards give you a "positive entry sensation," which results in faster typing speeds and fewer errors. The keyboards come in 84- and 101-key versions and work with the IBM PC, XT, AT, and compatibles.

The 101-key keyboard includes 12 function keys arranged across the top in the standard 101-style format. Both keyboards feature lighted indicators for the Caps Lock and Num Lock keys, and enlarged Shift, Enter, Control, Alt, and Backspace keys.

continued

2 new monitors for the System/2.™

2 good to be true.

Some people shy away from technological change. But at Amdek®, we look upon change as an opportunity.

And now that there's a new generation of PC's, we have the opportunity to introduce you to 2 new monitors from Amdek– the 732 color and 432 monochrome.

The meticulous details.

From the beautifully styled cabinet to the flicker-free screen, these new monitors are unmistakably Amdek.

Text so sharp, you'll think your newspaper is blurry by comparison. Graphics so colorful, you'll have a tough time trying to think of a hue you can't incorporate.

And because the Amdek 732 and 432 are compatible with IBM®'s new Personal System/2™ Video Graphics Array (VGA) and MultiColor Graphics Array (MCGA), the image of all your programs will look better than you've ever imagined.

The 732 allows you to choose from a palette of over 256,000 colors–up to 256 colors at once. And the text switch delivers clear single color text for word processing.

The black and white of it.

For the ultimate in monochrome, the 432 features a large 14-inch flat surface screen that projects visually larger black type against a high-contrast white phosphor background. The impression is more like ink on paper.

Combine these features with our non-glare screen and tilt/swivel stand, and you'll see that Amdek has thought of everything.

Then compare our monitor price against other monitors compatible with the System/2.

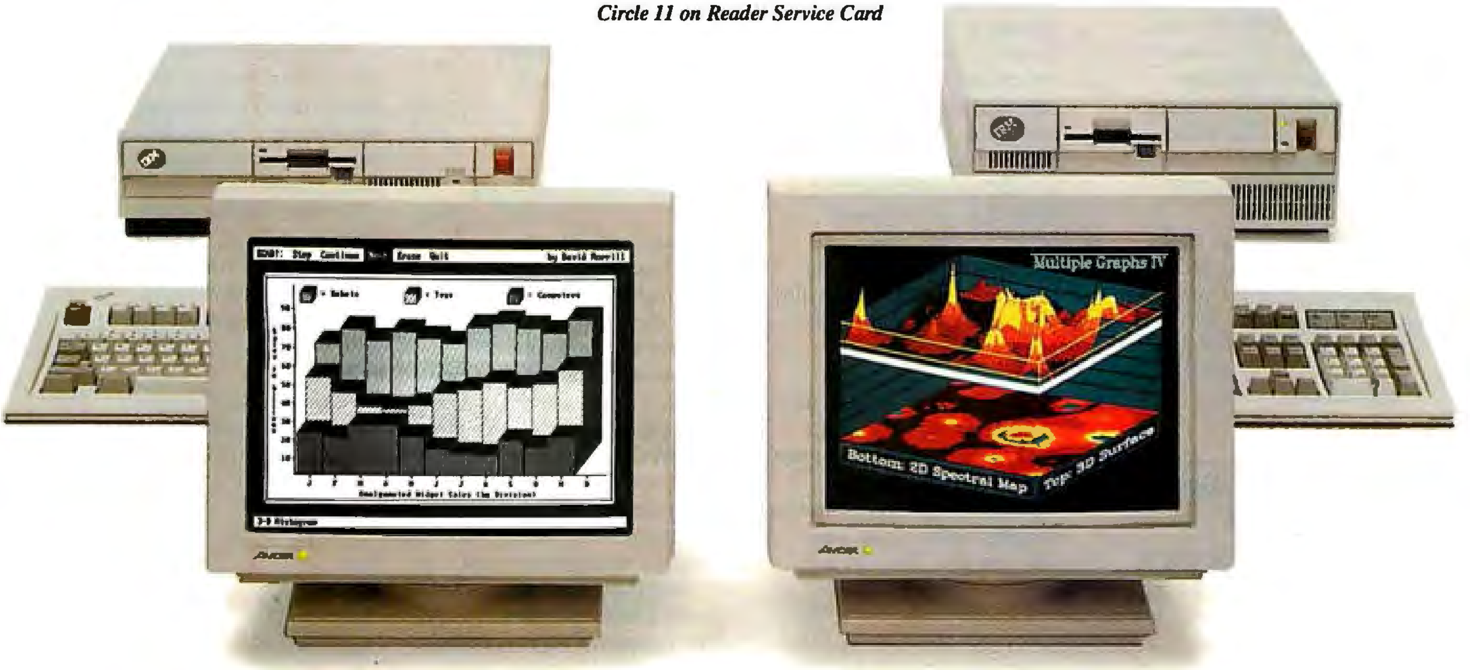
We think you'll enjoy that benefit, 2.

AMDEK

Clearly the finest in monitors.

1901 Zanker Road, San Jose, CA 95112 Phone: 800/PC-AMDEK (800/722-6335) FAX: 408/436-8187

Circle 11 on Reader Service Card



IBM is a registered trademark of International Business Machines Corp. Personal System/2 is a trademark of International Business Machines Corp.

Price: 84-key version, $79; 101-key version, $99.
Contact: Northgate Computer Systems, 2905 Northwest Blvd., Suite 250, Plymouth, MN 55441, (800) 328-8907 or (612) 553-0631.
Inquiry 770.

MagniView's Overhead Interface

MagniView 200 is a 12.6- by 12.5- by 1.1-inch monochrome LCD display with an 8.3- by 6.2-inch screen size and 640- by 240-bit resolution. The 3½-pound display fits onto a standard overhead projector, and the 4-3 aspect ratio of the screen (identical to that of a computer monitor) provides distortion-free image projection of text and graphics. A built-in fan, infrared filter, and heat-tolerant liquid crystals are designed for heat-tolerant operation. Top-mounted, on-board controls let you differentiate between like colors when translating them to a monochrome display.

MagniView 200 interfaces with the IBM PC, XT, AT, and compatibles with CGA graphics, and with Apple IIs and compatibles. The unit plugs into the RGB or composite video output. Included is Presentation Partner software for IBM PCs and compatibles, which lets you capture screen displays from applications software. Options are a "Y" video cable for displays to a second monitor, and a carrying case.
Price: $1195; video cable, $45; carrying case, $65.
Contact: Dukane Corp., 2900 Dukane Dr., St. Charles, IL 60174, (800) 634-2800; in Illinois, (312) 584-2300.
Inquiry 771.

MagniView 200 for overhead projectors uses a 4-3 aspect ratio.

Music Editing, Scoring, and Arranging

RolandCorp's M.E.S.A. (Music Editor, Scorer, and Arranger) has song, score, and print modes that let you record, edit, and print music.

You can enter music using a MIDI instrument, mouse, or keyboard. M.E.S.A.'s song mode includes a 65,000-note capacity, eight tracks, programmable tempo changes per beat, timing offsets of any beat by any number of clock pulses, and looping function by track or song. You can name or mark tracks; apply MIDI event filters during or after recording; and edit notes, beats, bars, phrases, and tracks. A graphics display assists you in cutting and pasting.

Score mode displays phrases in standard musical notation on your screen. You can create complete compositions and individual notes, and insert, delete, and modify MIDI events. You can also prepare scores for printing and insert text, phrase marks, triplet brackets, clefs, and other musical symbols. By using the mouse, you can also draw special symbols or markings on the screen.

In the print mode, you can view a page of music as it will appear when printed. With a Hercules monochrome or EGA display, you can view up to eight staves simultaneously; with a CGA, you can view five. You cut and paste individual measures and print to dot-matrix and laser printers and plotters.

M.E.S.A. runs on the IBM PC, XT, AT, and compatibles with MS-DOS 2.0 or higher; 512K bytes of RAM (640K bytes recommended); an EGA, CGA, or Hercules graphics cards; and a parallel printer port. A Microsoft or Mouse Systems mouse is optional. Also required are a Roland MPU-401 MIDI Processing Unit and Roland MIF-IPC interface card. The package supports all MIDI instruments, Roland reports.
Price: $695.
Contact: RolandCorp U.S., 7200 Dominion Circle, Los Angeles, CA 90040-3647, (213) 685-5141.
Inquiry 772.

Slideworks Graphics

Slideworks lets you create pie charts, bar and line graphs, text charts, pictures, and maps. The package includes a color palette of 64 hues and a library of 24 predefined images. Printer drivers support HP LaserJet II, IBM Color Jetprinter, and Epson dot-matrix printers.

Template guides, preformatted charts, and recommended colors are designed to speed chart formatting. Customizing tools include a draw feature for boxes, circles, ellipses, and lines; a choice of seven typefaces; support for an international character set; a choice of foreign languages; horizontal, vertical, and video frame templates; and editing features. The program supports most spreadsheet and word-processing programs.

Slideworks runs on the IBM PC, XT, AT, and compatibles with 512K bytes of RAM, MS-DOS or PC-DOS 2.1 or higher, and two floppy disk drives or one floppy disk drive and one hard disk drive. The program supports EGA and CGA graphics; it does not support Hercules graphics.
Price: $249.
Contact: Management Graphics Inc., 1450 Lodestar Rd., Unit 1, Downsview, Ontario M3J 3C1, Canada, (416) 638-8877.
Inquiry 773.

Thermistor Analyzer

ThermiCalc assists you in determining the resistance-temperature parameters of Fenwal Electronics' thermistors. You can develop complete curves in 1-degree increments for the product through its operating range, or develop specific resistance-temperature relations for a particular application.

You can select from several different types of thermistors. Basic formulas and descriptions of types of thermistor products are available in the program's glossary.

ThermiCalc runs on the IBM PC, XT, AT, and compatibles with IBM BASICA or equivalent, 256K bytes of RAM, and two floppy disk drives.
Price: $38.50.
Contact: Fenwal Electronics, 450 Fortune Blvd., Milford, MA 01757, (617) 478-5255.
Inquiry 774.

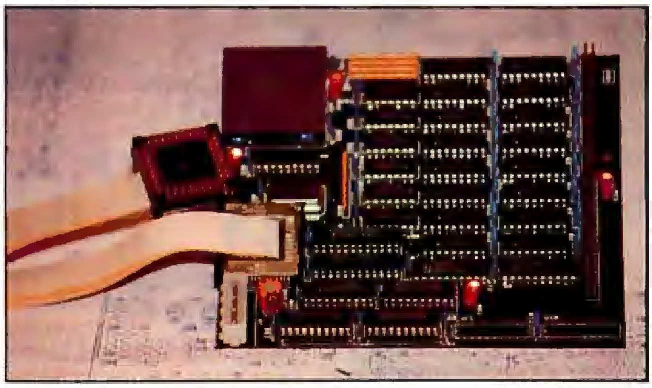
continued

No-slot 80386 Upgrade

The 386Eagle is an 80386-based add-in board designed to "sidesaddle" the floppy/hard disk controller of an IBM PC AT or compatible. It's also available for the IBM PS/2 Models 50 and 60. The board measures 4.4 by 5.5 inches and includes a 16-MHz 80386 processor, 512K bytes of RAM, and a 32-bit data bus.

On an AT or compatible, the board mounts onto the disk controller's G-35 connector and plugs into the socket of your system's 80286 chip via a conversion cable. According to Application Engineering, the 386Eagle can increase system performance by as much as four times.

An 80-pin connector lets you add up to 4 megabytes of expansion memory. An optional 80287 math coprocessor is also available.
Price: $1695.
Contact: Application Engineering and Associates Inc., 3420 East Shea Blvd., Suite 227, Phoenix, AZ 85028, (602) 996-7762.
Inquiry 775.



The 386Eagle upgrades your AT without using a slot.

Control Your VCR via Your PC

The VCR Controller Card from Innovative Tech Works lets you control up to two professional-style video tape recorders—those recorders equipped to use the SMPTE RS-422 serial communications protocol.

The card takes up a full slot in your IBM PC or compatible. You can control most front-panel functions of your VCR and develop your own control programs. A software-development driver works with most programming languages. You can perform single-frame edits and edit control sequences using the card.

The card is designed for use with most paint and animation programs and can control most devices that use the RS-422 protocol, including recently released special effects devices and character generators.
Price: Single-channel model, $795; dual-channel model, $995.
Contact: Innovative Tech Works, 405 Battleground Ave., Suite 200, Greensboro, NC 27401, (919) 370-0855.
Inquiry 776.

Prolog Compiler

The Cogent Prolog Compiler offers a full implementation of the standard Edinburgh Prolog language with over 150 predefined procedures. It provides support for real, string, and database reference types, and the development environment includes the Prolog Compiler and Interpreter, as well as a window-based debugger, help subsystem, and dynamic loader. The compiler also offers user-programmable windowing, screen control, and error trapping and handling.

Sample Prolog expert-system, language-processing, and decision-support programs are included. The run-time version requires an IBM PC or compatible with at least 256K bytes of RAM. To run the compiler, you need 384K bytes of RAM.
Price: $200.
Contact: Cogent Software Ltd., 21 William J. Heights, Framingham, MA 01701, (617) 875-6553.
Inquiry 777.

Menu At Work

Menu At Work is a DOS shell that lets you access DOS commands and hard disk directories and subdirectories via menus.

Menu At Work can organize a hard disk file menu automatically, or you can create custom menus. The program provides password protection and presents DOS commands in plain-English menus. Menu At Work also lets you create reports on system usage.

Menu At Work runs on the IBM PC, XT, AT, and compatibles with MS-DOS or PC-DOS 2.1 or higher, 256K bytes of RAM, and a hard disk drive.
Price: $195.
Contact: Management Science Associates Inc., 6565 Penn Ave. at Fifth, Pittsburgh, PA 15206-4490, (412) 362-2000.
Inquiry 778.

3-D Graphics with Lotus 1-2-3

Intex Solutions' 3-D Graphics is a Lotus 1-2-3 add-in that lets you create three-dimensional bars, joined bars, financial bars, lines, and surface charts using Lotus 1-2-3 data.

3-D Graphics enables you to select axes automatically or manually, add titles and axis labels, and produce either color or black-and-white graphics. You can represent 100-by-100 data arrays for surface plots, and 10-by-100 arrays for line and bar charts.

The program also lets you rotate the graph to view it at different angles, color contour, and view hidden lines on surface plots. You can save your graphics to disk as a picture file, then print or edit them with PrintGraph, Freelance, or Manuscript.

3-D Graphics runs on IBM PCs and compatibles with PC-DOS or MS-DOS 2.0 or higher; 256K bytes of RAM; a CGA, EGA, or Hercules board; and Lotus 1-2-3 version 2.0 or higher.
Price: $79.95.
Contact: Intex Solutions Inc., 568 Washington St., Wellesley, MA 02181, (617) 431-1063.
Inquiry 779.

Zoom's $199 2400-bps Modem

Zoom/Modem HC 2400 is a 2400-bit-per-second, half-card modem for the IBM PC, AT, XT, and compatibles. The modem supports the Hayes AT command set as well as Bell 103A, Bell 212A, and CCITT v.22 bis protocols. Features include auto-answering, auto-dialing, dial-tone detection, an on-board speaker, a second jack for a telephone set, adaptive equalization, on-board power-up, and analog loop-back diagnostics.

Zoom Telephonics says that the unit's high-speed 16450 UART is designed to ensure compatibility with the contemporary crop of 80286- and 80386-based systems that operate at 8 MHz and above.

The Zoom/Modem HC 2400 is FCC-registered and has a two-year warranty. It's shipped with ProComm communications software, which includes terminal emulation; XMODEM, YMODEM, and Kermit communications protocols; script files; and a host mode.

continued

Magnavox Multimode Display means VGA compatibility...

*Actual un-retouched photograph

. . . in addition to auto-frequency selection for CGA, EGA, PGA, and AT&T graphics. Plus we offer you a top-notch color image, full input selection, and line level audio for interactive applications.

For additional information on the Magnavox Multimode or our full line of computer display products, call 1-800-223-4432, in TN call 615-521-1601.

MAGNAVOX
Smart. Very smart.

Circle 125 on Reader Service Card

Multimode Display suggested retail $899.

Price: $199.
Contact: Zoom Telephonics Inc., 207 South St., Boston, MA 02111, (617) 423-1072.
Inquiry 780.

Complete 386-Based System

Hertz Computer Corporation has introduced the Hertz AT386 Plus Model 80, a fully loaded 80386-based system. Included in the 16-MHz system is 2.5 megabytes of high-speed 32-bit RAM, an 80-megabyte hard disk drive with an average access time of 28 ms, a 60-megabyte tape-backup unit, a 2400-bps internal modem with communications software, a combination MDA/CGA/EGA card, a color monitor, and MS-DOS 3.2.

The system comes with your choice of any combination of two of the following: a 5¼-inch 360K-byte disk drive, a 1.2-megabyte 5¼-inch disk drive, and a 720K-byte 3½-inch disk drive. The system has single parallel and serial ports, a 101-key keyboard, a 195-watt power supply, and eight full-length expansion slots, including two 32-bit, four 16-bit, and two 8-bit. There's also a socket for an optional 80287 or 80387 math coprocessor.
Price: $6995.
Contact: Hertz Computer Corp., 325 Fifth Ave., New York, NY 10016, (212) 684-4141.
Inquiry 781.

Reorganize Your Hard Disk

DS Optimize lets you reorganize your hard disk so that physical files reside in one location. According to Design Software, the program can optimize a badly fragmented 10-megabyte hard disk in 8 minutes.

DS Optimize lets you reorganize your entire hard disk, selected directories, or files within a directory. You can also place the two directories you read from and write to most often in certain areas on the disk to reduce access time.

The program features diagrams that display your hierarchical directory tree structure and which disk clusters are used by a directory or file. It also features copy, erase, and locate functions, and it enables you to move files between subdirectories and create and remove subdirectories.

DS Optimize runs on IBM PCs and compatibles with MS-DOS or PC-DOS 2.0 or higher and 256K bytes of RAM.
Price: $69.95.
Contact: Design Software Inc., 1275 West Roosevelt Rd., West Chicago, IL 60185, (312) 231-4540.
Inquiry 782.

Financial Reporting

Javelin Plus combines spreadsheet, database-management, and graphics capabilities to let you perform financial analysis. The program consists of a central information base surrounded by 10 ways of entering, manipulating, and reporting data. It is designed for working with both numeric and nonnumeric information that must be included as data in an analytic model.

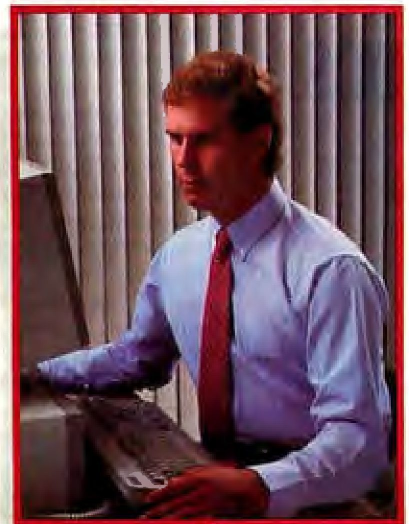
Javelin Plus lets you enter text or dates anywhere you can enter a number. You can use dates in formulas, calculate days between dates, compare dates, and find the start and end dates of variables. You can select or sort records in a list as part of your model and summarize database information over time.

The program also includes snap-in building blocks. These enable you to perform what-if analyses, design and implement data-entry forms, and perform multiple-regression and histogram analysis.

Other new features include a range restriction that lets macro-driven models

continued

Print Master lets people share printers



Get the most out of your expensive resources—your people, your computers and your printers—with Print Master by BayTech. Not only does it let your people and computers share all of your printers, its buffer keeps them working instead of waiting.

Easy to set up, easy to use

Initial setup is menu-prompted and allows you to configure Print Master to your application. You decide which ports are input and which are output. You also set such functions as baud rates, handshaking, timeout, how you select a printer, etc.

After setup, you simply cable Print Master between your computers and printers, and you're ready to go. BayTech's Memory Resident Program lets your PC users select printers via function keys that correspond to a menu window.

Print Master keeps everything running

All users can send data simultaneously to Print Master's dynamically allocated buffer, keeping your people and computers working. This first-in-first-out buffer also sends data to all printers simultaneously to keep them running at full capacity. The buffer size? A big 512K that's expandable to one megabyte.

Eleven models with serial and parallel

An advantage of Print Master is its flexibility: Any port can be a computer port or a printer port. Choose from eleven models with different combinations of serial and parallel ports. Print Master internally converts serial to parallel and vice versa on combination serial/parallel models. Six ports, $795. Eight ports, $895. Ten ports, $995.

Non-buffered models from $339. GSA pricing available.

Want details?

Contact your dealer or Bay Technical Associates for information about Print Master and BayTech's complete line of data communications products, proudly made in the U.S.A.

BayTech

Bay Technical Associates, Inc.
Data Communications Products Division
200 N. Second St., P.O. Box 387
Bay Saint Louis, Mississippi 39520
Telex 910-333-1618 BAYTECH
Phone 601-467-8231 or
800-523-2702

Circle 31 on Reader Service Card

specify a data-entry area on a worksheet and prevent unauthorized modification of other parts of the worksheet, and a `/File Run` command that lets you incorporate DOS commands or other programs into applications processing. You can automatically import all variables from another model, and read DIF, SYLK, WKS, WK1, WRK, and comma-delimited files. Graph types added include high/low/close, mixed line and bar charts, and XY plots with up to seven variables. Javelin Plus also supports the Apple LaserWriter and other PostScript printers.

Javelin Plus runs on IBM PCs and compatibles with MS-DOS or PC-DOS 2.0 or higher, 512K bytes of RAM, and two disk drives. The program comes on 5¼-inch floppy disks, but 3½-inch disks are available for $30 extra. It is not copy-protected.
Price: $249.
Contact: Javelin Software Corp., One Kendall Square, Bldg. 200, Cambridge, MA 02139, (617) 494-1400.
Inquiry 783.

Turn Your PC Into a Logic Analyzer

You can use Heathkit's IC-1001 Logic Analyzer to study circuits with sequential or combinational logic. It connects to your IBM PC or compatible through an RS-232C serial port, and you can use it in circuits operating at up to 10 MHz.

The IC-1001 is a 16-bit analyzer that provides 16 data lines for checking a 16-bit-wide data bus or 16 separate logic test points. A clock input and two clock qualifier inputs are also provided. It's compatible with both TTL and 5-volt CMOS logic.

The accompanying software lets you display state and timing, including hexadecimal/octal and ASCII equivalents. You can also perform checksum operations with bit selection.

You can configure the IC-1001 to capture a specific sequence of pulses and use a single or repeating trigger with selectable time delay to capture a window of pulses. A delay mode enables you to acquire data up to 50,000 pulses after trigger; a non-delay mode lets you view events 2000 pulses before trigger.

The IC-1001 can communicate between 300 and 19,200 bps. Oscilloscope trigger outputs are provided. The product measures 1.75 by 9.25 by 8.5 inches.
Price: $269.
Contact: Heath Co., Dept. 150-935, P.O. Box 1288, Benton Harbor, MI 49022, (616) 982-3200.
Inquiry 784.

PS/2-Compatible Light Pen

FTG Data Systems' PXL-350/4 is a light pen for the IBM PS/2 Model 30. The package includes a controller board, software, and an adapter cable.

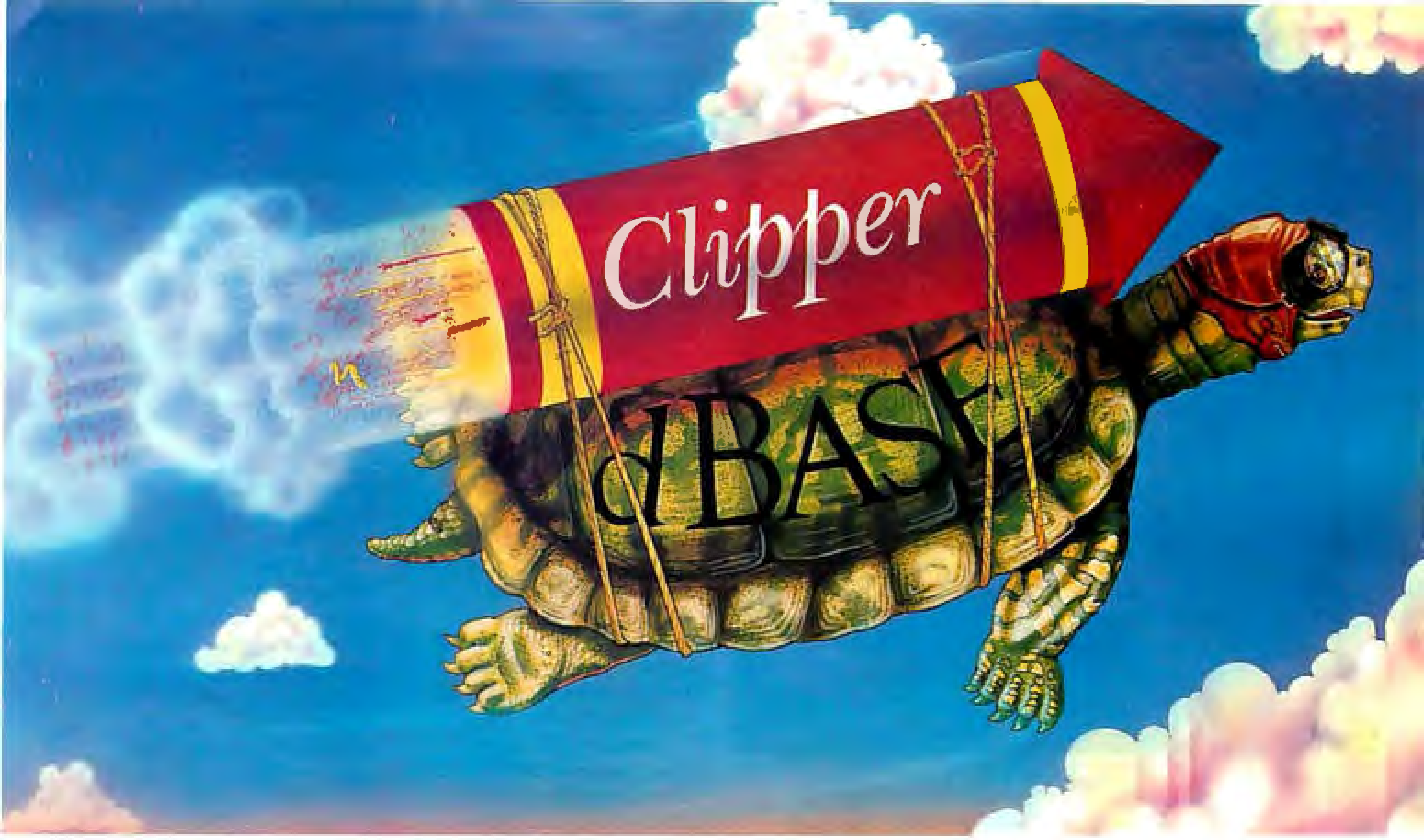
The company's light pen was originally developed for use with CGA and EGA adapters to deliver pixel-level resolution and hardware interrupts for Microsoft Windows. The PXL-350/4 is also compatible with DCA's E78Plus and other 3270 emulation products.
Price: $189.
Contact: FTG Data Systems, 10801 Dale St., Suite J-2, Stanton, CA 90680, (800) 962-3900; in California, (714) 995-3900.
Inquiry 785.

Interactive Waveform Analysis

Interactive Waveform Analysis is a program designed to assist physical- and life-science teachers and researchers in studying the properties of periodic functions and conducting experiments with waveforms.

You begin by entering be-

continued



Turtle Souped

Real programmers don't use dBASE. Or do they?

We're finding that some very swift programmers are using it to write some very fast applications, and are completing their projects much more quickly.

But they cheat.

They use our Clipper™ compiler to combine dBASE™ with C and assembler.

With dBASE used like pseudo-code, they can then quickly create prototypes that actually run.

Then, with dBASE doing the high-level database functions, they use our Clipper compiler to link in C or assembly language modules from their own bag of tricks.

And they're finding that they're linking in less than they expected because Clipper compiled code runs so fast and because of Clipper's built-in enhancements.

Clipper includes easy networking that provides file and record locking the way it should be done.

Fast screens that can be treated as memory variables and eliminate the need for direct screen writes and all that tortuous heap management code.

Box commands that make windowing a breeze. And more.

So if you'd like to use your time more productively, check us out: Nantucket Corporation, 12555 W. Jefferson Boulevard, Los Angeles, CA 90066.

Or if you're on deadline, call (213) 390-7923 today.

Clipper could get you out of the soup.

nantucket®

 In Europe: Nantucket Corporation (Europe) 2 Bluecoats Avenue, Fore Street, Hertford, Herts SG14 1PB Telephone 0992 554621.

Circle 146 on Reader Service Card (Dealers: 147)

tween 50 and 500 data items. The program then performs a Fourier analysis and produces a trigonometric series consisting of a constant term plus N/2 harmonic terms. You can view the analysis graphically in both time and frequency domains, in addition to tabular form. You can save results to disk and perform screen dumps to your printer.

After the initial analysis, you can compute derivatives of the original and all derived functions and display their waveforms and spectra.

The program also includes a module that lets you perform cross-correlations between series pairs with various lags and leads.

Interactive Waveform Analysis runs on IBM PCs and compatibles with PC-DOS or MS-DOS 2.0 or higher, 256K bytes of RAM, two disk drives, and a graphics display. It is available on both 3½- and 5¼-inch floppy disks.
Price: $59.95.
Contact: Mathematical Software Co., P.O. Box 12349, El Cajon, CA 92022, (619) 940-0343.
Inquiry 786.

W.O.R.K. At Home

Britannica Software's W.O.R.K. At Home is an integrated program that includes word-processing, database, spreadsheet, and report modules.

The word-processing module lets you create documents up to 10,000 characters long with 128K bytes of up to 24,000 with 256K bytes or more. The database module supports up to 20 fields, with a maximum field length of 33 characters. You can have up to 360 records per disk. The spreadsheet measures 26 columns by 99 rows.

W.O.R.K. At Home runs on IBM PCs and compatibles with MS-DOS 2.0 or higher and 128K bytes of RAM.
Price: $59.95.
Contact: Britannica Software, 185 Berry St., San Francisco, CA 94107, (415) 546-1866.
Inquiry 787.

24-pin Combination Printer/Plotter

JDL's 850 GL+ and 850 EWS 24-pin printer/plotters can print in 14 colors and support automatic paper feeding for A- through C-size paper and vellum in engineering and architectural formats. They print lines of 16-inch letter-quality text.

The JDL-850 GL+ is compatible with A- through E-size HP plotters. A 1-megabyte plot spooler and five standard fonts are provided. The GL+ prints 360 characters per second in draft mode and 144 cps in letter-quality mode. It's compatible with the Diablo 630, IBM 5182 Color Graphics, and Epson printers. Options include font cards, additional printer emulations, and a dual-bin sheet feeder.

The JDL-850 EWS is the same as the GL+ but is not compatible with the HP Graphics Language.

The EWS comes with either a Centronics parallel or RS-232C serial interface; the GL+ has both. They measure 25.5 by 16.8 by 7.5 inches.
Price: JDL-850 GL+, $3845; JDL-850 EWS, $2495.
Contact: JDL Inc., 2801 Townsgate Rd., Suite 104, Westlake Village, CA 91361, (805) 495-3451.
Inquiry 788.

SEND US YOUR NEW PRODUCT RELEASE
If you want us to consider your product for publication, send us full information about it, including its price, ship date, and an address and telephone number where readers can get further information. Send to New Products Editor, BYTE, One Phoenix Mill Lane, Peterborough, NH 03458. Information contained in these items is based on manufacturers' written statements and/or telephone interviews with BYTE reporters. BYTE does not represent itself has having formally reviewed each product mentioned.

Try our Turbo 101.™
Your computer will never be the same.

Why? Because there's a good chance that the keyboard that came with your computer has about as much feeling as a sponge. Not so with the DataDesk (made in the U.S.) Turbo 101 Enhanced keyboard. You get a keyboard that *feels* good to the touch. And its exclusive tactile-feedback keys let you type faster—with fewer mistakes than ever before—regardless of what PC/XT®/AT® or compatible you are using.

And that's not all. The "Selectric" typewriter layout of the Turbo 101 is ideal for word processing. The separate cursor and numeric pads end the need to toggle num-lock, dramatically simplifying all spreadsheet and data-entry operations. The Turbo 101 incorporates all the features of the new IBM Keyboard Standard plus many improvements carefully designed to improve your hands-on computing performance.

Better yet, it's never been easier to increase your productivity. There's nothing complicated to learn—just plug in the Turbo 101 and start typing! You'll get a new feeling about your computer. (You'll notice the difference from the very first keystroke).

Don't wait to get your hands on the DataDesk Turbo 101 Enhanced keyboard. Right now, you'll receive your choice of Borland's Turbo Lightning® or SuperKey® software absolutely free (a $99 value!) with each Turbo 101. And to be sure you're completely satisfied, you get a 30-day, money-back guarantee and a two year warranty. There's no better time to see how good you and your PC can feel with our Turbo 101.

Here's how the critics feel about the Turbo 101:

"I have absolutely no hesitation in recommending the DataDesk Turbo 101."
Jerry Pournelle
Byte Magazine

"DataDesk International has designed a sturdy and handsome keyboard that has a tactile response . . . it is the hardware bargain of the year."
Charles Humble
Oregonian

"For ingenuity of design and sheer dollar-value, the Turbo 101 is very hard to beat."
Curt Suplee

"I recently had a chance to use Word 3.0 and DataDesk's 101 Key keyboard. The result was superb."
Bill Machrone
PC Magazine

*with purchase of Turbo 101 enhanced keyboard

All DataDesk International products are registered trademarks or trademarks of DataDesk International, Inc. Other brand and product names are trademarks or registered trademarks of their respective holders. Copyright 1987 DataDesk International.

MADE IN USA

To order by phone call:
(800) 826-5398
In CA (800) 592-9602

Act now and get Borland's Turbo Lightning or SuperKey FREE with your Turbo 101 Keyboard. All for only:

$149.95*

Please send me ____ Turbo 101 Enhanced Keyboards at $149.95 each plus $10 each shipping and handling. California residents must also include $9.75 each sales tax. Include with each keyboard:

____ Turbo Lightning ____ SuperKey

Name ____________

Company ____________

Street Address ____________

City ____________ State ____ Zip ____

Phone ____________ Computer Type ____________

Method of paym ‹ Visa MasterCard

Credit Card No: ____________ Exp. ____

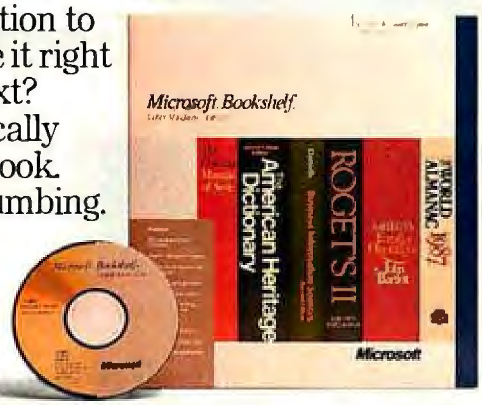
AT&T & PCjr compatible version available for $174.95; Macintosh compatible version with 101 KEYS™ for $169.95

datadesk
INTERNATIONAL
760 Haskell Avenue Van Nuys, CA 91406

Circle 71 on Reader Service Card (Dealers: 72)

Just hit ALT-Shift. Bookshelf's at your command. Use the keyboard or mouse to get a definition. Find a quote. Check a fact. Edit a phrase. Spell relief. Zip to a ZIP Code. More.

In just seconds, Bookshelf searches the source, calls the information to the screen. You can paste it right into your document. Next?



Say goodbye to physically jumping from book to book. There's no lifting. No thumbing. Agonizing. Stopping. Starting. No derailing your train of thought. Nothing comes between you and your writing.

It rooms with your PC.

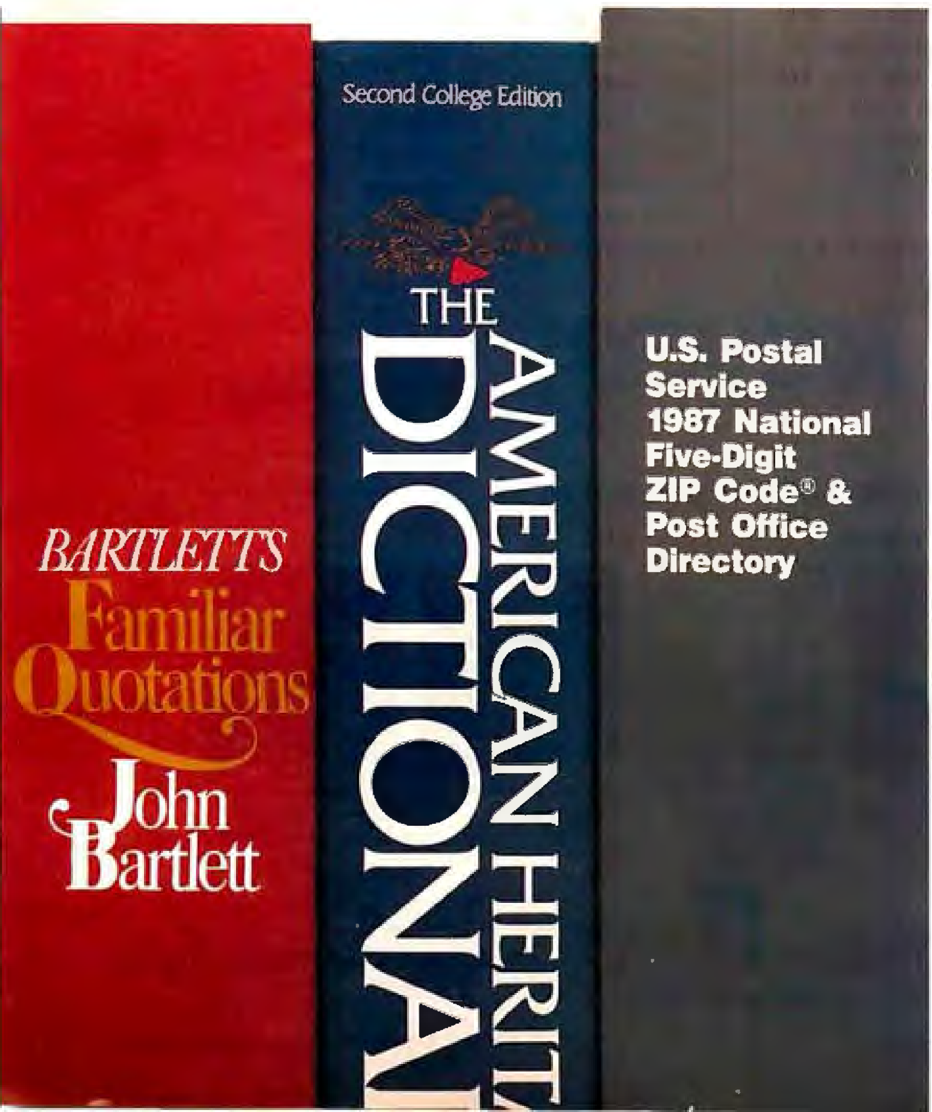
Bookshelf is memory-resident. It works from within your word processor. So it's always there. A keystroke away.

You'll need a CD ROM drive, so Bookshelf is available with the Amdek LASERDRIVE. Or you can use it with your own drive.

If this sounds like the best of all possible worlds, call (800) 426-9400 for more information or the name of the dealer nearest you.

Microsoft® Bookshelf™

deriving machine.



Reference sources used in Microsoft Bookshelf contain text-only material—no graphics that cannot be reproduced with IBM® extended character set.

In Washington State and Alaska, call (206) 882-8088. In Canada, call (416) 673-7638.

THE ADVANTAGE ENHANCED

Kaypro Corporation — electronics innovator since 1952 — has made a good thing even better. The KAYPRO 286i Model C now features a 40-MB hard drive and the 101-key AT-style keyboard. With the latest standard feature enhancements, the KAYPRO 286i is the smartest choice in advanced computer technology.

Advanced.
The heart of the KAYPRO 286i is the 80286 microprocessor — with a processing rate of 10 MHz and a 640-kilobyte RAM. The perfect match for today's high productivity software.

And Enhanced.
The KAYPRO 286i Model C has a 1.2-MB floppy disk drive, plus a hard disk with 40 MB of storage. The KAYPRO 286i AT-style keyboard features the new 101-key layout with separate cursor control, numeric keypad, and 12 programmable function keys.

Perhaps the nicest surprise about the KAYPRO 286i/C is the suggested retail price of $2995.

You won't find distinctive metal construction, 10-MHz processing, and free name-brand software that includes WordStar Professional Release 4 in any other AT-type computer. Other company's extras are Kaypro standard features.

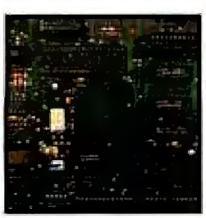
KAYPRO®
COMPUTERS
The Future's Built In

KAYPRO Lease-Link
Kaypro's Commercial Leasing

See us at COMDEX Las Vegas Booth No. 1260

KAYPRO CARD
Kaypro's Revolving Charge Plan

The KAYPRO 286i Model C features...


80286, 10-MHz Microprocessor.


An internal hard disk drive with 40 MB of storage.


Enhanced 101-key IBM PC/AT-style keyboard with security keylock.


A 640-KB RAM, expandable to 15 MB.


Bundled software includes WordStar Professional Release 4.

Trademarks: 286i, Kaypro Corporation; IBM, AT International Business Machines; WordStar Professional Release 4, MicroPro International.

For the Kaypro Dealer near you, call 1-800-4KAYPRO.

made in U.S.A.

Circle 112 on Reader Service Card (Dealers: 113)

The importance of IBM's latest microcomputer offering

The Technical Implications of the PS/2

G. Michael Vose

Is there any real technical substance to IBM's new Personal System/2 lineup of machines? Or is IBM simply peddling yesterday's technology augmented by new graphics and an as-yet-unseen, complex, untested new operating system?

These questions form the context for the analyses on the following pages. I've assembled the thoughts of staff editors here at BYTE and combined them with a sampling of opinions offered by people from the microcomputer industry whose ideas we at BYTE respect.

The introduction of the PS/2 generated thousands of column-inches of type given over to endless discussions of what the PS/2 means to clone makers, what it portends for software vendors, and what may happen to a soon-to-be-orphaned generation of machines. Our intent here, however, is to focus on the technology of the PS/2 computers to ascertain whether this technology will make significant changes in the future of personal computing on an IBM platform. These collected ideas may help shed some light on what implications the PS/2 generation holds in store.

The Big Picture

Overall, the observations that follow paint a very positive picture of the direction staked out by IBM's newest product line. The two most significant technologies include the Micro Channel bus and the OS/2 operating system. Closer looks at both of these technologies, as well as the PS/2's new VGA graphics, are offered elsewhere in this issue. [Editor's note: *See "The 32-bit Micro Channel" by Jon Shiell, "TSRs Past and Future: MS-DOS and OS/2" by Ray Duncan, and "PS/2 Video Programming" by Richard Wilton.*]

Several general observations can be made about the PS/2 family's technology. First, Intel's 80x86 architecture is well-understood, even though it's disliked by many people. This understanding makes possible chip sets and application-specific integrated circuits (ASICs) for building machines more cheaply, as well as reasonable development tools for writing software. Therefore, it is safe to say that an Intel microprocessor-based line of computers is fairly mature.

Maturity may not be considered a technical advantage in science, where creativity in problem-solving is more important. But in the venue of the average microcomputer user, maturity means compatibility and reliability.

In fact, some people think that software has replaced hardware as the standard target for building new computers. John Roach, chief executive officer of Tandy Corp., recently noted, "Most people don't care much about hardware anymore. They just want to know if their favorite software will run on a machine." If this attitude is widely shared, then the hardware of the PS/2 machines will satisfy many people—at least until the OS/2 finally arrives.

Compatibility implies standards, and as a standard, the PS/2 machines' technology seems solid. But whether IBM allows it to be adopted by other manufacturers concerns BYTE'S editor in chief, Philip Lemmons, who says, "With the introduction of the PS/2 machines, IBM has begun to compete in the personal com-

continued

G. Michael Vose is a BYTE senior technical editor. He can be reached at BYTE, One Phoenix Mill Lane, Peterborough, NH 03458.

Illustration: Robert Wisnewski © 1987

puter arena on the basis of technology. This development is welcome because the previous limitations of the de facto IBM standard were painfully obvious, especially in systems software. The new PS/2 'standard' offers numerous improvements: The Micro Channel is a better bus than the PC and AT buses, and it provides a full standard for 32-bit buses. The VGA graphics standard improves on the EGA. The IBM monitors for the PS/2 series take a new approach that will ultimately deliver superior performance at lower prices. IBM is using 3½-inch floppy disks that offer more convenience, capacity, and reliability than 5¼-inch floppy disks. And OS/2, the new system software jointly developed by Microsoft and IBM, will offer advances such as true multitasking and a graphic user interface.

"Yet a cloud hangs over all this outstanding new technology. Like other companies that have invested in the development of new technology, IBM is asserting proprietary rights in its work. When most companies do this in most product areas, we expect and accept it. When one company has the special role of setting the de facto standard, however, the aggressive assertion of proprietary rights prevents the widespread adoption of the new standard and delays the broad distribution of new technology.

"The personal computer industry has waited for years for IBM to advance the standard, and now, depending on IBM's moves, may be unable to participate in the advancement. If so, the rest of the industry and the broad population of computer users still need another standard for which to build and buy products—a standard at least as good as the one embodied in the PS/2 series."

Whether or not the PS/2 machines set a standard, BYTE senior technical editor Gregg Williams finds noteworthiness in the technology of the machines for a variety of reasons:

"The PS/2 design is noteworthy more because it was done by IBM than because of its inherent worth. The design is good, but not great; it opens large areas of future growth (and present incompatibility); and it determines what a large portion of the microcomputer industry, both companies and individuals, will and will not do.

"IBM's use of the Intel 80286 processor in the PS/2 Models 50 and 60 will have a strong fragmenting effect on the IBM PC market (in which I'll include the PC and PS/2 computers and all clones). Imagine what life would be like if IBM had introduced only 80386 machines: An 80386-based operating system, with its protected 8086 mode, would have allowed multiple *existing* MS-DOS applications to run in the same machine. This means that most of your existing IBM PC programs would do multitasking on the new machine, and software developers wouldn't have to agonize over whether to write MS-DOS- or OS/2-compatible programs.

"IBM's exclusive use of 3½-inch floppy disks puts the final nail in the coffin of 5¼-inch floppy disks, a process that was begun by Hewlett-Packard, Apple, Commodore-Amiga, and Atari. On the other hand though, IBM's insistence that the mouse pointing device be optional may hurt the company. If less than 75 percent of the installed user base buys mice, developers will think twice before writing programs that use them heavily—and that will limit the range and power of applications that can (and will) be written. With a huge number of PS/2 machines being sold before the graphic interface is available, the fate of the mouse as part of the standard PS/2 configuration is very uncertain.

"I'm not sure how many people will pay $395 for OS/2 (IBM may bring down

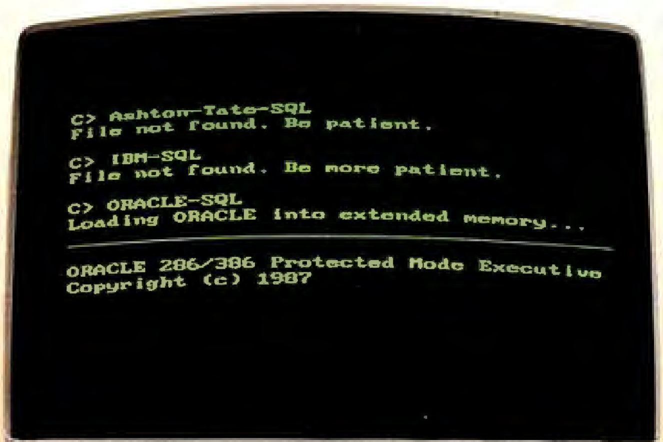
continued

 Circle 165 on Reader Service Card

TRY ORACLE'S $1295 SQL DBMS FOR ONLY $199 TODAY. OR ORDER VAPORWARE.

If you're looking forward to OS/2 and the next generation of PC database management systems to enable you to build larger, higher-quality PC applications, you'll be interested to know that:

- Ashton-Tate has announced its intention to replace its outdated database technology with a SQL DBMS under OS/2, and
- IBM has announced its intention to offer SQL for OS/2 in its Extended Edition.



All you have to do is wait. Wait for OS/2 and SQL. Wait until mid-1988...or later! Meantime you keep developing with soon-to-be obsolete dBASE™ technology.

But you don't have to wait for SQL on the PC.

Oracle has it now. And you don't have to wait for OS/2 to run programs larger than 640K. The ORACLE® DBMS allows you to run OS/2-size programs under today's MS-DOS.™

ORACLE is the number-one selling DBMS on both minicomputers and mainframes. If you try PC ORACLE, we think you'll make us number-one on PCs, too. That's why we're making you this special offer.

We've sold thousands of copies of PC ORACLE for $1295. But now, for a limited time, we will send you the same, full-function version of ORACLE for only $199.*

That's right. Only $199 for a PC SQL DBMS that is identical to the ORACLE that runs on minicomputers and mainframes. $199 for the *only* PC DBMS that lets you write larger-than-640K, OS/2-size programs and run them under today's MS-DOS.

ORACLE makes it easy to write large, high-quality applications using industry-standard SQL. Applications that run faster. Applications that are easier to use and have more capability. Multi-user, networked applications. Applications that can be developed on MS-DOS today, and then run unchanged on OS/2, minicomputers and mainframes.

So stop writing dead-end dBASE code today. And stop writing cramped, limited-to-less-than-640K applications. Call 1-800-345-DBMS today. Or use the rightmost coupon. Or mail one of the other coupons. And wait.

ORACLE®

COMPATIBILITY • PORTABILITY • CONNECTABILITY

Call 1-800-345-DBMS today.

Dear Ashton-Tate,

20101 Hamilton Avenue
Torrance, CA 90502

Since you've announced your intentions to go to SQL, I really don't feel like developing obsolete applications in dBASE. I have every confidence your first attempt will be full-featured and bug-free. Send me what you've got, when you get it.

Name
Title
Company
Street
City
State Zip

Dear IBM,

Old Orchard Road
Armonk, NY 10504

Attached is a blank check. I keep reading that you've already announced that you're going to announce SQL for OS/2 and the Personal System/2.

When you do, fill in the check amount. Hope to hear from you sometime in the next couple years.

Name
Title
Company
Street
City
State Zip

Dear Oracle,

PC ORACLE • Oracle Corporation
20 Davis Drive • Belmont, CA 94002

I want to run OS/2 ORACLE on MS-DOS today. I agree to use this license only for application development. I understand all prices include applicable taxes and postage.

Send me *(check only one)*:

- ☐ PC ORACLE $199
- ☐ ORACLE + 1MB Extended memory $599
- ☐ ORACLE + 2MB Extended memory $799

I'm paying by ☐ Check ☐ VISA ☐ MC ☐ AMEX

Print Name Date
Company Title
Street
City State Zip
Credit Card Number
Card Expiration Date
Signature BYTE

One Oracle Parkway • Belmont, CA 94002 • World Headquarters (415) 598-8000 • Calgary (403) 265-2622 • Ottawa (613) 238-2381 • Quebec (514) 337-0755 • Toronto (416) 598-7750 • ORACLE Systems Australia 61-2-959-5080 • ORACLE Europe 44-848-6911 • ORACLE Systems Hong Kong 852-5-266846

*PC ORACLE requires an "OS/2 ready" 286/386-based PC running DOS 3.1+ with 1MB extended memory (2MB is recommended) and a minimum 10MB hard disk drive.

the price) or how many vendors will design boards that use the 32-bit version of the Micro Channel bus. Another critical factor is IBM's Systems Application Architecture (SAA), which promises to provide a standard user interface for *all* applications. Noting that Apple's success with the Macintosh stems largely from its extremely consistent user interface, I cannot emphasize the role of SAA too much. If it's a lackluster design, it will either cripple the software that conforms to it, or designers will ignore it and go their independent ways—and I don't know which of these is worse.

"A computer design always grows, in both software and hardware, well beyond what is immediately planned for it, and both the PS/2 hardware and the OS/2 operating system have a lot of elbowroom for future growth. Once the PS/2 becomes the machine on most IBM users' desks, things will be quite nice. It's just the transition that's going to be messy, as usual."

A messy transition also worries Ed Tolson, president of SoftLogic Solutions (Manchester, New Hampshire), who states, "OS/2 is the missing link between promise and reality. Even once OS/2 is in general use, it will take an additional one to two years before applications developers fully exploit the potential of the PS/2 machines.

"In the meantime, the abrupt and radical shift in system architecture may create major headaches for large user's groups with quantities of both PCs and PS/2 machines. Support and maintenance, in particular, will be significantly complicated by the difficulties of data-swapping and software compatibility created by the 3½-inch disks and by the lack of compatibility of virtually every piece of hardware in the system, necessitating different spare parts for each family.

"Finally, one of the more subtle implications of the PS/2 line relates to the large-scale integration of the components on the motherboard. While this is ostensibly aimed at providing systems that are more reliable, the real motivation behind it is lower manufacturing costs. The upshot for users is that board-swaps for these highly integrated systems promise to be very high in price after the warranty period expires.

"It is hard to be completely comfortable with this new standard when confronted by the tremendous divergence of paths created by new hardware that lacks a clear commitment to the 80386. Although there is an 80386 member of the PS/2 family, it currently just mimics an 80286 machine, albeit twice as fast. The 80286 seems to be the legitimate heart of the PS/2 line. Does this mean there will be another quantum leap in technology in just another couple of years?"

The Micro Channel Bus

The most intriguing technology in the PS/2 family is the high-speed system bus, called the Micro Channel. This bus opens up many new approaches to distributing computing tasks, since it permits a variety of bus masters equal access to a common data path.

Jim Truchard, president of National Instruments (Austin, Texas), a maker of IEEE-488 bus devices, thinks the Micro Channel holds much promise. He says, "The Micro Channel architecture replaces the I/O adapter system found in previous IBM PCs. The new architecture supports the functionality of its predecessor but is not compatible with it, since it attempts to eliminate most of the PC's design deficiencies.

"The Micro Channel architecture supports three connector systems: a 16-bit system with 24-bit addressing, a 16-bit system with a video extension, and a 32-bit system with matched memory cycles

continued

The Ten Recommendments

MODEL 9000
—Fastest programmer on the market.
—Quick & Intelligent programming algorithms.
—Supports Megabit Eproms, including MPUs.
—Programs largest variety of chips incl. Prom replacements, Eproms and EEProms.

MODEL 7228
—Programs all popular chips to 512K.
—Intelligent programming algorithms.
—High performance to cost ratio.
—Free communications software.

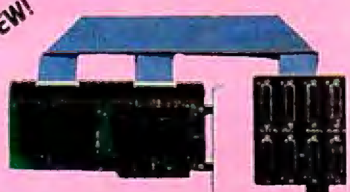
MODEL 7344 PAL DEVELOPMENT SYSTEM
—Includes programmer, GPC Compiler and communications s.w.
—Program, secure and fuctionality test 20 & 24 pin pals.
—Supports MMI, NAT, TI, Cypress Erasable CMOS PALS.
—A must for any new cost effective design.

MODEL 7128
—Compatible with virtually any computer.
—High performance to cost ratio.
—Programs all popular chips, incl. MPUs.
—Free communications software.

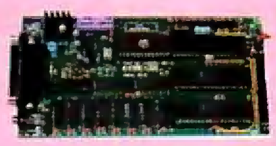
MODEL 7956 PLUS
—Programs 8 eproms at a time.
—Programs all chips, to one megabit, incl. single chip processors.
—Operates serial to 56Kbps or stand alone.
—Ultra fast communications software.

MODEL 705
—Stand alone operation.
—Programs Motorola 68705 Family MPUs.
—One button operation from cycle power to verify.
—Great for production.

NEW!



PCSS-8 SERIAL BOARD
—Eight RS232 ports per card (optional 4 port)
—All eight ports, 100% DOS compatible.
—32 ports may be added to a PC.
—Optional RS422 to 4000 ft.
—Interrupt driven BIOS Enhancement Software included free.



MODEL 2010 SINGLE BOARD COMPUTER
—2010B: Intel 8032 Basic v1.1 Solid State EEProm Disk System.
—2010M: Machine Language with built-in assembler/disassembler.
—64KByte RAM, 40 BIT programmable I/O lines.
—RS232 & expansion interface.

NEW!

MODEL ROMX-2 EPROM EMULATOR
—Emulates 2716-27256 eproms.
—Battery backed up, auto emulate on power-up.
—Low-cost, pays for itself on first project.
—Free 19.2K Serial Communications Software.
—150 NS access with faster optional.

CPEmulator
—Runs CPM programs on IBM PC/AT/PS2®/386s
—No hardware required. (V20 &V30 supported)
—Completely portable.
—Supports complete Z80 Instruction Set.
—CPEmulator & utilities only $99.

All GTEK products are manufactured by GTEK in the U.S.A., are fully warranted for 1 year and offer free technical support. These products do not occupy card slots (except PCSS-8) and work with IBM PS/2® and 386 machines. So, order toll free today at 1-800-255-4835.

GTEK®, INC

Developmental Hardware & Software
Drawer 1346; 399 Highway 90
Bay St. Louis, MS 39520
(601)467-8048
Telex: 315814 (GTEK UD)

IBM PS/2 is a registered trademark of International Business Machines, Inc.

and 32-bit addressing. Eight high-performance direct-memory-access channels provide DMA transfer rates from 4 to 8 megabytes per second. These rates are substantially higher than those of the PC AT.

"Nimble support of multiple bus masters makes possible the support of high-performance multiprocessor systems. The Programmable Option Select (POS) feature eliminates the headaches associated with conventional hardware jumpers and switches. The features of the Micro Channel combine to provide the simplicity and flexibility needed to carry the PS/2 machines into the next generation of applications, including networking and multiprocessing.

"On the negative side, the reduced form factor imposed by the Micro Channel (the cards are 40 percent smaller than AT cards) will tend to limit the variety of I/O options to high-volume applications that can support custom chips and surface-mount technology (SMT) or simple applications that require only small amounts of board real estate. The bandwidth of communication between devices plugged into the backplane is limited because the main processor shares its bandwidth with backplane devices. The option between boards with 16- and 32-bit data paths will encourage vendors to support only the 16-bit I/O function, thereby *reducing* performance on 80386 machines."

The Micro Channel's form factor has Trevor Marshall, vice president of engineering at Definicon Systems (Chatsworth, California), worried as well. He explains, "The biggest surprise I got from the PS/2 announcements was the Micro Channel architecture. I had certainly expected a proprietary bus with 32-bit capability, but I was astonished to find that the new expansion card was so much *smaller* than the old AT profile card (only 59 percent of the surface area).

"The effects of that change are really quite profound. Unless new technologies, such as ASICs or surface mounting, are used, it is not possible to continue to supply complex systems (such as 32-bit coprocessors) for the Micro Channel. Since these technologies require high-production volumes to be economical, small independent (and innovative) add-in houses will be unable to effectively compete in the Micro Channel marketplace.

"Even if a product's development can be commercially justified, a limitation has now been placed on the complexity and performance of the add-in system. For instance, looking at our latest coprocessor product, the IC package surface area of the CPU, the floating-point interface, and the Weitek floating-point chips alone take nearly 40 percent of the available Micro Channel board space. Our AT profile board is fully packed with 48 square inches of support circuits, including only 1 megabyte of high-speed RAM.

"The use of SMT or ASICs is necessary to increase this available RAM to a useful figure, even with the AT add-in technology. Porting such a product to the Micro Channel would require a size compression of these support circuits from 48 square inches to 22 square inches, an impossible task using SMT alone. Thus, IBM has mandated that we develop ASICs to meet the new form factor. This means fewer product variations and significantly increased lead time to market new CPU technologies.

"In addition, there is *no way* that such a system could be implemented on a Micro Channel board without considerable manufacturing investment, which would raise the cost to the end user.

"It is often said that the features of the Micro Channel are its high speed and its multiple bus-master modes. Why not just use the Micro Channel itself for the ex-

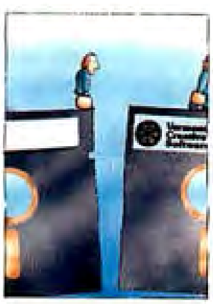
continued

WINDOWS FOR DATA®

Uncommon Screens

If you program in C, take a few moments to learn how Windows for Data can help you build a state-of-the-art user interface.

- ☐ Create and manage menus, data-entry forms, context-sensitive help, and text displays — all within windows.
- ☐ Provide a common user interface for programs that must run on different machines and operating systems.
- ☐ Build a better front end for any DBMS that has a C-language interface (most popular ones do).



FROM END TO BEGINNING

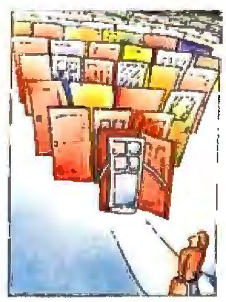
Windows for Data begins where other screen packages end, with special features like nested pop-up forms and menus, field entry from lists of choices, scrollable regions for the entry of variable numbers of line items, and an exclusive built-in debugging system.

NO WALLS

If you've been frustrated by the limitations of other screen utilities, don't be discouraged. You won't run into walls with Windows for Data. Our customers repeatedly tell us how they've used our system in ways we never imagined — but which we anticipated by designing Windows for Data for unprecedented adapatability. You will be amazed at what you can do with Windows for Data.

YOU ARE ALWAYS IN CHARGE

Control functions that you write and attach to fields and/or keys can read, compare, validate, and change the data values in all fields of the form. Upon entry or exit from any field, control functions can call up subsidiary forms and menus, change the active field, exit or abort the form, perform almost any task you can imagine.



OUR WINDOWS WILL OPEN DOORS

Our windows will open doors to new markets for your software. High-performance, source-code-compatible versions of Windows for Data are available for **PCDOS (OS/2** soon), **XENIX, UNIX,** and **VMS.** PCDOS versions are fully compatible with Microsoft Windows, TopView, and DESQview. **No royalties.**

You owe it to yourself and your programs to try Windows for Data. If not satisfied, return for a full refund. Call for **FREE DEMO.**

Vermont Creative Software

21 Elm Ave., Richford, VT 05476
Telex: 510-601-4160 VCSOFT FAX 802-848-3502
Tel.: 802-848-7731
Prices: PCDOS $395; XENIX, VMS, UNIX Call.

Circle 237 on Reader Service Card

pansion RAM? Well, it is already three times too slow for today's CPU technology. Cycle times on today's advanced microprocessors, such as Sun's SPARC, are typically 60 nanoseconds. As early as the first quarter of 1988, cycle times will have fallen to 40 ns. When the emitter-coupled logic (ECL) bipolar version of the SPARC is available in 1989, then 10 ns will be the system speed requirement.

"The Micro Channel has an absolute minimum cycle time of 200 ns. Clearly, the Micro Channel alone is not the bus for the future. Innovative add-in technology will be required if the PC is to keep pace with the expanding world of the super-microcomputer workstation."

Another view of the Micro Channel reveals some other interesting quirks. Jon Shiell, systems architect at CGAA (Sunnyvale, California), who writes about the new bus in his article entitled "The 32-bit Micro Channel" on page 59 in this issue, sees it this way:

"The Micro Channel is a big step in the right direction, even though it's not complete; the current 32-bit Micro Channel slots don't appear to support matched memory cycles for anything other than the system microprocessor. In addition, the current versions of the Model 80 don't support 32-bit addresses for DMA, or, for that matter, 32-bit data. There is nothing that precludes future machines from doing so.

"POS is another good move; the fewer switches, the better. However, I suspect that as time goes on, either the POS setup program that IBM provides will have to get smarter or some people will end up with unconfigurable PS/2 machines. (Note that I am ignoring the idea that someone else could provide an equivalent program.)

"My major problem with the Models 50 and 60 is that they should have been zero-wait-state machines, and the Model 50's hard disk drive should have been a faster, 32-megabyte unit. I would also like to see a version of the Model 80-111 running with a cache at 20 megahertz with zero wait states and 32-bit DMA and MMC (matched memory cycle) for other bus masters. Such a system would be great for adding a second processor as a bus master with its own cache.

"However, there is no question in my mind that the most important thing about the new machines is that they are not closed. Whether they can be cloned is another question from the legal sense, but I see nothing in the technical sense that would prevent it."

Fitting the Pieces Together

The Micro Channel is an obviously important technical innovation, but how do the pieces of the PS/2 puzzle fit together? Ray Duncan, president of Laboratory Microsystems (Marina del Rey, California), suggests that the new machines change the nature of the whole ball game. He notes, "The new IBM PS/2 Models 50, 60, and 80 are slick machines in their own right, but I think that their real significance lies in their role as portents of the future.

"First, the PS/2 machines dramatically raise the baseline level of computing power that people can reasonably expect of any desktop computer. Whereas today's clone maker can still foist off 8088-based machines with floppy disk drives on the buying public, the low-end clone builder of two years from now will have to provide, at minimum, an 80286- or 80386-based machine with 2 megabytes of RAM, a hard disk drive, and a high-resolution graphics system with an analog monitor—or be shut out of the marketplace. And I would hate to try to predict what kind of machine the high-end clone manufacturers, like Compaq, will be delivering in two years.

"Next, the long-range significance of the new Micro Channel bus can only be dimly imagined at this point. As designers learn to exploit its wider data path and ability to support multiple bus masters, we should see the emergence of coprocessor, disk-controller, and graphics expansion cards that will triple or quadruple the power of a bare PS/2 machine. We may also see network adapters with an on-board processor and RAM that will allow a PS/2 machine to be used as a server with no degradation whatsoever in its performance for the local user.

"Finally, the PS/2 line represents a significant redirection of some of IBM's most prized resources. These machines remind us what formidable high-tech talent IBM can bring to bear—when it chooses to—in the areas of styling, mechanical design, large-scale integration, and efficiency of manufacturing. By comparison, the original PC line of computers are just clunky-looking boxes built from off-the-shelf components from the corner electronics store.

"Where does the PS/2 Model 30 fit in? In my opinion, nowhere. Its inclusion of the old bus, an 8086 processor that can't run protected-mode operating systems, and an idiosyncratic video controller, mark it as an interim machine with no growth path. This is a machine that will, or at least ought to, quietly fade away in a couple of years like its philosophical predecessors, the IBM Portable PC, PCjr, PC AT/370, and PC XT/286. If you like the Model 30, maybe you'd also be interested in this neat Osborne computer sys-

continued

tem that I've got in the closet. . ."

The long-range potential of the PS/2 line also impresses Carrell Killibrew, graphics processor manager at Texas Instruments (Houston, Texas), who says, "The IBM PS/2 architecture is the foundation of the most powerful personal computing systems of the next 10 years. PS/2 systems will have adapters for performing almost any computer-capable task, and they will be able to access hundreds of megabytes of disk storage and display results on monitors with speeds and resolutions surpassing today's best engineering workstations. These capabilities are not available today, but are the promise of the PS/2 machines' more sophisticated and flexible bus structure, the Micro Channel architecture (MCA); faster disk subsystems; and standard display interface.

"The MCA is a work of art. There will be more variation in the types and numbers of adapters available for the PS/2 family, since the MCA extends I/O device addressing to the full 64K bytes supported by the Intel 80x86 family (the PC and PC AT only supported 1K-byte addresses). The POS registers required to be in each adapter will allow the system to determine what each adapter is, what it does, and how to communicate with it. A nice benefit of this feature is that users will no longer worry about setting adapter-card switches to avoid address conflicts. Support for multiple system bus masters is much better thought out than it was for the PC AT bus that preceded it.

"Gone are the days of plugging strange and delicate cables into the original motherboard when upgrading to the next-generation Intel processor—your 80986 option adapter (available perhaps in the year 2001) will have no trouble taking over PS/2 system resources. And if you are the type who wants to have multiple (and possibly different) processors in your system, the MCA supports arbitration between 16 different bus masters.

"The 1-to-1 sector interleave available in PS/2 hard disk subsystems will make data-intensive applications and database programs run noticeably faster than they do on the more primitive PC AT versions. This situation will be improved even further when the IBMCACHE.SYS driver is used to create fast disk-caching.

"Note that disk-caching is inherently superior to RAM disks, since disk writes cause an actual write to the hard disk media. Need more hard disk capacity? The PS/2 machines should be capable of supporting many physical (not logical) disk subsystems with the MCA's eight DMA channels (of course, what I would do with eight 115-megabyte disk systems,

continued

"There's No Reason To Look Beyond PC TOOLS"

–PC Magazine, Editor's Choice Award, Two-Time Winner

Data Recovery · ***Fast*** *Hard Disk Backup* · *Disk Management*

Now we've given you ev
reason–PC Tools™ Deluxe.

PC Tools Deluxe gives you all the best features of **Fastback**,™ **Norton**,™ **XTREE**,™ **Mace**,™ **Sidekick**,® **Lightning**™ and **Disk Optimizer**–a $700 value–for just $79.

Other utilities may claim they "do it all," but only PC Tools Deluxe delivers:

• A DOS shell so complete 80 Micro called it their "overwhelming choice based on ve
satility, ease of use, and cost." • The best UNDELETE available–instead of merely "guessing," it recovers all data even on fragmented files. • HARD DISK BACKUP faster and every bit as reliable as Fastback. • The leading UNFORMAT for hard disks and floppy disks, enabling you to recover from almost any disk disaster. • PCFORMAT for 100% safe formatting of floppy and hard disks. • A quick, reliable DISK CACHING utility for speeding up disk access. • A *fast* COMPRESS feature that like Disk Optimizer dramatically improves hard disk performance. • And a better resident mini WORD PROCESSOR than Sidekick, including word wrap, search/replace, formatted page printing and more.

Before you buy any of these other utilities, take a look at PC Tools Deluxe. Like PC Magazine, you'll find there's no reason to look any further. **For the dealer nearest you, or to order direct, call (503) 244-5782,** M–F, 8 a.m.–5 p.m. (West Coast time).

Central Point Software
INCORPORATED
9700 S.W. Capitol Hwy.
Portland, OR 97219
(503) 244-5782

Circle 46 on Reader Service Card

I can't imagine). The one drawback to all this disk performance and capacity is the probability that it is the bare minimum necessary to support disk-hungry operating systems, such as Unix and (I suspect) OS/2.

"The PS/2 line will never suffer the display schizophrenia of the PC/XT/AT family, thanks to IBM's placing the VGA on the motherboard and providing an auxiliary video extension (AVE) as part of the MCA. The VGA, regardless of the display monitor used, will allow developers to write to a single display device for the majority of applications.

"For those who need or want to take advantage of the performance and capabilities offered by display adapters with high-performance drawing processors, the AVE supports redriving VGA-sourced display information through the added display adapter. This clever architectural innovation should result in an increasing flood of display adapters available to the PS/2 market, since providing backward compatibility with the VGA is unnecessary. IBM's own 8514/A display adapter uses this feature of the MCA to supply VGA compatibility."

OS/2: The Missing Link

Another important technology driving the PS/2 systems is the multitasking operating system, OS/2. Few people have experience with OS/2, but the people who have seen it are very excited by what they've observed. But all the experts queried for this article agreed that it was much too early to speculate on the potential of OS/2.

Skeptics of OS/2 were dismissed by Bill Gates of Microsoft in a speech to the Silicon Valley User Society recently as "the same kind of people who were reluctant to switch from CP/M to MS-DOS."

The Implications

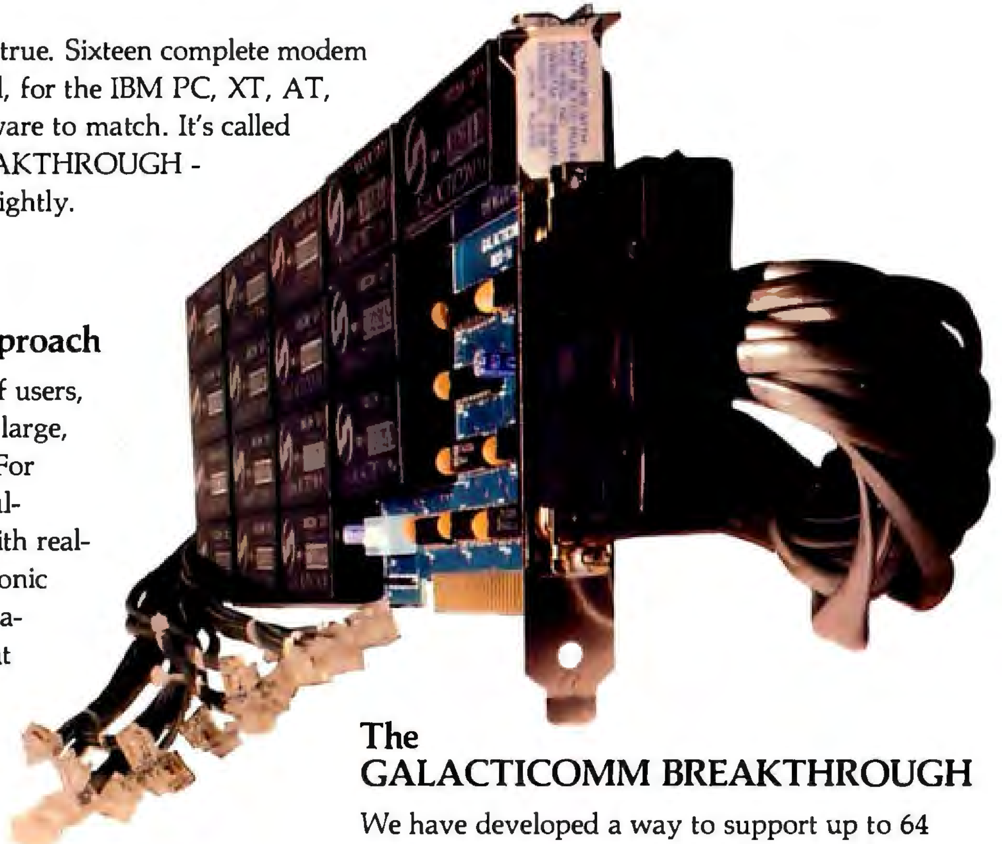
From these reactions, we can conclude that the PS/2 machines inject just enough new technology into the PC milieu to enliven our lot for a year or two. But the Micro Channel and 80286-based machines by themselves do not usher in a new generation of computing power. And OS/2's potential is still long on promise and short on reality.

The real technological breakthroughs peek over the horizon, however. The 80386/80486-based machines running OS/3 (or whatever the next operating system will be called) and built around an even higher performance bus will be the machines we await most eagerly.

In the meantime, we'll be busy porting and adapting software for the VGA and OS/2. That will keep us occupied until the next wave unfurls. ■

16 Modems on one card??

It sounds incredible, but it's true. Sixteen complete modem modules on one plug-in card, for the IBM PC, XT, AT, and compatibles. With software to match. It's called the GALACTICOMM BREAKTHROUGH - and we don't use that term lightly. Read on to find out why.



The Conventional Approach

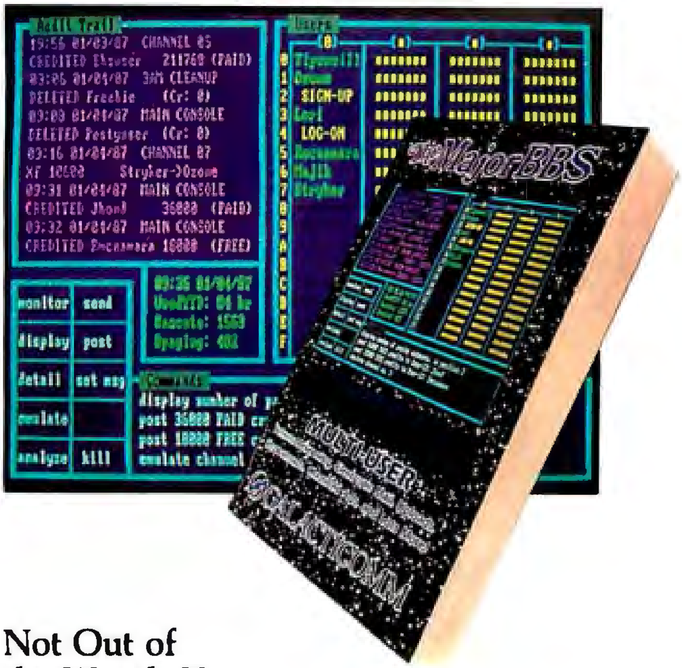
To handle a large number of users, you are supposed to have a large, expensive computer, right? For example, to support 64 simultaneous users in a system with real-time teleconferencing, electronic mail, upload/download capability and so on, you need at least a minicomputer. You are supposed to spend about $50,000 to $100,000, right? And you need special, high-capacity disk drives, maybe 9-track tape — not to mention plenum flooring, climate control, and a battalion of technicians on call 24 hours.

The GALACTICOMM BREAKTHROUGH

We have developed a way to support up to 64 simultaneous users on a PC/AT, with real-time teleconferencing, electronic mail, upload/download capability, and so on. Average response times are better, in many cases, than equivalently loaded mainframes. The system bottleneck tends to be the hard disk more than anything else — which shows the performance of the communications subsystem. There are actually two distinct breakthroughs here, one in software and one in hardware.

The SOFTWARE BREAKTHROUGH

If you want to develop your own high-performance multi-user dial-up systems, this is for you. We supply a library of about 40 subroutines, written in assembly language but designed to interface to C, that takes care of all of the low-level, nitty-gritty component of your application: all of the hardware interrupt servicing, the "ring buffer" management for I/O data, echo, command and status queues, etc. etc. This not only saves you programming time, but more importantly, these routines are just unbelievably efficient. They form the core of the multi-user capability we provide, and, using them, you can afford to be more relaxed about the efficiency of your application-specific programming than you might think.



Large or Small, It Works

Slip four Model 16 cards into your AT, and you can support up to 64 data channels, or users, at one time. Even on an XT-class machine, you can generally handle 24 users or so without any problem. And if you're just getting off the ground, we also offer the Model 4, a half-slot card with only 4 modems on it.

Support? 24-hour telephone hotline support for one full year is included in the base price of every card. You also automatically get free software library updates for one full year after buying a card. And our limited warranty on materials and workmanship also extends for one full year.

Want off-the-shelf software? Start with The Major BBS, our demo multi-user Bulletin Board System, and adapt it to your exact requirements. For a free sample of what it's like, call (305) 922-3901 with your modem (8-N-1). If you want a copy, the C source and a 175-page book are included with it — and we're practically giving it away, to get you hooked. Give us a call.

Not Out of the Woods Yet

But hooking up software like this to conventional modems is awkward and costly. You have to use multi-port serial cards. You have to run great wads of serial cable out the back of your PC to stacks, or racks, of external modems, which in turn have their own power supply cables, telephone cables, and so forth. The result is a wiring nightmare — and all those separate cables and connectors between you and the telephone company don't do wonders for reliability, either.

The Model 16 Modem Card

The Model 16 solves all that. One card holds 16 complete 300/1200 baud modems, including serial port interfaces, an individual microprocessor in each modem, all of the line filtering and auto-equalization logic needed, and 16 individual FCC-registered DAAs (Direct Access Arrangements, required when connecting to the public telephone network). That's pretty unbelievable too, but please read Steve Ciarcia's November 1985 Circuit Cellar column if you find yourself waxing skeptical. Steve is of course BYTE magazine's foremost columnist, and he seems to have a very high opinion of the devices we are using.

...or only 4?

(IBM PC, XT, and AT are trademarks of International Business Machines Corp.)

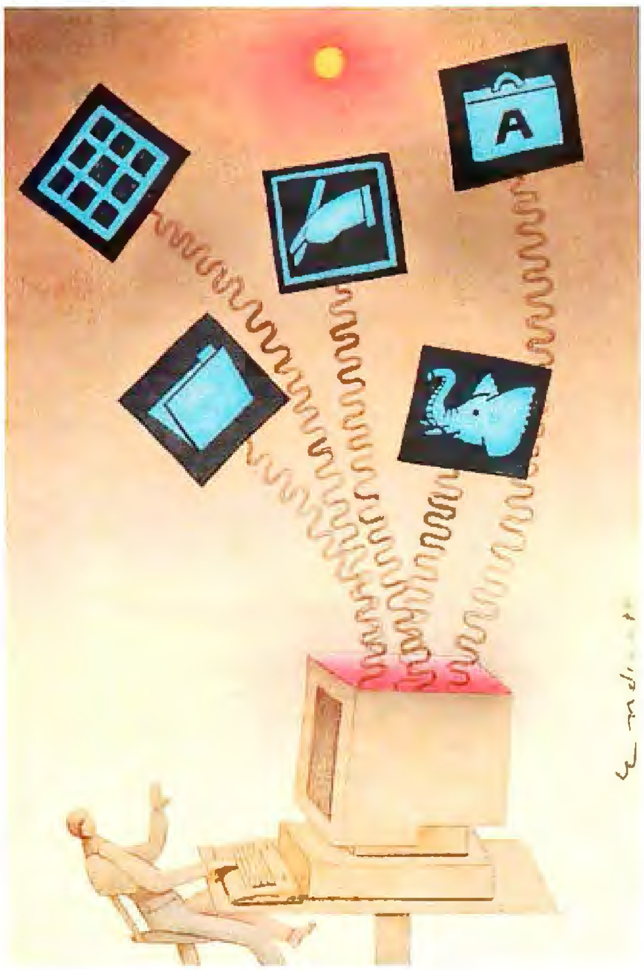
GALACTICOMM

GALACTICOMM, Inc., 11360 Tara Drive, Plantation, FL 33325 (305) 472-9560

How OS/2 facilitates writing well-behaved TSRs

TSRs Past and Future: MS-DOS and OS/2

Ray Duncan



Terminate-and-stay-resident utility programs (TSRs) are a unique phenomenon of the MS-DOS marketplace. You load a TSR into memory like any other application program, but its life thereafter is sharply different. Instead of performing its processing when you invoke it, it reserves the memory it occupies with a special MS-DOS function call and then lurks in the background until you need it, while returning overt control to the system's command processor, COMMAND.COM.

TSRs can be broadly categorized as either active or passive. Borland's SideKick is one of the earliest and most popular TSR products, and it exemplifies an active TSR. It sends out tendrils here and there in the system so it can monitor key presses while you are running other programs. When SideKick recognizes its special activation key sequence, or *hot key*, it grabs control of the screen and keyboard, displays a pop-up window, and awaits instructions. When you dismiss SideKick by entering the appropriate command, SideKick restores the screen to its previous state, returns control to the foreground application, and retreats into the shadows.

A passive TSR usually performs its work silently, augmenting the capabilities of the operating system in some way. A passive TSR does not typically interact with you after you load it.

Examples of passive TSRs are MS-DOS's APPEND command, which is the counterpart of PATH for data files, and the new FASTOPEN command, which saves information about previously accessed files so your programs can call and reopen data files more quickly.

TSRs under MS-DOS

As soon as MS-DOS programmers understood the generality and convenience of the TSR concept, their fertile imaginations bore fruit in incredibly varied resident utilities: keyboard macro expanders and enhancers, pop-up calculators, phone dialers, calendars, notepad editors, spelling checkers, synonym finders, idea outliners, debugging aids, program profilers, online manuals, file squeezers and unsqueezers, screen dimmers, and many more.

As TSRs proliferated, problems began to emerge, especially as users found it increasingly desirable to load many different TSRs at the same time.

One class of these problems stems from MS-DOS TSRs' disposition to take over hardware interrupts and perform direct operations on device controllers and the video refresh buffer to obtain maximum performance and detect hot keys. The presence of one TSR in the system rarely causes difficulties, but when multiple TSRs battle over the same timer-tick interrupt or keyboard-scan code, their interaction might cause unpredictable system behavior or complete crashes. Inevitably, a whole new class of TSR now exists solely to activate, deactivate, unload on command, and arbitrate conflicts between other TSRs. Various proposals for peaceful TSR coexistence have been put forth, such as the Borland API, the Lotus Metro, and the public domain Ringmaster specification, but none seems to be achieving widespread acceptance yet.

continued

Ray Duncan is a software developer and author of Advanced OS/2, *which is scheduled to be published by Microsoft Press in January 1988. He can be reached c/o Laboratory Microsystems Inc., P.O. Box 10430, Marina del Rey, CA 90295.*

Listing 1: *A demonstration of the use of the* VIOPOPUP *and* VIOENDPOPUP *calls.*

```
                                    ; Flag controls action if
                                    ; window not available.
PopFlags    dw      1               ; 1 = wait , 0 = no wait

msg         db      'This is a pop-up window'
msg_len     equ     $-msg

            .
            .
            .
                                    ; put up pop-up window...
            push    ds              ; address of pop-up flags
            push    offset DGROUP:PopFlags
            push    0               ; VIO handle (reserved)
            call    VIOPOPUP        ; transfer to OS/2.
            or      ax,ax           ; did function succeed?
            jnz     error           ; jump if function failed.

                                    ; now display message...
            push    ds              ; address of message
            push    offset DGROUP:msg
            push    msg_len         ; length of message
            push    12              ; y-coordinate
            push    (80-msg_len)/2  ; x-coordinate (center it)
            push    0               ; VIO handle (reserved)
            call    VIOWRTCHARSTR   ; transfer to OS/2.
            or      ax,ax           ; did write succeed?
            jnz     error           ; jump if write failed.

            push    0               ; pause for 3 seconds.
            push    3000            ; (user must be quick reader!)
            call    DOSSLEEP        ; transfer to OS/2.

                                    ; take down pop-up window...
            push    0               ; VIO handle (reserved)
            call    VIOENDPOPUP     ; transfer to OS/2.
            or      ax,ax           ; did call succeed?
            jnz     error           ; jump if call failed.
            .
            .
            .
```

Another category of TSR-related problems derives from MS-DOS's character as a single-user, single-tasking operating system. Since MS-DOS was not designed to support the execution of multiple concurrent processes, most of its functions are not reentrant. This poses severe problems for a TSR that needs to perform file operations but that might be activated as a result of a hardware interrupt that occurs while MS-DOS kernel code is executing. To be fair, the MS-DOS functions that make a TSR program possible (INT 27h or INT 21h function 31h) were originally intended for installation of device drivers or interrupt handlers after the system is booted, not for interactive resident utilities. After a couple of years of experimentation and disassembly of critical paths within MS-DOS, developers have gradually evolved strategies for safe TSR operation that involve the use of undocumented but fairly stable MS-DOS function calls and internal flags. (One of the best sources of this folklore is the ms.dos/secrets conference on BIX.)

TSRs have been victims of their own success. As TSRs have grown in power, they have also grown chubby, and the user who configures his or her system with half a dozen handy resident utilities might find that the amount of RAM remaining is barely adequate to run a word processor or spreadsheet. We jokingly refer to this situation as "RAM cram." The options for the TSR developer are limited. The 640K-byte ceiling imposed by IBM on the RAM allotted for MS-DOS and its programs is solid enough that it's not worth fighting over. Alternative solutions that involve the use of expanded (Lotus/Intel/Microsoft) or extended (above the 1-megabyte boundary) memory for storage of TSR code or data are worthless to PC owners without the proper hardware.

TSRs under OS/2

Now that OS/2, Microsoft's multitasking, single-user operating system, has finally arrived, what is the outlook for TSRs? After all, traditional multitasking (the concurrent execution of multiple programs doing different jobs, usually on separate data sets) is quite a bit different from the TSR model of multiple utilities brought to bear on the same data, with one foreground program controlling the input and output files. If OS/2 were to lack the necessary facilities to support TSRs, software vendors would find it much more difficult to work around the operating system than they did under MS-DOS, since OS/2 runs the Intel 80286 in protected mode.

There's good news and there's bad news. The good news is that the OS/2 designers have provided excellent downward compatibility with MS-DOS, even emulating many of its undocumented functions and internal tables and flags so that many currently available TSRs will run unchanged in real mode within OS/2's DOS 3.X compatibility box. The bad news is that none of the existing TSRs will run in protected mode at all; each one will have to be completely rewritten to conform to the new OS/2 Application Program Interface (API) and the various restrictions imposed by protected-mode operation. But more good news is that the positive aspects of 80286 protected mode, coupled with particular OS/2 function calls aimed specifically at the needs of TSR programmers, should make the creation of powerful, well-behaved TSR utilities easier than ever before.

Many of the problems that beset TSR writers and users under MS-DOS become irrelevant in the OS/2 environment. The memory protection and system of privilege levels that OS/2 provides in protected mode should eliminate unpleasant interactions between TSRs and other programs. The protected-mode virtual address space of 1 gigabyte will vanquish memory limitations for the foreseeable future. Also, since OS/2 supports multitasking and is fully reentrant, a TSR can call on operating-system services whenever it needs to, without regard for the state of other programs that might be running. In fact, the term *TSR* becomes a misnomer: A pop-up utility does not need to terminate and then play tricks later to regain control (but I will continue to use the term throughout this article).

Key OS/2 Features for TSR Writers

Three key OS/2 features, or families of functions, are of particular interest to TSR programmers: threads, pop-up displays, and device monitors.

Like other multitasking operating systems, OS/2 supports the concept of a process, which represents the execution of an application program and the ownership of any resources (files, memory, and so on) associated with that execution. Processes can spawn other processes and can exchange data with other processes via a variety of interprocess communication mechanisms. Unlike other multitasking systems I am familiar with, OS/2 also has the notion of threads, which represent points of execution within a process and are the dispatchable entity known to the system's scheduler.

Each process has a primary thread that receives control from OS/2 when the process is started; it begins executing at the program's designated entry point. However, that primary thread can start up additional threads within the same process. Multiple threads within a process execute asynchronously to one another; can have different priorities; can manipulate one another's priorities; and can share access to the same memory, files, pipes, queues, and semaphores.

TSRs are a natural application of the threads concept. For example, a screen-dump utility could contain three asynchronous threads that would communicate by semaphores: a thread to intercept critical errors (such as "printer off-line"), a high-priority thread to filter the keyboard character stream and watch for an activation hot key, and a low-priority thread that would copy the screen buffer into a disk file on demand.

Under MS-DOS, each TSR has to contain logic to save and restore the previous contents of the video display when it is activated. Handling this properly for all possible text and graphics display modes is quite a chore, especially since some video cards have huge refresh buffers (enhanced graphics adapters) or lack readable registers to determine the current mode (Hercules graphics cards).

OS/2, on the other hand, provides TSRs with an adapter-independent way to communicate with the user. A TSR can call the function VIOPOPUP to assert control over the display. OS/2 saves the current screen contents and display mode, pushes the current foreground program into the background, switches into an 80- by 25-character text mode, clears the screen, and returns control to the TSR, which can then write to the screen at will. When the TSR no longer needs the display, it calls the function VIOENDPOPUP, which causes OS/2 to restore the previous screen contents, display mode, and foreground program (see listing 1).

Device Monitors

Ordinarily, the flow of data between a character device and an application program follows a well-defined pathway through the system (see figure 1). By the time a character reaches an application program, it has been filtered, buffered, and generally detoxified. To let a TSR intercept characters at a low level without running afoul of protected-mode hardware restrictions, OS/2 incorporates a set of special support functions for another unusual concept: device monitors. A program can use these functions to insert a probe into the character stream within a device driver, sifting the characters before they are passed upstream to the operating system, and thence to applications.

To become a device monitor, a program first calls the DOSMONOPEN function with the logical name of the character device it is interested in. The OS/2 kernel acts as the intermediary and asks the physical device driver whether it will support device monitors. If the driver approves the request, OS/2 returns a monitor handle. The program must then call DOSMONREG, passing the monitor handle and the addresses of two data buffers that will be used for input and output of device-driver data packets. An additional parameter to the DOSMONREG call specifies where the program would like to be in the chain of all monitors attached to this particular driver: first, last, or "don't care." If additional programs later request the first (or last) position, the first program to make the request comes first (or last) in the monitor chain, the second program comes next, and so on.

If the DOSMONREG call succeeds, the program has successfully registered as a device monitor and becomes responsible for passing the keyboard data through its monitor buffers. This is accomplished with the functions DOSMONREAD (which obtains a data packet from the driver that contains a character code, scan code, time stamp, and other information) and DOSMONWRITE, which returns a packet to the device driver. Typically, the program dedicates a high-priority thread to a tight loop that just performs successive DOSMONREADs and DOSMONWRITEs, to avoid degrading the keyboard response that the user perceives. Between each read and write, the program can inspect the character that is passing by, translate it, or consume it. The program can even add characters to the keyboard data stream by performing extra DOSMONWRITE calls.

When the program wants to unhook itself from the character stream it is monitoring, it calls DOSMONCLOSE with the monitor handle that was returned from the original DOSMONOPEN. The program can then release its other resources and terminate without any untoward effects. Listing 2 shows a skeleton of the code that such a TSR program would use.

Stamper: An Example

My example program, called Stamper, demonstrates the essential elements of an OS/2 pop-up utility. Stamper registers as a device monitor for the keyboard driver (KBD$) and then creates a new thread

continued

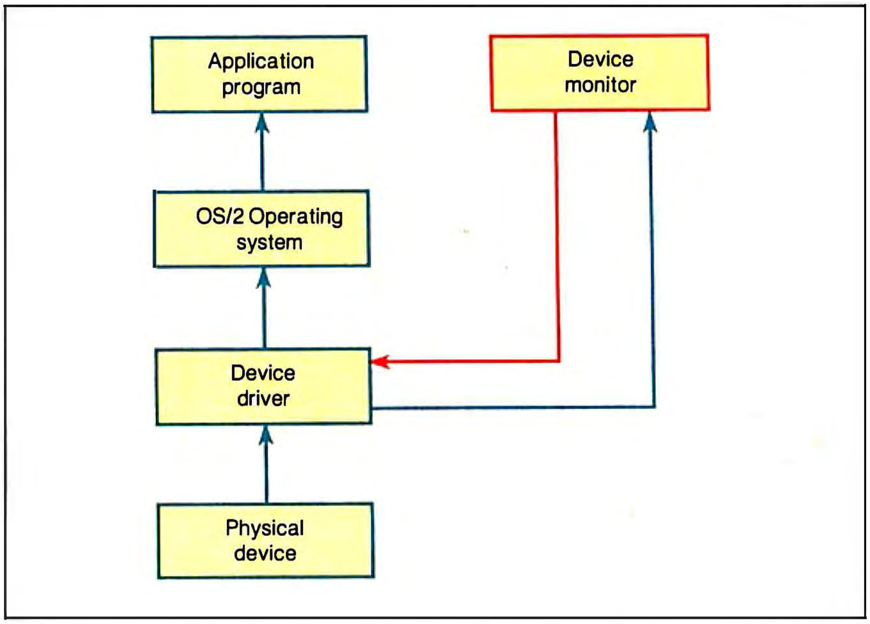


Figure 1: *The left side of this diagram shows the normal flow of data from a character device to an application. Registering a device monitor changes the flow to add the box on the right. The characters are passed from the driver to the device monitor in packets, where they can be translated or deleted, and new characters can be added. The characters are then passed back to the driver and from there flow through the operating system to the applications.*

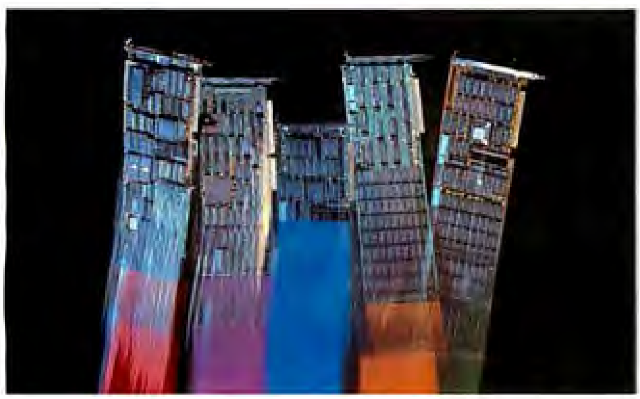
Listing 2: *A skeleton of the application code to create a device monitor function.*

```
KbdName     db    '\DEV\KBD$',0        ; keyboard logical device name
KbdHandle   dw    ?                    ; handle from DOSMONOPEN

                                       ; buffers for keyboard monitor
KbdMonIn    dw    128,64 dup (0)       ; monitor input buffer
KbdMonOut   dw    128,64 dup (0)       ; monitor output buffer

KbdPacket   db    128 dup (0)          ; holds keyboard data packet
KbdPktLen   dw    ?                    ; length of data in KbdPacket

ScrGroup    dw    ?                    ; current screen group number

            .
            .
            .
                                       ; open monitor connection...
            push  ds                   ; address of name \DEV\KBD$
            push  offset DGROUP:KbdName
            push  ds                   ; address to receive handle
            push  offset DGROUP:KbdHandle
            call  DOSMONOPEN           ; transfer to OS/2.
            or    ax,ax                ; was open successful?
            jnz   error                ; jump if open failed.

                                       ; register keyboard monitor...
            push  KbdHandle            ; handle from DOSMONOPEN
            push  ds                   ; address of monitor input buffer
            push  offset DGROUP:KbdMonIn
            push  ds                   ; address of output buffer
            push  offset DGROUP:KbdMonOut
            push  1                    ; request front of list.
            push  ScrGroup             ; push the screen group number.
            call  DOSMONREG            ; transfer to OS/2.
            or    ax,ax                ; was registration successful?
            jnz   error                ; jump if registration failed.
            .
            .
            .
nextchar:                              ; this loop is the actual
                                       ; keyboard monitor...

                                       ; set max length for read.
            mov   KbdPktLen,KbdPktLen-KbdPacket
                                       ; get next keyboard packet...
            push  ds                   ; address of input buffer
            push  offset DGROUP:KbdMonIn
            push  0                    ; wait until data available.
            push  ds                   ; address to receive packet
            push  offset DGROUP:KbdPacket
            push  ds                   ; address of length variable
            push  offset DGROUP:KbdPktLen
            call  DOSMONREAD           ; transfer to OS/2.
            or    ax,ax                ; did read succeed?
            jnz   error                ; jump if read failed.

                                       ; is the key the hot key?
            cmp   byte ptr KbdPacket+3,HotKey
            jz    ProcessKey           ; jump if it is...

                                       ; Not hot key, pass it on...
            push  ds                   ; address of output buffer
            push  offset DGROUP:KbdMonOut
            push  ds                   ; address of keyboard packet
            push  offset DGROUP:KbdPacket
            push  KbdPktLen            ; length of data packet
            call  DOSMONWRITE          ; transfer to OS/2.
            or    ax,ax                ; was write successful?
```

continued

Is your PC an endangered species?

With PC technology evolving at breakneck speed, it's become survival of the fastest. But before you decide your PC, XT or even AT is doomed to extinction, take a look at PC Technologies' complete family of enhancement boards:



386 Express™ — Our 16-MHz 80386 accelerator — specially suitable for LANs — will make your IBM AT two to three times more productive *and* maintain full software compatibility including OS/2, all at a remarkably low price — $995.

286 Express™ — The first half-slot 80286 accelerator card and winner of AFIPS/FORTUNE Magazine's Hardware Product-of-the-Year Merit Award. Engineered to deliver two to six times more speed for your PC, XT or compatibles.

286 RAMracer™ — Two performance upgrades, one low price. Combines 8-MHz 80286 acceleration *and* up to 2MB of EEMS/EMS- compatible expanded memory for your larger, demanding software applications.

RAMpartner™ — Batter down the DOS 640K memory barrier with up to 2MB of expanded memory. Ideal for spreadsheets, databases, windowing software . . . and as a companion to the RAMracer or AST and Intel expanded memory products.

286 Rainbow Plus™ — Our one-slot, five-function board contains an Enhanced Graphics Adapter, a 10-MHz 80286 accelerator, *PLUS* a Microsoft® InPort™ mouse interface, parallel printer port, and clock/calendar.

To find out how easy — and economical — it is to take your PC off the endangered species list, see your nearest PC Technologies dealer. Or call us today, direct, at 800-821-3086 (outside Michigan) or 313-996-9690.

PCT PC Technologies Inc.

704 Airport Blvd., Ann Arbor, MI 48108
313-996-9690/Telex 503589/FAX 313-996-0082,
800-821-3086 (outside Michigan)

Circle 169 on Reader Service Card

```
            jnz    error            ; jump if monitor write failed.

            jmp    nextchar         ; wait for another key.
            .
            .
            .
done:                               ; close monitor connection...
            push   KbdHandle        ; handle for monitor
            call   DOSMONCLOSE      ; transfer to OS/2.
            or     ax,ax            ; did close succeed?
            jnz    error            ; jump if close failed.
            .
            .
            .
```

that sits on top of the keyboard data stream watching for three particular key codes. Stamper also uses the `VIOPOPUP` and `VIOENDPOPUP` functions to announce its installation and eventual exit.

The source code for my example program is in the file STAMPER.ASM, and the module-definition file that describes the program's segment behavior is named STAMPER.DEF. [Editor's note: *Both files are available on disk, in print, and on BIX. See the insert card following page 208 for details. The listings are also available on BYTEnet. See page 4.*] To assemble, link, and run the Stamper utility, you will need the Microsoft OS/2 software development kit.

First, to assemble the file STAMPER.ASM into the relocatable object module STAMPER.OBJ, enter the command

```
[C:\] MASM STAMPER;
```

Next, to build the actual program STAMPER.EXE, use the Microsoft segmented executable linker to combine the file STAMPER.OBJ, the module-definition file STAMPER.DEF, and the dynalink reference file DOSCALLS.LIB as follows:

```
[C:\] LINK STAMPER,,,
   DOSCALLS,STAMPER
```

Finally, to run the utility, you can enter

```
[C:\] DETACH STAMPER
```

from any protected-mode screen group. The utility will briefly flash a sign-on message, followed by a new prompt from the system's protected-mode command interpreter (CMD.EXE). While you are running other programs, Stamper remains resident, waiting to act upon one of three Alt keys. When it sees an Alt-D key sequence, it substitutes the current date as a sequence of ASCII characters; the current time similarly replaces an Alt-T key sequence. When it detects an Alt-X key sequence, Stamper turns off keyboard monitoring and terminates.

Stamper is active only for the screen group in which it was loaded; it is not aware of keys entered when another screen group comes to the foreground with the session manager. However, you can load a copy of Stamper into each screen group you want to use it with. The overhead of such additional copies is minuscule, since OS/2 lets all the instantiations share the same machine code segment. With some extra work, you could also modify Stamper so that a single copy registers itself as a device monitor for all the screen groups in the system.

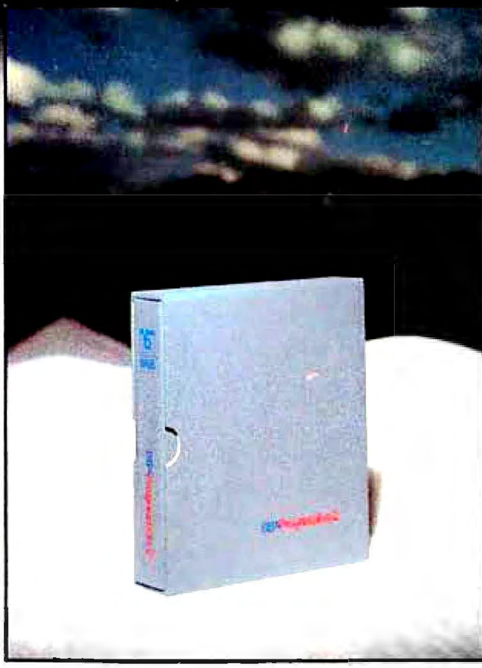
Summary

The past few years have seen the emergence of an entirely new class of MS-DOS program, the TSR pop-up utility. These utilities provide MS-DOS users with a variety of services that they can invoke with a hot key, regardless of the application program that is currently running. Unfortunately, because TSRs under MS-DOS must often capture interrupt vectors, depend on undocumented system functions, or exercise direct control over the hardware, unpredictable side effects are common.

OS/2 offers programmers a variety of special services that will expedite the creation of robust, powerful TSR-like utilities for protected mode and eliminate any need for direct hardware access. Over time, as the Windows/Presentation Manager graphical interface for OS/2 becomes dominant, pop-up TSRs as you know them today might fade away in favor of desktop utilities like those found on the Macintosh. In the meantime, TSR software houses should have a field day with the vastly superior memory resources and multitasking facilities available from the OS/2 kernel. ■

THIS IS BBx™.

THE MOST WELL KNOWN PROGRAMMING LANGUAGE YOU'VE NEVER HEARD OF.



If you're in the business of programming business solutions, you need to know about BBx.

Throughout the world, the industry's best and brightest programmers are discovering the power that BBx brings to Business BASIC.

And the numbers are growing. This year, over 30,000 copies of BBx are performing across the United States, Canada, Europe, Asia and South America.

BBx IS GOOD COMPANY WITH SOME GOOD COMPANIES.

The BBx name is well known to some of the most well respected names in the computer industry.

Open Systems, Altos and State of the Art have adopted the BBx language for their accounting packages. NCR selected BBx as their standard Business BASIC offering.

These diverse companies, with very different needs, depend on the wide-ranging capabilities of BBx and the availability of over 1,000 BBx vertical application packages ranging from hospitality to waste management.

IT MATTERS THAT YOU'RE RUNNING BBx. IT SELDOM MATTERS WHAT COMPUTER.

The power of BBx is unique in its enhancement of the Business BASIC language. You'll discover, though, that there need not be anything special about the computer you're programming.

BBx will make you a believer on most every major computer brand available today. And BBx is portable. With BBx, you have *guaranteed portability* of your application across all supported operating environments. This means a program written on a single-user PC can run with NO modification on over 35 different UNIX computers!

BBx IS SETTING NEW STANDARDS WITHIN A 15–YEAR STANDARD.

BBx is simply the most contemporary implementation of one of the most traditional products in computer programming.

Features like pop-up windowing on dumb terminals, device independent graphics, multi-keyed files, string arrays, long variable and function names, trigometric functions, matrix arithmetic, and others are simpler and swifter than you ever thought possible.

There are many other examples of BBx excellence like its callable business graphics utility set that makes graphics presentation of your data easy.

BBx IS THE SINGLE BIGGEST STEP YOU CAN MAKE TO OPTIMIZE YOUR BUSINESS BASIC LANGUAGE.

And it begins with this step. Call us at BASIS Incorporated. Find out how you can stay in step with the thousands of BBx programmers and bring an entire new dimension to your business applications.

We'll send you information on our just released BBx *Progression/2*, the newest, most advanced programming language from BASIS.

BBx *Progression/2* will do much more than add personality to your language. That's not the half of it.

BBx *Progression/2* will give your Business BASIC a whole new character.

To order BBx *Progression/2* in the United States please call TOLL FREE directly,
1-800-423-1394
In Canada, 1-416-494-0472

For technical support or more information,
1-505-821-4407

To order BBx *Progression/2* in Europe please contact one of our European distributors,

EDIAS, Hans Kirchhoff, Pfingstbornstrasse 25
6200 Wiesbaden, West Germany
Phone: (0 61 22) 2016 Telex: 418 25 63 edia d

PI Informatique, 8 rue Benjamin Constant
75019 Paris, France
Phone (1) 40.05 10.65 Telex: 214.583

MULTISYS A S, Torgeir Vraas Plass 5A
3044 Drammen, Norway
Phone: (03) 83 86 05 FAX: (03) 89 02 59

BASIS Incorporated
P.O. Box 20400
Albuquerque, New Mexico USA 87154
Electronic Bulletin Board 1-505-821-2933
MCI Mail, Basis (255-8242)
Telex 6502558242 mci

BBx is available on most popular UNIX, DOS and Network environments. Please call for assistance with your particular requirements.

UNIX is a trademark of Bell Laboratories.
NCR is a trademark of NCR Corporation.
Altos is a trademark of Altos Computer Systems.
Xenix is a trademark of Microsoft Corporation.

BBxProgression/2™

Persoft. Emulations. Communications. Utilities. And Wow.

Presenting some products you probably already know and one that will simply blow you away.

Our SmarTerm® line of PC-based emulations of DEC®, DG®, and HP terminals have earned acclaim as being state-of-the-industry by industry leaders. *Digital Review* even proclaimed SmarTerm 240 as its 1987 Target Award winner for Best Connectivity Software Product.

SmartMOVE® is the cost-efficient PC-to-the-rest-of-the-World communications package that doubles as a precise VT100 terminal emulator.

Referee® earned the Software Publishers Association's 1986 Excellence in Software Award for Best Utility. It's also earned countless kudos from users for keeping their RAM-resident programs under control.

Now there's IZE™. For you, IZE will open a new era of productivity at your keyboard. Its textbase management and organization will give you greater vision. When you see what our new IZE can do, you won't believe your eyes.

So visit booth H8706 at this Fall's Comdex. If what we are showing knocks you out, we'll just grab the smelling salts and fan you with a spec sheet.

In other words, don't miss what we have to show at the show. Period.

If you miss Comdex, use inquiry codes 50 (Emulation) and 51 (IZE), visit your dealer, or phone us at 608-273-6000.


smarterm
240
persoft
Communications Software
smartmove
persoft
Referee
persoft
IZE
IZE
IZE
Greater Vision

QNX vs. OS/2 UNIX

QNX: Bend it, shape it, any way you want it.

ARCHITECTURE If the micro world were not so varied, QNX would not be so successful. After all, it is the operating system which enhances or limits the potential capabilities of applications. QNX owes its success (over 30,000 systems sold since 1982) to the tremendous power and flexibility provided by its modular architecture.

Based on message-passing, QNX is radically more innovative than UNIX or OS/2. Written by a small team of dedicated designers, it provides a fully integrated multi-user, multi-tasking, networked operating system in a lean 148K. By comparison, both OS/2 and UNIX, written by many hands, are huge and cumbersome. Both are examples of a monolithic operating system design fashionable over 20 years ago.

MULTI-USER OS/2 is multi-tasking but NOT multi-user. For OS/2, this inherent deficiency is a serious handicap for terminal and remote access. QNX is both multi-tasking AND multi-user, allowing up to 16 terminals and modems to connect to any computer.

INTEGRATED NETWORKING Neither UNIX nor OS/2 can provide integrated networking. With truly distributed processing and resource sharing, QNX makes all resources (processors, disks, printers and modems anywhere on the network) available to any user. Systems may be single computers, or, by simply adding micros without changes to user software, they can grow to large transparent multiprocessor environments. QNX is the mainframe you build micro by micro.

PC's, AT's and PS/2's OS/2 and UNIX severely restrict hardware that can be used: you must replace all your PC's with AT's. In contrast, QNX runs superbly on PC's and literally soars on AT's and PS/2's. You can run your unmodified QNX applications on any mix of machines, either standalone or in a QNX local area network, in real mode on PC's or in protected mode on AT's. Only QNX lets you run multi-user/multi-tasking with networking on all classes of machines.

REAL TIME QNX real-time performance leaves both OS/2 and UNIX wallowing at the gate. In fact, QNX is in use at thousands of real-time sites, right now.

DOS SUPPORT QNX allows you to run PC-DOS applications as single-user tasks, for both PC's and AT's in real or protected mode. With OS/2, 128K of the DOS memory is consumed to enable this facility. Within QNX protected mode, a full 640K can be used for PC-DOS.

ANY WAY YOU WANT IT QNX has the power and flexibility you need. Call for details and a demo disk.

THE ONLY MULTI-USER, MULTI-TASKING, NETWORKING, REAL-TIME OPERATING SYSTEM FOR THE IBM PC, AT, PS/2, THE HP VECTRA, AND COMPATIBLES.

Multi-User	10 (16) serial terminals per PC (AT).	**C Compiler**	Standard Kernighan and Ritchie.
Multi-Tasking	40 (64) tasks per PC (AT).	**Flexibility**	Single PC, networked PC's, single PC with terminals, networked PC's with terminals. No central servers. Full sharing of disks, devices and CPU's.
Networking	2.5 Megabit token ring. 255 PC's and/or AT's per network. 10,000 tasks per network. Thousands of users per network.		
Real Time	2,800 task switches/sec (AT).	**PC-DOS**	PC-DOS runs as a QNX task.
Message Passing	Fast intertask communication between tasks on any machine.	**Cost**	From US $450. Runtime pricing available.

QNX

For further information or a free demonstration diskette, please telephone (613) 591-0931.

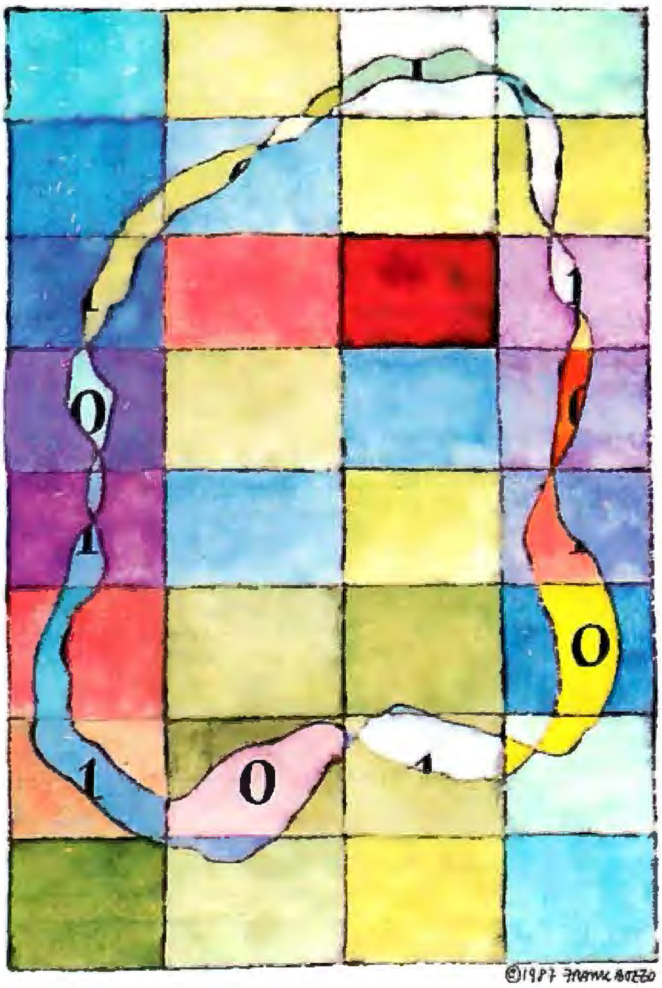
Quantum Software Systems Ltd. • Kanata South Business Park • 175 Terrence Matthews Crescent • Kanata, Ontario, Canada • K2M 1W8

Circle 190 on Reader Service Card

The Personal System/2 Model 80 takes the 32-bit road to more power, more memory, more speed

The 32-bit Micro Channel

Jon Shiell



The 32-bit Micro Channel opens a window on IBM's future. The three 32-bit bus slots in the Personal System/2 Model 80 support the 80386 processor with 32-bit addressing and 32-bit data transfers. The direction in which IBM's personal computers are headed is no longer implied; it is defined: the 32-bit road to more power, more memory, more speed. No big surprise—but the question of which way IBM will go and how it will get there is always of major interest to anyone seriously involved with personal computers.

The Personal System/2

As I write this, the new IBM PS/2 family consists of five basic machines, three of which are based on a proprietary new bus called the Micro Channel. The Models 25 and 30 are 8-megahertz 8086-based desktop machines that use an IBM PC XT bus with three slots. The next two are 10-MHz 80286-based machines: the Model 50, a desktop model with four 16-bit Micro Channel slots, and the Model 60, a floor-standing model with seven 16-bit Micro Channel slots. Finally, the Model 80 is an 80386-based floor-standing machine available in both 16-MHz and 20-MHz versions with three 32-bit and four 16-bit Micro Channel slots.

Previous BYTE articles have covered the Model 30 (see "The IBM PS/2 Model 30" by Curtis Franklin Jr., July); the Models 50 and 60 (see "The IBM PS/2 Model 50" and its accompanying text box on the Model 60, by Richard Grehan, July); the 16-bit Micro Channel (see "Under the Covers" by Steve Ciarcia, August); and information on the PS/2 in general (see "First Impressions: The IBM PS/2 Computers," June).

Because of this heavy coverage, I'll focus on the unique features of the Model 80's 32-bit bus and not attempt a comprehensive overview of the Micro Channel. Information for this article came from a variety of sources including the *IBM Personal System/2 Model 80 Technical Reference*.

The 32-bit Difference

The Model 80's Micro Channel differs from that of the Model 50 and 60 in that, in addition to 16-bit slots, it also has 32-bit slots. The Micro Channel connector for the 32-bit extension extends the 16-bit Micro Channel connector to accommodate 32-bit addressing and 32-bit data transfers (see figure 1). It connects to the "bottom" of the 16-bit extension (see figure 1 in "Under the Covers," August, page 104).

The 32-bit extension itself consists of the seven control lines for 32-bit data (-BE0 through -BE3, -CD DS 32, -DS 32 RTN, and TR 32), eight additional address lines (A24 through A31), and 16 data lines (D16 through D31).

Lines -BE0 through -BE3 ("byte enable" 0 through 3) are used during 32-bit slave data transfers to tell the bus which bytes are to go on it. Line -CD DS 32 (card data size 32) indicates that the data port at the location addressed is a 32-bit data port. Line -DS 32 RTN (data size 32 return) is a negative OR of -CD DS 32 and provides a check for the channel on data-size information. TR 32 (translate 32) provides an indicator as to what logic is driving -BE0 through -BE3. If TR 32 is inactive, the 32-bit bus master is in charge of these lines; if it is active, the central-translator logic is driving them.

continued

Jon Shiell is a contributing editor for BYTE (One Phoenix Mill Lane, Peterborough, NH 03458). He can also be reached on BIX as jshiell.

There is also a Micro Channel connector for the matched-memory extension. This connector extends the 32-bit Micro Channel connector to accommodate matched-memory cycles (see figure 2). It connects to the "top" of the 8-bit section of the Micro Channel in the same position that an auxiliary video extension might occupy on a 16-bit Micro Channel.

The matched-memory-cycle section consists of three signals, -MMC, MMCR, and -MMC CMD. Signal -MMC (matched-memory cycle), driven by system-board logic, indicates that the CPU is in control of the bus and can run a matched-memory cycle. Signal -MMCR (matched-memory-cycle request) is driven by a slave on either the 16-bit or 32-bit channel to request a faster cycle. Since the 80386 is the only controlling device allowed to run matched-memory cycles, if an 8-bit or 16-bit channel slave—or a 32-bit slave in a nonmicroprocessor bus cycle—requests -MMCR, the system will run a basic-transfer cycle. Signal -MMC CMD (matched-memory-cycle command) defines when the data on

continued

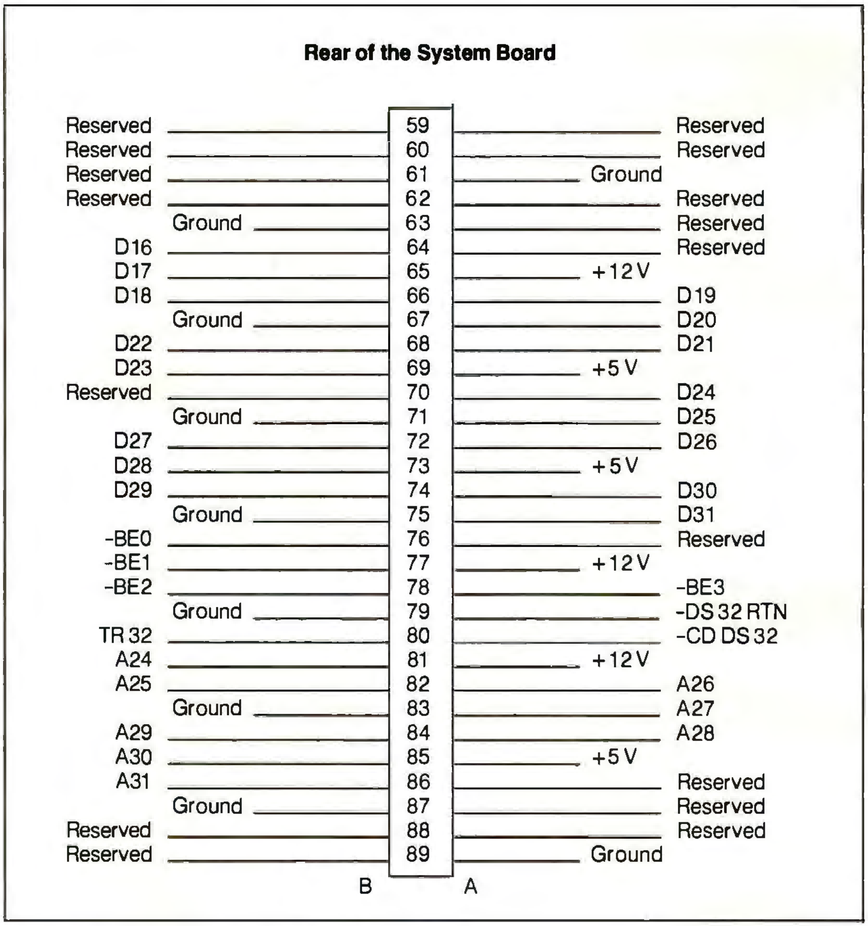


Figure 1: *The 32-bit Micro Channel extension.*

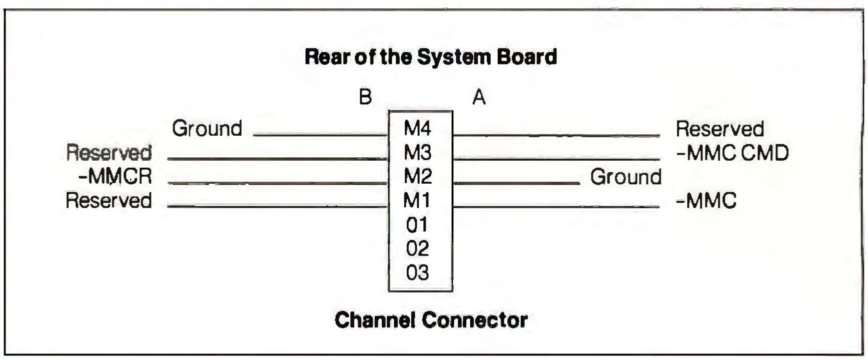


Figure 2: *The matched-memory extension to the 32-bit Micro Channel.*

It's not if you'll lose data.

It's when.

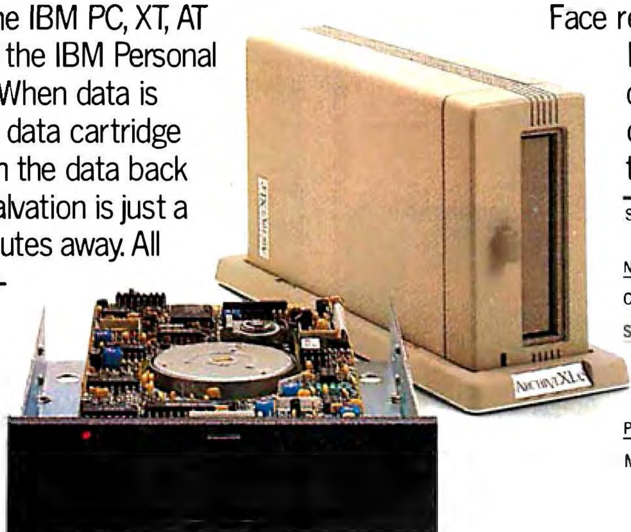
Head crashes. Software glitches. Operator errors.

They're the grim realities of computer life in the 80's. To protect yourself from the inevitable data losses, get an ArchiveXL or ArchiveXLe.

These simple, low-cost tape backup systems are specially designed for the IBM PC, XT, AT and compatibles, as well as the IBM Personal System/2 product families. When data is lost, just insert your backup data cartridge in the tape drive and stream the data back onto your hard disk. Data salvation is just a few keystrokes and just minutes away. All this protection takes up precious little space—even as little as a 3.5 inch floppy disk drive. And the ArchiveXL systems are complete with everything you need for maximum data protection—including Archive's QICstream™ utility software.

A final thought, the ArchiveXL and ArchiveXLe are based on QIC-40, the tape interchange standard for data compatibility. That's a big Archive extra.

Face reality, you need tape backup. Find out today how little a lot of personal data protection costs. For more details call toll-free 1-(800) 237-4929.



Sure, I want to know more about the ArchiveXL and ArchiveXLe. Please send me an information kit ASAP.

NAME ____________

COMPANY ____________

STREET ____________

CITY ____________

STATE ________ ZIP ________

PHONE ( ) ____________

Mail to: Archive Corporation
1650 Sunflower Ave.
Costa Mesa, CA 92626

ARCHIVE®
CORPORATION

Circle 17 on Reader Service Card (Dealers: 18)

The Micro Channel uses the 80286's addressing scheme for 8-bit and 16-bit transfers and the 80386's scheme for 32-bit transfers.

the bus is valid, but only during a matched-memory cycle. Together, these signals extend the Micro Channel to use full 32-bit addresses and data.

Photos 1a through 1c provide a telescopic view of the Micro Channel starting with an inside view of the Model 80 (photo 1a), zooming in to a closer view that contains the Micro Channel on the left and shows the 80386 and optional 80387 chips on the right (photo 1b), and ending with a close-up of the Micro Channel (photo 1c). Photo 1c shows the three 32-bit channel connectors, 1, 2, and 4 (counting from the bottom); notice the short extension on the left end of each, the matched-memory extension. Also note the longer extension to the left in slot 6; this is a 16-bit channel connector with the auxiliary video extension.

How the 32-bit Bus Works

The PS/2 Model 80 has special logic, called the address-bus translator, that lets 16-bit devices communicate with 32-bit slaves, and vice versa. The 32-bit slaves (and 32-bit devices) use -BE0 through -BE3 instead of the A0 (address bit 0, the least-significant address bit) and -SBHE (system byte high enable, which indicates and enables data transfer on the high byte of the 16-bit data bus—i.e., D8 through D15) bus lines of the 16-bit bus. A0 and -SBHE are used together to distinguish between high-byte (D8 through D15) and low-byte (D0 through D7) data transfers on the 16-bit bus.

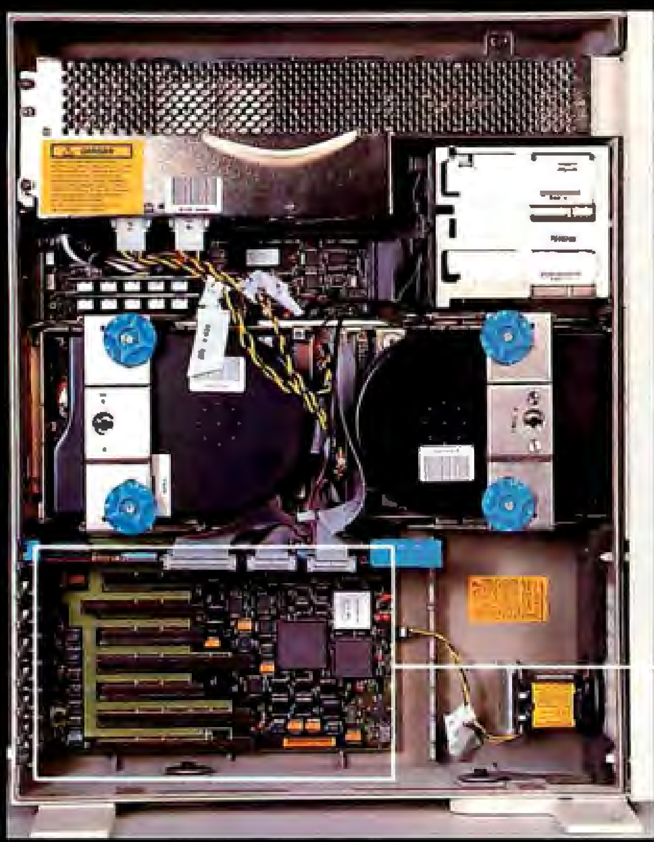
Sixteen-bit and 32-bit transfers use different signals because the 80286 (16-bit) and 80386 (32-bit) system microprocessors address memory differently. The Micro Channel uses the 80286's addressing scheme for 8-bit and 16-bit accesses and the 80386's scheme for 32-bit transfers. Thus, no translation is required when a processor accesses native memory; however, when an 80386 accesses 16-bit memory or I/O, translation is needed.

The signal TR 32 is driven inactive by 32-bit devices; this signal is used by the address-bus translator along with the -CD DS 16 (card data size 16) and -CD DS 32 (card data size 32) signals (which indicate the slave's data size) to determine if translation is required and which party is 32-bit. In addition to the address-bus translator, data-bus-steering logic is required to cross data between D16 through D31 and D0 through D15 because 16-bit devices (and slaves) don't use the high-order 16 data lines.

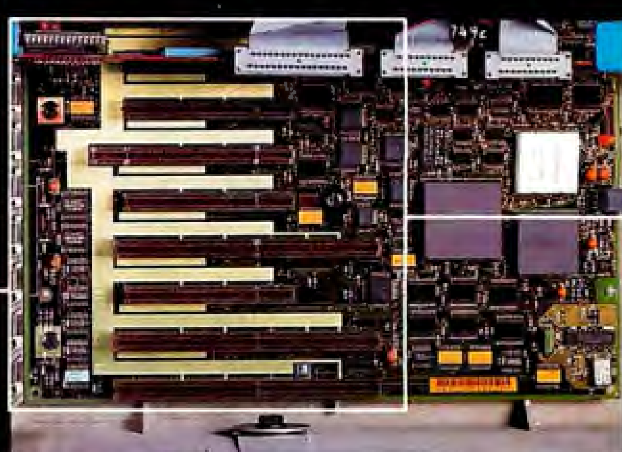
Four different types of bus cycles are defined for the 32-bit Micro Channel. In order from fastest to slowest, they are matched-memory cycles, basic-transfer cycles, synchronous extended-transfer cycles, and asynchronous extended-transfer cycles. While the Micro Channel is defined as an asynchronous bus, the first three of these cycles are synchronous special cases.

The Matched-Memory Cycle

The matched-memory cycle is a synchronous cycle supported only by the Model 80. It provides the most efficient data transfer between the 80386 and the Micro Channel and is this bus's equivalent to the PC AT's zero-wait-state memory. The system board's ROM, the 80386's RAM, and the 32-bit memory-expansion adapter follow the matched-memory-cycle



◀ **Photo 1a:** *Overall view of the inside of the Model 80.*



▼ **Photo 1b:** *Enlarged view of the box in photo 1a, showing the Micro Channel on the left and the 80386 and 80387 chips on the right.*

Photo 1c: *Enlarged view of the box in photo 1b, showing the three 32-bit connectors and four 16-bit connectors in the Model 80's Micro Channel.* ▶

protocol (i.e., -MMC, -MMCR, and -MMC CMD). A channel slave can be either a memory slave or an I/O slave with either a 16-bit or 32-bit data bus width. However, 8-bit devices cannot run matched-memory cycles, nor can DMA.

Two types of 80386 bus cycles are supported on the Micro Channel for the Model 80: matched-memory cycles and basic-transfer cycles. A matched-memory cycle occupies at least three 16-MHz clock cycles, or 187.5 nanoseconds. A basic-transfer cycle takes at least four 16-MHz clock cycles, or 250 ns. The channel slave must issue an -MMCR request for each bus cycle if it wants a matched-memory cycle; if it doesn't, it will receive a basic-transfer cycle as the default.

If the channel slave issues an -MMCR request, then the system microprocessor responds with -MMC CMD. If the channel slave doesn't return the -MMCR request, the system runs a basic-transfer cycle. These two 80386 bus cycles can be mixed and matched any way you wish; the process is totally dynamic and is determined on a cycle-by-cycle basis.

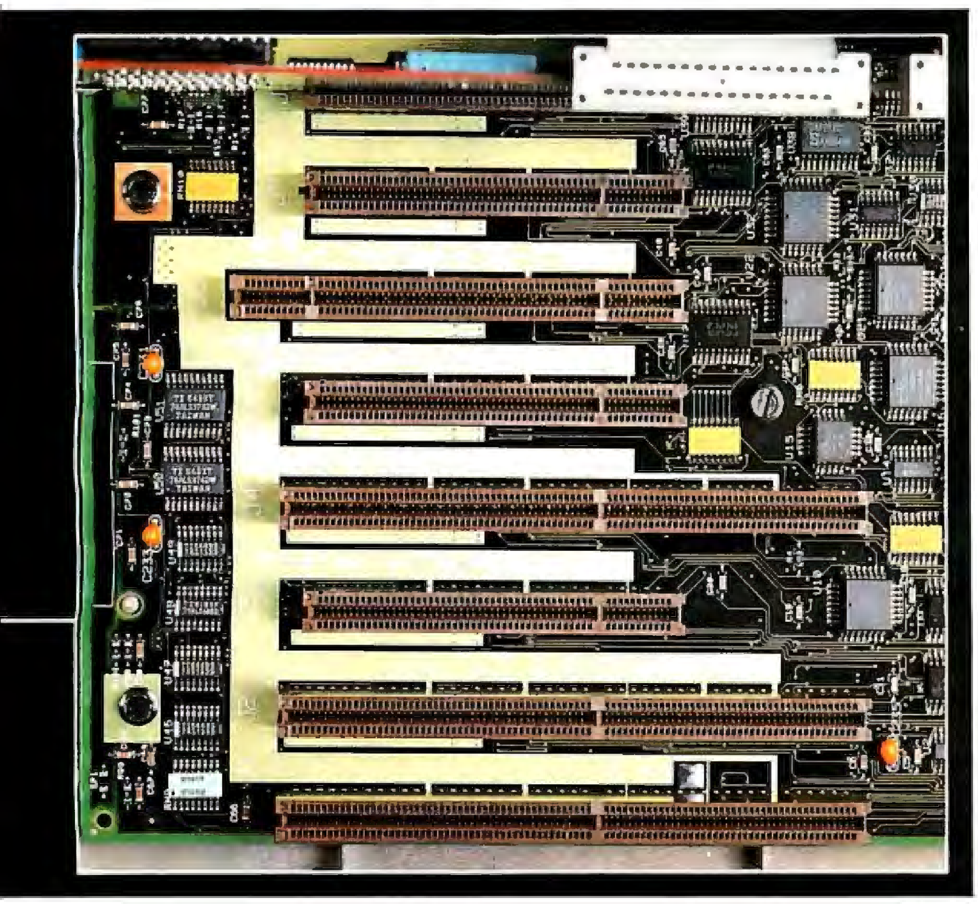
You can extend bus cycles until -CD ChRdy (channel ready) is found active. A memory or channel slave can set -CD ChRdy inactive to allow more time to complete a matched-memory cycle or a basic-transfer cycle when the default length of the cycle is not long enough.

A warning from the technical reference: "When MMC is active, matched-memory-cycle 32-bit and 16-bit devices should not use -MADE 24, A0, A1, or -SBHE in logic that generates -MMCR, -CD DS 16/32, -SEL FBK, and -CD ChRdy."

Variations on a Theme

The line -CD ChRdy is used by slave programs to tell the Micro Channel when their data is ready. It may take as much as 3 microseconds for a slave to make -CD ChRdy active. Data transfers other than matched-memory transfers generally use the same control sequence.

1. The address bus, -MADE 24 (memory-address enable 24, which indicates when an extended address is used on the bus), M/-IO (memory/input output, which distinguishes a memory cycle from an I/O cycle), and -Refresh (if applicable) become active. The cycle begins.
2. The status signals, -S0 and -S1, become active. (Bus addresses must be valid before either -S0 or -S1 becomes active.)
3. The -ADL (address-decode latch) signal becomes active.
4. In response to active -ADL, -MADE 24, and M/-IO, the adapter returns -CD Sfdbk (card-selected feedback, which indicates, when active, that the slave addressed by the system microprocessor is present at the address specified), -CD DS 16 if the attachment can handle 16-bit operation, and both -CD DS 16 and -CD DS 32 if it can handle 32-bit operation.
5. In response to active -ADL, -MADE 24, M/-IO, -S0, and -S1, the adapter drives -CD ChRdy inactive if the bus cycle is too short for the transfer and needs to be extended.
6. For a Write cycle, the Write data appears on the Micro Channel. (Various combinations of M/-IO, -S0, and -S1 indicate whether a transfer is a read or write operation and whether it is to or from memory or I/O.)
7. -CMD (command, which is used to determine when data on the data bus is valid) becomes active, and -ADL becomes inactive.
8. -S0 and -S1 become inactive.
9. Address signals become inactive in preparation for the next bus cycle.
10. In response to an address change, the attachment sets -CD Sfdbk, -CD DS 16, and -CD DS 32 inactive.
11. If -CD ChRdy has been set inactive, the system remains in this state until -CD ChRdy is set active. (This interval should not exceed 3 ms.)
12. For a read cycle, the attachment places read data on the Micro Channel to be ready for the trailing edge of -CMD.
13. The address, -S0, -S1, and M/-IO for the next cycle may become active. (You can overlap activating the address and status indicators with the preceding cycle to minimize the impact that memory-access time makes on performance.)
14. -CMD goes inactive. The cycle ends.



The remaining three bus cycles (the basic-transfer cycle, the asynchronous extended-transfer cycle, and the synchronous extended-transfer cycle) are variations on this theme. The basic-transfer cycle is the default bus cycle and is synchronous. While it requires a minimum of 200 ns, it requires four clock cycles of 62.5 ns each, or 250 ns, in the 16-MHz Model 80. The asynchronous extended-transfer cycle and the synchronous extended-transfer cycle are differentiated by the slave's use of -CD ChRdy. If it is an asynchronous cycle, the slave releases -CD ChRdy asynchronously and provides the read data within 60 ns of the release. Similarly, if the cycle is synchronous, the release is synchronous and occurs within 30 ns of the leading edge of -CMD. While the asynchronous extended-transfer cycle requires a minimum of more than 300 ns, the synchronous extended-transfer cycle is only slightly faster, requiring a minimum of 300 ns.

continued

You could have an 18-megabyte system with one 32-bit slot still available.

A special note: If activation of the status indicators overlaps with the previous -CMD cycle in the two extended cycles, -CD ChRdy is invalid during the overlap. This varies from the control sequence described above.

Details to Remember

The 32-bit Micro Channel is a superset of the 16-bit Micro Channel; thus, POS (Programmable Option Select), arbitration, and timing are the same (except in the matched-memory cycle). One point worth remembering: While bus masters on the Micro Channel can access all *memory* addresses on the system board, they can't access *I/O* addresses less than 100 hexadecimal on the system board; this is true of all Micro Channel systems, not just the Model 80.

While the Model 80's Micro Channel supports 32-bit addresses for the 32-bit slots, the system DMA channels on the Model 80 support only 24-bit addresses. This can be a problem if you want to use more than 16 megabytes of memory; all DMA (i.e., disk I/O and so forth) must be moved to low storage, then moved by the processor to high storage. The memory-remapping facility may provide a way around this by letting you remap banks of memory between high and low addresses. That is, the operating system might reserve, for example, four banks of memory between 8 megabytes and 12 megabytes for remapping; memory remapping in the Model 80 is in units of 1 megabyte. Then, when the operating system needs to perform DMA I/O at a high memory location—the bank at physical address 22 megabytes, for example—it could remap that bank to one of the four banks reserved for remapping, perhaps the one at location 10 megabytes. After the I/O completes, the operating system moves the relocated bank of memory back into its original high-memory position.

This problem is unlikely to arise in the next few years, because OS/2 and other 80286 operating systems can address up to 16 megabytes, as can the 80286 itself. Also, the performance that the 80386 provides, whether at 16 or 20 MHz, pipelined with one wait state, won't require more than 16 megabytes of memory in most cases.

Future Directions?

The three 32-bit slots in the Model 80 are primarily meant for memory cards; assuming the use of 1-megabit dynamic RAM chips, a normal memory card will contain about 8 megabytes of memory. Using two slots for memory and up to 2 megabytes on the motherboard, you could have an 18-megabyte system with one 32-bit slot still available. What can you do with that third 32-bit slot? Well, you could use it for an expansion chassis or another 8 megabytes of memory. But you could also presumably use it for an additional 32-bit processor (with its own memory cache); however, the current Model 80 doesn't have the bus bandwidth to support this.

Let's assume we have a Model 80-71; that is, a 16-MHz 80386. The 80386 runs pipelined with one wait state; according to Intel, that gives a bus utilization of about 86 percent. RAM Refresh takes less than 5 percent of the bus (see table 1), leaving about 10 percent of the bus bandwidth available for DMA transfers when the processor is 100 percent busy. Ten percent of a second is 100 ms for DMA each second. Since the processor can use the 300-ns arbitration time to access memory, as can RAM Refresh, we will count only DMA transfer time. At 400 ns per transfer, this gives 100 ms/400 ns or 25,000 DMA transfers per second; this would be about 50K bytes per second if we used 16-bit transfers. If you assume an I/O rate of 90K bytes per second as being two-thirds 16-bit transfers and one-third 8-bit transfers, you get 300 ms/400 ns or a total of 75,000 transfers per second. That's 300 ms of each second for DMA transfers, plus 50 ms for RAM Refresh, leaving 650 ms for the processor.

If we have 100 percent processor utilization, the 80386 uses 86 percent of the bandwidth; so if only 65 percent of the bandwidth remains, the 80386 can run at only about 75 percent utilization. Under normal operation, a system microprocessor runs nearer to 75 or 80 percent than 100 percent busy anyway. However, this exercise points out an interesting fact: The Micro Channel doesn't have the bandwidth to support multiple processors unless all the processors use caches to cut the required bus bandwidth.

If you have a 64K-byte direct-mapped buffered store through cache using a 16-byte line, you will have a hit rate of about 96 percent, assuming four reads for every write. The 80386's 86 percent bus utilization in this case decreases to about 33 percent; that is, $0.86 \times (4 \times 0.04 \times 0.80$ 16-byte reads $+ 0.20$ 4-byte writes$) = 33$ percent, where the 4 equals the four 32-bit reads required to fetch a line, and the 0.04 is the miss rate (1 minus the hit rate of 0.96). I assumed the memory can burst matched-memory cycles. This ignores the effects of device bursting and makes the processor wait till memory is free. Thus, the Micro Channel appears to have enough bandwidth to support up to two processors and their I/O. If future PS/2 systems use a cache for the system processor, multiprocessor systems will be much more viable—and I suspect that they won't have the 24-bit DMA limit that the current Model 80 has. ■

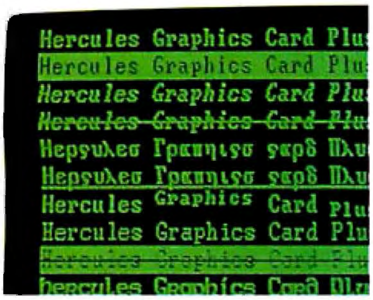
Table 1: *The Model 80's internal timings.*

	Model 80-41, 71	Model 80-111
Processor clock speed	16 MHz	20 MHz
Minimum system board RAM access time	187.5 ns	100 ns
Minimum system board ROM access time	187.5 ns	100 ns
Minimum system board I/O time	500 ns	200 ns
Minimum system board video (8 bits)	2000 ns	700 ns
Basic bus cycle	250 ns	200 ns
Arbitration cycle time (minimum)	375 ns	300 ns
DMA		
Minimum bus burst timing	375+500*n* ns	300+400*n* ns
System board burst timing	375+625*n* ns	300+400*n* ns
Bus master		
Minimum bus burst timing	375+387.5*n* ns	300+200*n* ns
System board burst timing	375+625*n* ns	300+400*n* ns
Refresh cycle	8 MHz, 5%	8 MHz, 5%

Where *n* is the number of doublewords, words, or bytes transferred.

Improve Your Word Processor With Hercules.

Now your word processor can have full display and printing flexibility without slowing down. The secret is the exclusive Hercules graphics card mode called RamFont. Which isn't really graphics at all.



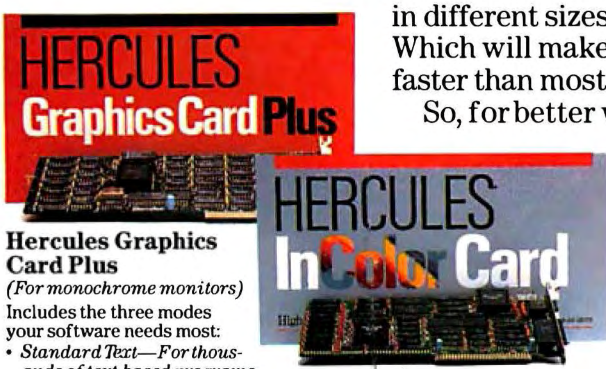
RamFont lets most popular word processors display italics, subscripts, superscripts, foreign languages, different font styles and symbols etc. at lightning speed.

You see, graphics mode is slow. That's why most IBM compatible word processors run in fast, but limited text mode (they can only display 256 ASCII characters).

No Compromises with RamFont.

Some choice. Slow down software by using graphics mode to display italics, subscripts, superscripts, large characters, different fonts, etc. Or forget all that in favor of speedy text mode.

Hercules solved the problem without compromise some time ago ... almost 200,000 graphics cards ago to be specific. Because in addition to text mode and 720x348 graphics, we include RamFont. Today's standard for text-based programs. RamFont gives the flexibility of graphics at the speed of text by displaying 3072 software defined characters instead of ASCII's limited 256. Available in two versions for your PC: the Hercules Graphics Card Plus (for monochrome TTL monitors) and the Hercules InColor Card (for "EGA-type" and multi-synchronous monitors).



Hercules Graphics Card Plus
(For monochrome monitors)
Includes the three modes your software needs most:
- *Standard Text—For thousands of text-based programs*
- *Hercules 720x348 graphics—highest popularly supported standard*
- *Hercules RamFont—3,072 software definable characters improve many popular software programs. (Call for the latest list.)*

Improve Today's Software.

RamFont runs Microsoft Word complete with its excellent on-screen display of attributes and fonts. But there's a difference. It's up to four times faster than the graphics mode in which it was written. Only Hercules cards let Lotus Manuscript display boldface, italics, subscripts, superscripts, and many equations in text processing mode. And RamFont expands Broderbund ForComment's viewing area to a full 80x43.

Programs like Design Enterprise's Alexander use it to display symbols (musical notes and foreign languages) which, before RamFont, were possible only in the much slower graphics mode. Other programs including Nota Bene, Word-MARC, and Qalam all have optimized displays with RamFont.

Tomorrow's software will display large and small characters in an almost infinite variety of styles at speeds heretofore impossible.

In fact, every word processor can run better with Hercules. Independent add-on packages (TurboFonts, the Enhance Star series, etc.) let word processors like Word Perfect, Multimate, WordStar, and Displaywrite take advantage of the same sort of display and printing flexibility.

Improve The Software of Tomorrow.

WYSIWYG for underlining, italics, subscripts, font display, etc. is just the beginning. A whole new generation of word processors will use RamFont to mix a nearly infinite variety of fonts in different sizes on-screen at sizzling text speed. Which will make them look like, but run much faster than most of today's desktop publishers.

So, for better word processing, ask for Hercules with RamFont. It's the most compatible way to improve your writing.

Hercules InColor Card
(For multi-synchronous and enhanced color "EGA-type" monitors.)
Runs all Hercules monochrome software in 2 colors, most popular software in up to 16 colors selected from a palette of 64.
- *Standard Text—All programs run in 2 or more colors.*
- *Hercules 720x348 graphics in up to 16 colors—15% higher than EGA.*
- *Hercules RamFont—3,072 software definable characters in 16 colors up to 12,288 in 2 colors.*

For more information call Hercules toll-free at:
1-800-532-0600 Ext. 304. In Canada, 1-800-323-0601 Ext. 304.

Hercules
We run your software better.

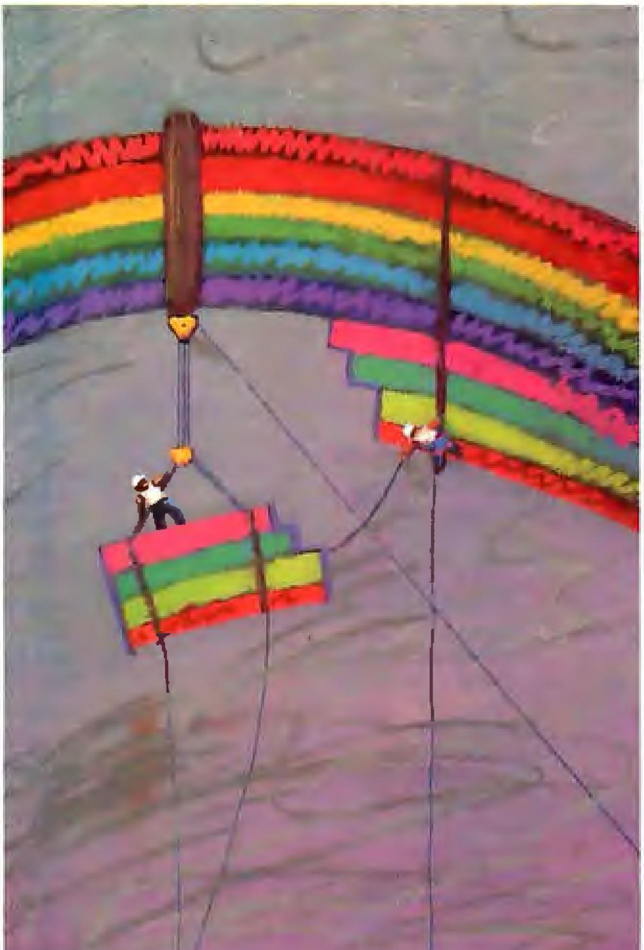
Hercules Computer Technology, 921 Parker Street, Berkeley, California 94710; Tech Support: (415) 540-0749; Sales: (415) 540-0212. Hercules, RamFont and InColor are trademarks of Hercules Computer Technology. Other products are trademarks of their respective holders.

Circle 98 on Reader Service Card (Dealers: 99)

The ins and outs of the PS/2 machines' video systems

PS/2 Video Programming

Richard Wilton



The IBM PS/2 series introduces two new video subsystems, the multicolor graphics array (MCGA) and video graphics array (VGA). This article is an overview of the MCGA and the VGA from a programmer's point of view. If you are already familiar with older video adapters, such as the CGA, EGA, or Hercules cards, this article will point out the similarities and differences between the PS/2 video subsystems and previous IBM video adapters. If you are new to video hardware programming, you can use the examples in this article as a focus for further exploration of the PS/2 hardware.

Unlike the IBM PC, XT, and AT, into which you must install a separate card that supports the necessary hardware to drive a video display, all the PS/2s are equipped with a built-in video subsystem on the motherboard. The Model 30 comes with the MCGA, while the Models 50, 60, and 80 use the VGA.

For compatibility (and, no doubt, in hopes of selling lots of hardware), IBM also offers a VGA adapter that implements the VGA subsystem on a card for the XT, AT, or PS/2 Model 30.

Monitors

The MCGA and VGA differ from previous IBM video adapters in that both require that you use an analog monitor instead of a digital monitor. Adapters such as the CGA and the EGA use digital monitors in which the RGB color signals generated by the adapter are digital signals (on or off). This limits the number of different colors that the subsystem can display. For example, IBM's enhanced color display, which is driven by six RGB signals as generated by an EGA, can display a total of 64 (2^6) different colors.

In contrast, the PS/2-compatible monitors use RGB signals with voltage levels that are continuously variable instead of simply on or off. Because the displayed brightness of a color corresponds to the voltage level of the color drive signals, an analog monitor can display a much larger variety of colors.

IBM offers one monochrome and two color monitors for use with the MCGA and the VGA. You can use a monochrome or color monitor with either video subsystem. You can also use EGA-compatible monitors with analog capability, such as the NEC MultiSync and Sony MultiScan monitors, with the MCGA and VGA.

Compatibility

Programs that run on the CGA can run unchanged on the MCGA, even if they bypass the video BIOS and program the hardware directly. Also, because the PS/2 Model 30 has a PC-compatible bus, a monochrome display adapter or Hercules adapter can coexist with the MCGA in the PS/2 Model 30.

The VGA is similar in its programming interface to the EGA. Its control ports and buffer addressing are EGA-compatible, so programs that run on an EGA generally run on a VGA as well. The VGA is compatible enough with the EGA at the hardware level that the VGA can usually run ill-behaved programs that access the EGA control registers directly.

From a programmer's perspective, there is not much resemblance between the MCGA and the VGA. The I/O port assignments in the MCGA and VGA differ significantly. So does the layout of

continued

Richard Wilton (6236 West Sixth St., Los Angeles, CA 90048) is author of The Programmer's Guide to PC and PS/2 Video Systems, *which is due out in November.*

video RAM in the two subsystems. A program that bypasses the video BIOS to control the hardware directly will probably not run on both the MCGA and the VGA unless it contains special code for programming each subsystem independently.

Documentation

The programming interface to PS/2 video hardware is documented in the IBM technical reference manuals for the Models 30, 50, and 60. The video BIOS is covered by a separate set of IBM reference manuals, the *Personal System/2* and *Personal Computer BIOS Interface Technical Reference*. Obviously, this article does not cover all the details of the video hardware implementation. If you need to understand the hardware or firmware in detail, you should obtain the appropriate IBM technical manuals.

The MCGA

The heart of the MCGA circuitry lies in two proprietary gate arrays: the memory-controller gate array, which incorporates the functions of a CRT controller, and the video-formatter gate array, which controls video mode selection and color-attribute decoding.

You can program the memory controller through a set of 8-bit registers (see table 1) mapped to I/O ports 3D4 and 3D5 hexadecimal. [Editor's note: *For the remainder of this article, addresses will be in hexadecimal.*] As on the CGA, you access the registers by first writing the register number to the port at 3D4, and then writing or reading the specified register at 3D5. Unlike the CGA, however, you can read and write all the memory-controller registers. This is a handy feature if you are debugging programs, although it's not a good idea to rely on it if you are concerned about maintaining CGA compatibility.

The first 16 memory-controller registers are analogs of the registers on the Motorola 6845, the CRT controller chip used in the CGA. This means that CGA-compatible programs that access these registers directly can also run on the MCGA. Because the default horizontal and vertical CRT timing parameters used on the MCGA differ from those used on the CGA, you might want to write-protect the first seven registers so that CGA-compatible programs that attempt to update these registers do not inadvertently disrupt crucial CRT timing signals. Bit 7 of the mode-control register (register 10h) is the write-protect bit for the timing registers.

The remaining memory-controller registers control video mode selection and the alphanumeric character generator. These registers do not exist on the CGA. They support functions that are similar to what is available on the VGA: additional graphics modes and RAM-loadable alphanumeric character sets.

The video formatter supports three CGA-compatible control registers. The mode-control register (I/O port 3D8) controls video mode selection. The color-control register (port 3D9) controls palette and graphics mode background-color selection. The status register (port 3DA) is a read-only register whose contents indicate the status of the CRT's horizontal and vertical timing signals. All three of these registers are compatible with the analogous registers on the CGA.

In addition to the six video modes supported by the CGA, the MCGA offers a 640 by 480 two-color graphics mode (video BIOS mode 11H) and a 320 by 200 256-color graphics mode (BIOS mode 13H). You can set up both new modes, as well as all the CGA-compatible modes using `INT 10h` function 0 (see listing 1).

Table 1: *MCGA memory-controller registers. Registers 0 through 0F hexadecimal are comparable to those in the CGA's CRT controller.*

Register number	Function
0	Horizontal total
1	Horizontal displayed
2	Start horizontal sync
3	Sync pulse width
4	Vertical total
5	Vertical total adjust
6	Vertical displayed
7	Start vertical sync
8	(Reserved)
9	Scan lines per character
0A	Cursor start
0B	Cursor end
0C	Start address high
0D	Start address low
0E	Cursor location high
0F	Cursor location low
10	Mode control
11	Interrupt control
12	Character generator, sync polarity
13	Character-generator pointer
14	Character-generator count

Listing 1: *Video mode selection using the video BIOS.*

```
mov  ah,0                ;AH = INT 10h function #
mov  al,VideoModeNumber  ;AL= 11h (640x480 two-color)
                         ;    12h (640x480 16-color)
                         ;    13h (320x200 256-color)
int  10h  ;Call video BIOS
```

New Features

Two features of the MCGA are of special interest to programmers. One is that the vertical resolution of both alphanumeric and graphics modes is greater than on previous IBM video adapters. The other is that the MCGA can display up to 256 different colors at one time out of a possible 262,144 (256K) colors.

The vertical resolutions of the default BIOS video modes are listed in table 2. In alphanumeric modes, the vertical resolution is 400 scan lines—twice that of the CGA and better than the EGA's 350-line "enhanced" modes. Since the BIOS still displays 25 rows of characters in alphanumeric modes, the vertical size of each displayed character is 16 scan lines. These higher-resolution characters are sharp and easy to read.

When the MCGA emulates the CGA graphics modes (640 by 200 two-color and 320 by 200 four-color), it doubles the vertical size of pixels so that each is two scan lines high. Thus, although the CGA-compatible graphics modes use the same resolution in terms of pixels, the displayed resolution is still 400 lines, so these modes have a sharper appearance on the MCGA than they do on a CGA.

Another feature of both the MCGA and the VGA is expanded color display capability. This is provided by a digital-to-analog converter (DAC) that generates the analog RGB signals used to drive the PS/2 monochrome and color monitors. (The monochrome monitor responds only to the green color signal; the color monitors recognize all three.)

The video DAC uses a set of 256 eighteen-bit internal registers, each of which specifies an RGB combination. Each of the three primary colors is allotted 6 bits of each color register; the DAC converts each 6-bit value to a corresponding analog voltage level in the signals it outputs to the monitor. Thus, the video DAC can produce any of 64 color intensities for each of the three primary colors in a color register, thereby generating 256K (64^3) color combinations. Since there are 256 video DAC color registers, the video subsystem can display any 256 of the 256K color possibilities at one time.

Video BIOS

The video BIOS on the Model 30 provides the same set of functions as the motherboard ROM BIOS on the PC. The programming interface is the same: You access all video BIOS functions through interrupt 10h and pass parameters to the BIOS routines in the CPU's registers.

Also, several new `INT 10h` functions are available in the Model 30, as well as the other models in the PS/2 series (see table 3). IBM has expanded the `INT 10h` function 10h to provide access to the video DAC color registers. Function 12h has several new subfunctions that let you vary the default actions of other BIOS routines. For example, you can call `INT 10h` function 12h with the BL register set to 31h to enable or disable default palette loading when the video mode is changed.

`INT 10h` functions 1Ah and 1Bh are new to the PS/2 series. Your programs can call these `INT 10h` functions to determine the state of the video subsystem. A call to function 1Ah returns the video subsystem's display combination code, which indicates what type of monitor is in use. Function 1Bh returns a table whose contents describe the current state of the video BIOS: the current video mode, active video page, amount of video RAM available, and so on. These functions are useful in programs designed to run in more than one video mode, as well as in pop-up RAM-resident programs that must determine the current video state to produce appropriate video output.

Alphanumeric-Mode Programming

Despite the MCGA's improved resolution, programming in alphanumeric modes is virtually the same as on the CGA. The important difference is that the alphanumeric character generator on the MCGA can display user-defined characters. (The EGA, VGA, Hercules Graphics Card Plus, and Hercules InColor Card also have this capability.) The MCGA's alphanumeric character

continued

Table 2: *New BIOS video modes on the MCGA and VGA.*

Mode number		MCGA	VGA
0	40 by 25 16-color alphanumeric (320 by 400 resolution)	X	
0	40 by 25 16-color alphanumeric (360 by 400 resolution)		X
2	80 by 25 16-color alphanumeric (640 by 400 resolution)	X	
2	80 by 25 16-color alphanumeric (720 by 400 resolution)		X
11H	640 by 480 two-color graphics	X	X
12H	640 by 480 16-color graphics		X
13H	320 by 200 256-color graphics	X	X

Table 3: *New* `INT 10h` *functions on the MCGA and VGA.*

Function 10h: Color-palette interface

AL=3:	Toggle alphanumeric intensity/blink state
AL=7:	Read individual palette register (VGA only)
AL=8:	Read overscan (border color) register (VGA only)
AL=9:	Read all palette registers and overscan register (VGA only)
AL=10h:	Set individual video DAC color register
AL=12h:	Set block of video DAC color registers
AL=13h:	Select video DAC color page (VGA only)
AL=15h:	Read individual video DAC color register
AL=17h:	Read block of video DAC color registers
AL=1Ah:	Read video DAC color-page state (VGA only)
AL=1Bh:	Perform gray-scale summing

Function 11h: Character-generator interface

AL=4:	Load 8 by 16 alphanumeric characters
AL=14h:	Set alphanumeric mode using 8 by 16 characters
AL=24h:	Load 8 by 16 graphics characters

Function 12h: Alternate select

BL=30h:	Select vertical resolution for alphanumeric modes (VGA only)
BL=31h:	Enable/disable default palette loading
BL=32h:	Enable/disable video addressing
BL=33h:	Enable/disable default gray-scale summing
BL=34h:	Enable/disable alphanumeric cursor emulation (VGA only)
BL=35h:	Display-switch interface

Function 1Ah: Display combination code

AL=0:	Read display combination code
AL=1:	Write display combination code

Function 1Bh: Functionality/state information

Function 1Ch: Save/restore video state (VGA only)

AL=0:	Return state buffer size
AL=1:	Save video state
AL=2:	Restore video state

Listing 2: *Establishing an 80 by 50 alphanumeric mode on the MCGA.*

```
; load video BIOS 8x8 characters into alphanumeric character generator
        mov     ax,1102h                ; AH = INT 10h function number
                                        ; AL = 8x8 character-set load
        mov     bx,0                    ; BX = block to load
        int     10h                     ; load 8x8 characters into RAM
        mov     ax,1103h                ; AH = INT 10h function number
                                        ; AL = character-generator load
        mov     bx,0                    ; BX = blocks to load
        int     10h                     ; load 8x8 characters into
                                        ; character generator
; program CRT controller to display 8x8 characters
        mov     dx,3D4h                 ; DX = MCGA I/O port address
        mov     ax,309h                 ; AL = 9 (register number)
                                        ; AH = 3 (value for register)
        out     dx,ax                   ; update scan-lines register
        mov     al,0Ah                  ; AL = 0Ah (register number)
        out     dx,ax                   ; update cursor-start register
        mov     al,0Bh                  ; AL = 0Bh (register number)
        out     dx,ax                   ; update cursor-end register
; update status variables in video BIOS data segment
        mov     ax,40h
        mov     ds,ax                   ; DS -> video BIOS data segment
        mov     word ptr ds:[4Ch],80*50*2  ; update CRT_LEN
        mov     byte ptr ds:[84h],49       ; update ROWS
        mov     word ptr ds:[85h],8        ; update POINTS
```

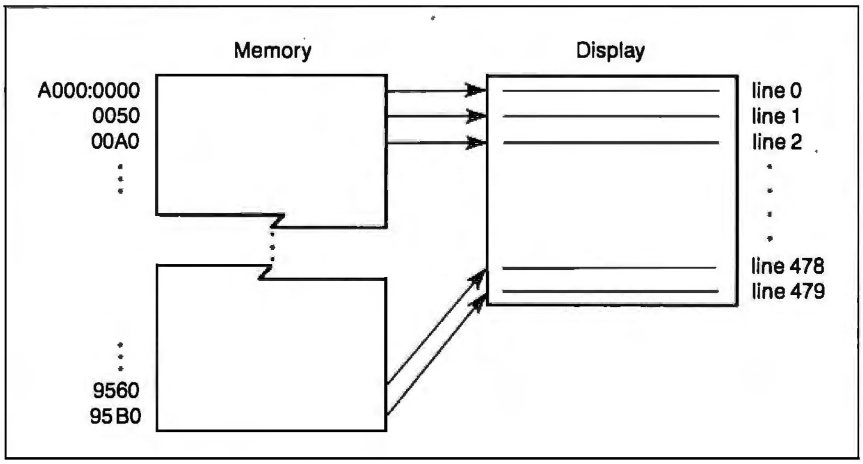


Figure 1: *A video-buffer map in 640 by 480 two-color graphics mode.*

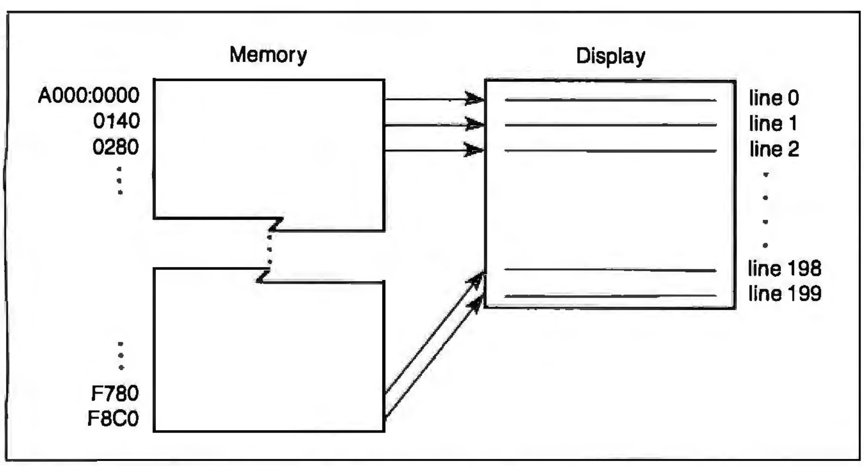


Figure 2: *A video-buffer map in 320 by 200 256-color graphics mode.*

generator can display characters from any of four different 256-character tables defined in video RAM.

To make the MCGA's character generator display one of these character sets, you first load the bit patterns that define the characters into video RAM. Then you program the character generator to copy the bit patterns from video RAM into one of its two internal character-definition tables. (These character-definition tables are called *font pages* in the IBM technical literature.)

It is easy to use the MCGA's RAM-loadable character sets on the MCGA to display characters that are smaller than the default 16-scan-line characters. In listing 2, the program calls the video BIOS to load the default graphics-mode character definitions—in which characters are only eight lines high—for use by the alphanumeric character generator. The code then reprograms the MCGA's memory controller to display only eight scan lines in each row of characters; the result is 50 rows of 80 characters each.

Apart from supporting RAM-loadable character sets, the MCGA replicates almost all the CGA's capabilities. However, the MCGA is not troubled by problems with display interference in alphanumeric modes. On the CGA, you must carefully synchronize CPU accesses to the video RAM with horizontal and vertical retrace intervals in the display refresh cycle. If you don't, you might see random patterns of interference or snow on the screen each time a program accesses the video buffer. The hardware design of the MCGA is such that this sort of display interference does not occur.

Surprisingly, the MCGA cannot generate a colored border. In alphanumeric modes on the CGA, you can display a border in any of 16 colors selected by programming the color-select register (I/O port 3D9). On the MCGA, you can still program the color-select register, but the MCGA does not display a border, regardless of the value you store in the register.

Graphics-Mode Programming

In CGA-compatible 640 by 200 two-color and 320 by 200 four-color modes, the MCGA emulates the CGA. The system maps pixels with a two-way interleave in the video buffer at B800:0000, just as they are mapped on the CGA. Video-buffer addressing is different, however, in 640 by 480 two-color and 320 by 200 256-color modes.

The video-buffer map in both 640 by 480 two-color and 320 by 200 256-color modes starts at A000:0000. Both modes map pixels linearly in the buffer from left

continued

Order Status, Technical & Other Info. (602) 246-2222

Call for programs not listed

WAREHOUSE DATA PRODUCTS

SPECIAL
MICROSOFT WORD 4.0
$195

TOLL-FREE ORDER LINE 1-800-421-3135

FREE SOFTWARE! FREE SOFTWARE! FREE SOFTWARE!

Purchase over $100 and receive one of these disks absolutely FREE! Purchases over $250 get two free disks, over $400 get three, or get all four disks when your purchase is over $500! **1) MIXED BAG** — A great assor ment of utilities and games all packed on one disk. **2) PC-WRITE** — Try this famous feature packed word processor. It's a winner! **3) FONT-SET** — Lets you set popular fonts like bold, underline, etc. on most late model printers from Citizen, Epson, NEC, Okidata, Panasonic, Star, Toshiba, etc. You can even use your printer like a typewriter! **4) ABC-LIST** — Great mailing list program! Sort on any field, do qualified searches, print reports and mailing labels, and more!

— SOFTWARE —

ACCOUNTING
Cyma ... Call
Dac Easy Acct. ... $39
Dac Easy Payroll ... 29
Dollars & $ense ... 94
In House Acct. ... 39
Managing Your Money 3.0 ... 117

COMMUNICATION PROGRAMS
Carbon Copy Plus ... 115
Crosstalk XVI ... 89
Crosstalk MK4 ... 110
Remote ... 89
Smartcom II ... 79

DATA BASE MANAGERS
Clipper ... 379
Condor 3 ... 325
DBase III Plus ... Call
DB-XL ... 82
Fox Base Plus ... 202
Genifer ... 194
Paradox 2.0 ... 398
PFS: Pro File ... Call
Powerbase ... 169
Q&A ... 207
Quickcode Plus ... 138
Quickreport ... 138
Quicksilver ... 295
Revelation ... 464
R Base System V ... Call
Reflex ... 81
Relate & Report ... 112
VP Info ... 48

DESKTOP PUBLISHING
Pagemaker ... 479
PFS: First Publisher ... 59
Ventura Publisher ... 489

GRAPHICS
Chartmaster ... Call
Diagram Master ... Call
Easy Cad ... 109
Energraphics 2.01 ... 294
Generic Cad ... 59
In-A-Vision ... 275
Microsoft Chart 3.0 ... 229
Newsroom Pro ... 65
Printshop ... 33
Prodesign 2 ... 148

INTEGRATED
Ability ... 56
Ability Plus ... Call
Enable ... Call
Framework II ... Call
Smart System ... 429
Symphony ... Call

LANGUAGES
Lattice C Compiler ... 242
Microsoft C Compiler ... 249
Microsoft Fortran ... 255
Microsoft Macro Assembler ... $84
Microsoft Pascal ... 166
Microsoft Quick Basic ... 55
Microsoft Quick C ... 55
Ryan McFarlan Fortran ... 305
Ryan McFarlan Cobol ... 479
Turbo Basic ... 55
Turbo C ... 55
Turbo Pascal ... 55
Turbo Prolog ... 55

MULTI-USER SOFTWARE
Fox Base ... 299
Word Perfect ... 310
Word Perfect Modules ... ea 75
Microsoft Word ... Call

PROJECT MANAGER
Microsoft Project ... 219
Super Project Plus ... Call
Timeline 2.0 ... 242
Total Havard Man. 2 ... Call

SPREADSHEET
Hal ... 115
Lotus 1-2-3 ... Call
Silk ... 149
Spreadsheet Auditor ... 82
Supercalc 4 ... Call
VP Planner ... 48

UTILITIES
Copy II PC ... 19
Copywrite ... 39
Cubit ... 30
Deskview 2.0 ... 72
Direct Access ... 49
Eureka ... 95
Fastback ... 85
Formtools ... 56
Graph in the Box ... 55
Mace ... 55
Microsoft Windows ... 55
Norton Advanced ... 75
Norton Utilities ... 48
PC Tools ... 19
Prokey 4.0 ... 70
Q DOS ... 49
Rightwriter ... 75
Sidekick ... 55
Sideways ... 39
Sqz ... Call
Superkey ... 55
Turbo Lightning ... 55
XTree ... 25

WORD PROCESSING
Microsoft Word 4.0 ... 195
Multimate Advantage II ... Call
Volkswriter 3 ... 139
Webster Spellcheck ... 37
Word Perfect ... 195
Word Perfect Executive ... $109
Word Perfect Library ... 59
Wordstar ... 162
Wordstar Pro ... 233
Wordstar 2000+ ... 206

— HARDWARE —

ACCESSORIES
Brooklyn Bridge ... 88
Copy II PC Bd. ... 75
Curtis Ruby ... 59
Mach III Joystick ... 36
Masterpiece ... 88
Masterpiece + ... 99
150 Watt Power Supply ... 69

BOARDS
AST
Advantage Premium ... 422
Sixpac Premium ... Call
Sixpac ... 145
HERCULES
Color Card ... 145
In Color Card ... 302
Graphics Plus ... 182
INTEL
Above PC 64K ... 225
Above 286 ... 319
Orchid Tiny Turbo ... 389
SUNTEK
IO XT ... 65
IO AT ... 65
TALLTREE
J Ram 3 ATP ... Call

COMPUTERS

AZ 386
80386-16 Micro Processor, 1 MB of Ram, Teac 1.2 MB disk drive, 220 watt power supply, 6 layer mother board, RT keyboard ... $2995

AZ TURBO AT
IBM AT Compatible, 512K, 6 & 8 MHZ, keyboard, graphics card, 220 watt power supply, MS/DOS 3.2 GW Basic ... $1025

AZ TURBO XT
135 watt power supply, One 360K drive, 640K, keyboard ... $519

AZ 10
150 watt power supply, dual 360K drive, 10 MHZ mother board, 640K, keyboard ... $670

AST 286 PREMIUM COMPUTER
512K, expandable to 2MB on the system board, RT enhanced style keyboard, parallel, serial and clock, 1.2 MB floppy disk drive, 7 expansion slots, two 32 bit fastram slots, DOS 3.1 & Basic 1 year warranty ... Call

EGA BOARDS
ATI Ega Wonders ... 209
Nec GBI ... Call
Paradise Auto480 ... 155
Quad EGA Plus ... 295
Vega Deluxe ... 269

EGA MONITORS
AMDEK 722 ... 455
Casper ... 410
NEC
Multisync ... 559
Multisync Plus ... Call

HARD CARDS
AZ 20 MB ... 425
AZ 30 MB ... 499
AZ 40 MB ... 625
Plus 20 MB ... Call
Plus 40 MB ... Call

HARD DRIVES
Seagate 20 MB ... 275
Seagate 30 MB ... 309
Seagate 30 MB AT ... Call

MODEMS
AZ 300/1200 ... 75
Everex 300/1200 ... 89
Hayes 1200 ... Call
Hayes 1200B ... Call
Hayes 2400 ... Call
U.S. Robotics 2400 ... 335

MICE
Logitech ... Call
Microsoft Bus ... 106
Microsoft Serial ... 119
Optimouse w/Dr Halo ... 96
Optimouse w/DPE ... 185

MONITORS
AMDEK
410 Amber ... $157
Magnovox RGB ... Call
PRINCETON
Max 12 ... 138
HX 12E ... 460
SAMSUNG
TTL Amber ... 72
Color w/tilt ... 249

PRINTERS
CITIZEN
MSP 10 ... 249
MSP 15 ... 315
MSP 20 ... 285
120 D ... 159
Premiere 35 ... 471
Tribute 224 ... 639
EPSON - Call on all models
NEC
P5XLP ... 840
P7 Parallel ... 619
8850 ... 1059
P6 Parallel ... 439
P960XL ... 1035
OKIDATA - Call on all models
PANASONIC
1080-I/M2 ... 172
1090-I/M2 ... 199
1092-I ... 310
1524 ... 572
1592 ... 392
1595 ... 439
3131 ... 259
3151 ... 407
STAR MICRONICS
NB24-10 ... 456
NB24-15 ... 610
ND10 ... 275
NP10 ... 139
NX10 ... 160
NX15 ... 306
TOSHIBA
321 SL ... 510
341 SL ... Call
351 Model II ... 910

RAM
64K 150NS ... 16.50
256K 150NS ... 33

TERMS: Shipping on most software is $5.00. AZ orders +6.7% sales tax. Personal check/company check - allow fourteen (14) days to clear. We accept purchase orders from authorized institutions for 3.5% more than cash price. All returns are subject to our approval. There will be a 20% restock fee. Minimum phone order $50. All prices are subject to change. Due to copyright laws we cannot take back any open software.

No Charge for Master Card or Visa

We do not guarantee compatibility

TOLL-FREE ORDER LINE 1-800-421-3135
WAREHOUSE DATA PRODUCTS
2701 West Glendale Ave. • Phoenix, AZ 85051

Phone Hours: Monday, Wednesday & Thursday 7 am-9 pm; Tuesday & Friday 7 am-5 pm; Saturday 9 am-5 pm. MST.

to right and from top to bottom on the screen. In 640 by 480 two-color mode, a bit represents one pixel, so there are eight pixels to each byte in the video buffer and 80 bytes in the buffer per row of pixels on the screen (see figure 1).

In 320 by 200 256-color mode, each pixel value comprises 8 bits, so the video buffer is mapped as 200 320-byte rows (see figure 2).

You can write routines that manipulate pixels in these MCGA graphics modes by modifying code that runs in CGA-compatible graphics modes. The routine in listing 3 is an example of code that updates a single pixel in 320 by 200 256-color mode. The program computes the video-buffer address by multiplying the number of pixels in each row by the pixel *y* coordinate and then adding the pixel *x* coordinate. Since each byte in the buffer represents one pixel, updating a pixel consists of a single machine instruction.

The MCGA's 640 by 480 two-color graphics mode deserves attention because its horizontal resolution and vertical resolution are the same in terms of the number of pixels displayed per inch. (Programmers sometimes describe this circumstance by saying that the pixels are "square.") This means that when you draw a figure in 640 by 480 graphics mode, you do not need to scale the figure to accommodate different horizontal and vertical resolutions.

Listing 3: *Setting the value of a pixel in 320 by 200 256-color mode.*

```
sp13    PROC    near
; call with: AX = y coordinate
;            BX = x coordinate
;            CL = pixel value
; compute address of pixel in video buffer
        mov     dx,320
        mul     dx
        add     bx,ax     ; BX = X + 320*Y
        mov     ax,0A000h
        mov     ds,ax     ; DS:BX -> pixel in video buffer
; update the pixel value
        mov     [bx],cl
        ret
sp13    ENDP
```

Listing 4: *Updating a video DAC color register.*

```
        mov     ah,10h          ;AH = INT 10h function #
        mov     bx,7            ;BX = 7 (register #)
        mov     dh,RedValue     ; DH,CH,CL = 6-bit RGB values
        mov     ch,GreenValue
        mov     cl,BlueValue
        int     10h             ;Call video BIOS
```

Video DAC Programming

Programming the MCGA's video DAC is straightforward when you use the video BIOS. In all modes except the 320 by 200 256-color graphics mode, you can use only the first 16 video DAC registers. The video BIOS loads these registers by default with a set of 16 CGA-compatible color values. You can, however, update any of these color registers using any of the 256K color combinations available.

For example, listing 4 shows how you could change the color value in video DAC register 7. The color-register value is actually 18 bits in size—red, green, and blue components are each 6 bits. The higher the value you specify for each component, the higher the displayed intensity of that color.

If a monochrome display is attached to the MCGA, the video BIOS performs a gray-scaling computation before it loads a color value into the specified video DAC color register. The video BIOS performs gray-scaling by taking a weighted average of the red, green, and blue values you specify. (The formula used is 30 percent red + 59 percent green + 11 percent blue.) The result is a gray-scale value that corresponds to the overall intensity of the specified color combination.

The VGA

The VGA subsystem takes its name from the video graphics array, a proprietary VLSI gate-array circuit that incorporates the functions of several EGA components: the CRT controller, the sequencer,

continued

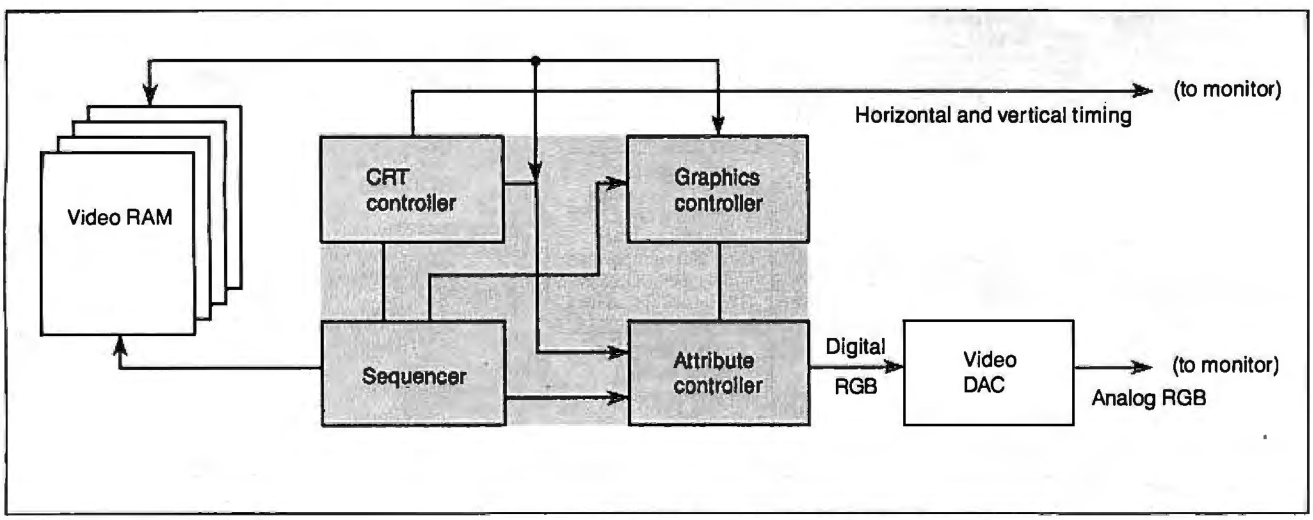


Figure 3: *The VGA subsystem. Components of the VGA chip are outlined in red.*

Developing a complex PC application with the wrong database has its pitfalls.

Any PC database manager can do simple applications and queries.

But when you want more complex applications, a limited product can be your downfall.

PC/FOCUS® is not just a database manager. It's also a true fourth-generation language with a complete set of developer's tools, including a screen painter, report writer, statistics, graphics, and micro-mini-mainframe communications. Programmer productivity gains are often ten to one, or more.

Release 3.0 A Window on the PC

And with Release 3.0, PC/FOCUS can be completely window-driven. You can use windows to program. Or create window-driven programs. Or both.

In its table-driven mode, PC/FOCUS is superb for quickly developing prototypes for large applications. Yet it also has the programming richness and power demanded by advanced users. And they won't run out of memory, because PC/FOCUS breaks the 640K barrier with extended memory support.

It's Better with Data

And PC/FOCUS delivers real access to real data. In fact, you can apply the powerful data analysis tools of PC/FOCUS directly against a dBASE file. By communicating with FOCUS in a mainframe, PC/FOCUS can also access data from every important mainframe database manager, including DB2 and IMS.

And Multi-user PC/FOCUS is available today for all the major LANs.

We give PC/FOCUS the same support as its mainframe counterpart: local help lines in 12 regional offices, a central hotline, and a national network of technical support and training centers.

Release 3.0

Some new features of PC/FOCUS:

- Developers' windows
- ModifyTalk (simplifies the programming of database updates)
- Scripting Language (generates communications macros)
- Full-screen asynchronous communication
- Protocol converters
- Completely menu-driven user interface
- 3-D graphics

Before you take another step, learn more about PC/FOCUS. Call 1-212-736-4433, Ext. 3700. Or write Information Builders, Inc., Dept. 07, 1250 Broadway, New York, NY 10001. We won't let you down.

PC/FOCUS
Information Builders, Inc.

Listing 5: *Establishing an 80 by 50 alphanumeric mode on the VGA.*

```
; establish 400-line resolution in alphanumeric mode
        mov    ax,1202h   ; AH = 12h (INT 10h function number)
                          ; AL = 2 (select 400-line modes)
        mov    bl,30h     ; BL = alphanumeric scan-lines select
        int    10h
        mov    ax,3       ; call video BIOS to set mode
        int    10h
; load video BIOS 8x8 characters into alphanumeric character generator
        mov    ax,1112h   ; AH = INT 10h function number
                          ; AL = 8x8 character-set load
        mov    bl,0       ; BL = block to load
        int    10h        ; load 8x8 characters into RAM
```

the graphics controller, and the attribute controller (see figure 3).

You program each component of the VGA as you would on the EGA. Each contains a number of control registers mapped to 8-bit ports. As with the MCGA, you access each register by writing a register number to an I/O port and then reading or writing the specified register. The CRT controller has 25 registers addressed at ports 3D4 and 3D5; the sequencer has 5 registers at ports 3C4 and 3C5; the graphics controller maps 9 registers to 3CE and 3CF; and the attribute controller has 21 registers, including 16 palette registers, mapped to 3C0 and 3C1.

Almost all of the many VGA control registers have the same function on the EGA, so if you are familiar with the EGA, you will be comfortable programming the VGA as well. The function of each of the registers is documented in IBM's technical reference manual for the PS/2 Models 50 and 60.

The VGA supports all the video modes available on the EGA, as well as the 640 by 480 two-color and 320 by 200 256-color graphics modes found on the MCGA. One additional graphics mode is unique to the VGA: a 640 by 480 16-color graphics mode (BIOS mode 12H) that is similar to the EGA-compatible 640 by 350 16-color mode, but with higher vertical resolution.

As on the MCGA, the default alphanumeric modes on the VGA have 400-line vertical resolution. Unlike the MCGA, however, you can set up the VGA's CRT controller to display alphanumeric characters with 200-line or 350-line resolution for compatibility with the CGA and the EGA.

Video BIOS

As on the MCGA, the VGA video BIOS provides support for all the CGA- and EGA-compatible `INT 10h` functions. The VGA BIOS also supports `INT 10h` functions 1Ah and 1Bh, which return information regarding the hardware configuration and video BIOS status as they do on the MCGA.

The VGA video BIOS also provides a video-state save/restore capability through `INT 10h` function 1Ch. This function can save and restore all control registers, video DAC registers, and video-related information from the BIOS data area in RAM, using a buffer provided at a user-specified address. The ability to save the state of the VGA and subsequently restore it lets a program switch between video modes or program the palette and video DAC registers freely without losing the context of a previously established video state.

Alphanumeric-Mode Programming

As on the MCGA, CGA-compatible alphanumeric-mode programming on the VGA is straightforward. Again, the video buffer is addressed starting at B800:0000 and mapped with alternating character codes and attributes. Video-BIOS support for character I/O is the same as it is on other IBM video subsystems.

As for the EGA and MCGA, you can configure the VGA to display user-defined alphanumeric character sets. You can also program the VGA's CRT controller to display characters of different vertical sizes, so that you can display more than the default 25 rows of alphanumeric characters. Listing 5 is a simple example of how you can call the video BIOS to set up an 80 by 50 alphanumeric mode using the 8 by 8 character definitions found in the BIOS ROM.

Graphics-Mode Programming

If you can program the EGA and MCGA in graphics modes, you can program the VGA. Routines that read and write pixels

continued

in CGA- and MCGA-compatible graphics modes also run on the VGA in these modes. In EGA-compatible graphics modes (320 by 200 16-color, 640 by 200 16-color, and 640 by 350 16-color), the same routines you use on an EGA should also run on a VGA. The only video mode unique to the VGA is 640 by 480 16-color mode, but it is almost identical to the EGA-compatible 640 by 350 16-color mode.

In these EGA-compatible modes, the video buffer is set up as a set of four parallel bit planes, each of which shares the same range of addresses starting at A000:0000. Data bytes are transferred to and from the bit planes in parallel whenever the CPU executes a read or write instruction.

This limited parallel processing is carried out by the VGA's graphics controller, which contains a set of four 8-bit internal registers or latches. Whenever the CPU executes an instruction that performs a read from an address in the video buffer, the graphics controller copies the contents of each of the parallel bit planes at the specified address into the latches. Thus, for example, when the CPU executes a `MOV reg,mem` instruction, the graphics controller copies 4 bytes of data from the bit planes into the latches.

The converse process occurs when a CPU executes a write instruction. In this case, the graphics controller combines the data byte written by the CPU with the contents of each of the latches and writes the result to the bit planes. Thus, the sequence of events in updating the video buffer in graphics modes is to execute a CPU read followed by a CPU write. This can be a sequence of two `MOV` instructions, as well as a single CPU instruction, such as `MOVS`.

Pixels are represented by the set of corresponding bits at the same address in each of the bit planes. Since there are four bit planes, a pixel can have any of 16 (2^4) different values, and the number of different colors you can display at one time is 16. You might think of the contents of the graphics controller latches as eight adjacent pixel values instead of 1 byte from each of the four bit planes.

As on the EGA, the key to graphics-mode programming on the VGA is to control the way the graphics controller manipulates the data bytes (pixel values) it reads from and writes to the bit planes. On the VGA, two graphics-controller read modes and four write modes affect what the graphics controller does during CPU reads and writes.

Graphics-Controller Read Modes

The two graphics-controller read modes are the same as those implemented on the

EGA. In read mode 0, the value of one in the four latches is copied to the CPU each time the latches are loaded by a CPU read operation. In read mode 1, the eight pixel values in the latches are compared to a reference value stored in the graphics controller's color-compare register. The graphics controller returns the result of the eight comparisons in a single byte to the CPU. Each bit of the byte contains a 1 bit where a latched pixel value matches the reference value.

Read mode 0 is useful for transferring data out of the bit planes into system RAM because you can access the contents of each bit plane separately. You can use read mode 1 for graphics operations, such as region fills, where you must scan the video buffer for pixels that match a predetermined value.

Graphics-Controller Write Modes

Each of the four graphics-controller write modes is also designed to simplify certain kinds of programming tasks. Write mode 0 is the one that the video BIOS routines use most frequently. In write mode 0, the graphics controller combines the eight latched pixel values with either the data byte written by the CPU or with a pixel value stored in the graphics-controller set/reset register. The graphics controller can AND, OR, or XOR pixel values, as well as replace them with CPU or set/reset data. You control this activity pixel by pixel by storing a bit mask in the graphics controller's bit-mask register; the bit mask indicates which of the eight latched pixel values is updated and which is left alone during the operation.

Consider what happens in listing 6, which uses write mode 0 to update the value of a pixel in 640 by 480 16-color mode. First, the routine computes the address of the pixel in the video buffer, as well as a bit-mask value for the bit-mask register. Then the graphics-controller registers are set up for the operation; write mode 0 is selected, the desired pixel value is stored in the set/reset register, the set/reset function is enabled for all four bit planes, and the bit-mask value is placed into the bit-mask register. Then the OR instruction updates the bit planes. Finally, the graphics-controller registers are updated with values that correspond to those used by default by the video BIOS, so that subsequent video BIOS routines run as expected.

Clearly, most of the work involves configuring the graphics controller; only one CPU instruction actually updates the pixel. Note the sequence of events that occurs during execution of the OR instruction: First, a CPU read occurs, so the latches are loaded with the eight pixel values at the specified address. Then, the CPU performs a logical OR of a register with the value it read from the graphics controller and performs a CPU write with the result.

The graphics controller ignores the byte written by the CPU because it is configured to use the pixel value in the set/reset register to update the latches. The bit-mask-register value specifies which of the eight latched pixel values is replaced with the set/reset value as the latched data is copied to the bit planes during the CPU write operation.

In graphics-controller write mode 1, the contents of the four latches are simply copied to the bit planes. Thus, write mode 1 is useful in filling the video buffer with a solid color or a pixel pattern.

In write mode 2, the pixels in the latches are updated with the pixel value specified in the CPU data byte instead of in the set/reset register. Consequently, you can use write mode 2 as easily as write mode 0 for updating the value of individual pixels in the buffer.

The VGA also supports a graphics-controller write mode 3. It is similar to write mode 0, except that its bit-mask value is derived by combining the data byte written by the CPU with the value in the bit-mask register using an AND operation. This lets you change the bit-mask pattern without programming the bit-mask register. However, because the EGA does not support write mode 3, you must avoid using it if you are designing a program to run on the EGA as well as the VGA.

Video DAC Programming

Using the video DAC is somewhat more complicated on the VGA than on the MCGA because the VGA's attribute controller plays a role in accessing the video DAC. The VGA does not restrict you to using only the first 16 video DAC color

continued

Listing 6: *Setting the value of a pixel in 640 by 480 16-color mode.*

```
sp12    PROC    near ; call with: AX = y coordinate
;                                 BX = x coordinate
;                                 CL = pixel value
; compute the pixel address in the video buffer
        push    cx          ; push pixel value
        mov     cx,bx
        and     cl,7
        mov     ch,10000000b
        shr     ch,cl       ; CH = bit mask for pixel
        mov     dx,80
        mul     dx          ; AX = Y*80
        mov     cl,3
        shr     bx,cl       ; BX = X/8
        add     bx,ax       ; BX = Y*80 + X/8
        mov     ax,0A000h
        mov     ds,ax       ; DS:BX -> pixel in video buffer
; set up the graphics controller
        mov     dx,3CEh
        mov     ax,0005
        out     dx,ax       ; set up write mode 0
        pop     ax          ; pop pixel value
        mov     ah,al
        mov     al,0
        out     dx,ax       ; set up set/reset register
        mov     ax,0F01h
        out     dx,ax       ; set up enable set/reset register
        mov     ah,ch
        mov     al,8
        out     dx,ax       ; set up bit-mask register
  ; update the pixel
        or      [bx],al     ; update latches during CPU read
                            ; update bit planes during CPU write
  ; restore default graphics-controller register values
        mov     ax,0000
        out     dx,ax       ; default set/reset value
        mov     ax,0001
        out     dx,ax       ; default enable set/reset value
        mov     ax,0FF08h
        out     dx,ax       ; default bit-mask value
        ret
sp12    ENDP
```

registers in alphanumeric modes and 16-color graphics modes. You can program the attribute controller to address the 256 video DAC registers in 16 register blocks.

On the VGA, each 4-bit attribute value (in alphanumeric modes) or 4-bit pixel value (in graphics modes) is processed by the attribute controller, which uses the value to select one of its 16 palette registers (see figure 4). Each of the palette registers contains a 6-bit value that combines with the value in the attribute controller's color-select register to form an 8-bit value; this 8-bit value is passed to the video DAC.

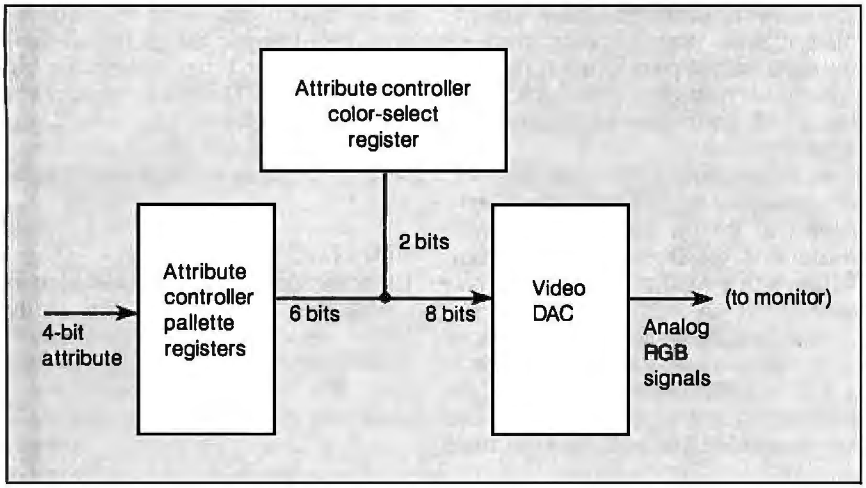


Figure 4: *VGA color control.*

Listing 7: *Using a block of video DAC color registers.*

```
Buffer  db  3*64 dup(?); buffer for color-register values
            .
            .
            .
; copy first 64 video DAC color registers to second 64 registers

        mov   ax,1017h     ; AH = 10h (video-BIOS function number)
                           ; AL = 17h (read block of color registers)
        mov   bx,0         ; BX = first register
        mov   cx,64        ; CX = number of registers to read
        mov   dx,seg Buffer
        mov   es,dx
        mov   dx,offset Buffer    ; ES:DX -> buffer
        int   10h          ; read registers into buffer
        mov   ax,1012h     ; AH = 10h
                           ; AL = 12h (set block of color registers)
        mov   bx,64        ; BX = first register to set
        mov   cx,64        ; CX = number of registers to set
        mov   dx,seg Buffer
        mov   es,dx
        mov   dx,offset Buffer    ; ES:DX -> buffer
        int   10h          ; set registers
; perform gray-scale summing
        mov   ax,101Bh     ; AH = 10h
                           ; AL = 1Bh (perform gray-scale summing)
        mov   bx,64        ; BX = first register to sum
        mov   cx,64        ; CX = number of registers
        int   10h
; use the gray-scale values
        mov   ax,1013h     ; AH = 10h
                           ; AL = 13h (select video DAC color page)
        mov   bx,101h      ; BL = 1 (select specified color page)
                           ; BH = 1 (color-page specifier)
        int   10h
; use the default color values again
        mov   ax,1013h
        mov   bx,001h      ; BH = 0 (color-page specifier)
        int   10h
```

The video DAC in turn uses the 8-bit value to select one of its 256 color registers, each of which contains an 18-bit RGB specification. In this way, a 4-bit attribute decodes into the set of three analog RGB values output by the video subsystem to the monitor.

Both the values in the palette registers and the value in the color-select register determine which video DAC color registers are referenced to generate color output.

By default, the video BIOS maintains EGA compatibility by initializing the attribute-controller palette registers with the same 6-bit values as on the EGA, as well as the first 64 video DAC registers with RGB values that produce the same 64 colors available on the EGA.

You could use the three remaining 64-register blocks of video DAC color registers by programming the attribute-controller color-select register. Video BIOS `INT 10h` function 10h supports this. (In IBM's technical documentation, blocks of video DAC color registers are referred to as *color pages*.)

In listing 7, I used the video BIOS to copy the contents of the first 64 video DAC registers into the second block of 64 registers. Then I called the BIOS gray-scaling function to replace the second block of color-register values with their gray-scale equivalents. At this point, I could call `INT 10h` function 10h again to select either the default color values in the first 64 color registers or the gray-scaled values in the second 64 color registers.

I Could Go On...

Although there are many more PS/2 video programming techniques than I can cover in the space of this article, it is easy to draw one conclusion from this brief overview: The MCGA and the VGA fall squarely into the mainstream of IBM video subsystems. Apart from its ability to display more colors and somewhat improved resolution, the MCGA strongly resembles the CGA in its capabilities and in the way you program it. Similarly, the VGA offers nearly complete compatibility with the EGA.

The VGA represents an incremental improvement over the EGA in terms of versatility, but it does not introduce any significant improvements in speed or resolution when you compare it with the EGA or with "enhanced" EGA clones. Nevertheless, it seems that many second-source vendors of video adapters for PCs and ATs regard the VGA as a new de facto hardware standard. ■

GET SERIOUS

...ABOUT ANALYZING YOUR DATA.

You might be spreading your spreadsheet a little too thin. Or maybe you're starting from scratch. But if you're serious about data analysis, you're ready for SPSS/PC+™—a full software family that brings you eight high-powered ways to complete any data analysis task.

Enter it. SPSS Data Entry II™ is a fully integrated data entry, cleaning and editing tool.

Analyze it. The SPSS/PC+ base package provides a powerful array of statistical and reporting procedures.

Picture it. SPSS/PC+ Graph-in-the-Box™ featuring New England Software's Graph-in-the-Box™ offers full color "snapshot" graphics.

Examine it. SPSS/PC+ Advanced Statistics™ lets you get more serious with your data.

Predict it. SPSS/PC+ Trends™—our latest option—is the complete time series analysis/forecasting tool.

Table it. SPSS/PC+ Tables™ produces presentation-ready tables instantly.

Chart it. SPSS/PC+ Graphics™ featuring Microsoft® Chart creates show-stopping graphs and charts.

Map it. SPSS/PC+ Mapping™ featuring MAP-MASTER™ creates maps where vast amounts of data can be summarized and presented in one, simple picture.

SPSS/PC+ products are being put to productive use by serious fact finders in business, government and education. For countless purposes such as market research. Wage and salary studies. Survey analysis. And quality control. Plus each product is superbly documented and supported by SPSS Inc., a leader in statistical software for nearly 20 years. While specially tailored customer support is available through the VALUE PLUS™ plan.

So if you're serious about data analysis, step up to SPSS/PC+. For details, contact our Marketing Department.

CALL 1/312/329-3315

SPSS inc.

SPSS Inc. • 444 North Michigan Avenue, Suite 3000 • Chicago, Illinois 60611
In Europe: SPSS Europe B.V. • P.O. Box 115 • 4200 AC Gorinchem, The Netherlands • Telephone: +31183036711 • TWX: 21019

For anyone who's eve
IBM presents persona

Now it's easier than ever to become a newsmaker in business, thanks to IBM's new desktop publishing system.

It's called the IBM® SolutionPac™ Personal Publishing System. And it comes complete with everything you need to start

The IBM SolutionPac for Personal Publishing lets you design and print near typeset quality documents.

making headlines: the hardware, software, service and support.

Just one purchase, and you'll be on your way to designing and printing the kinds of presentations, reports and newsletters bound to open people's eyes.

The Personal Publishing System is as easy to set up as it is to buy. All the components are designed to work together, from the Personal System/2™ Model 30 to IBM's laser printer, the Personal Pageprinter.

We've also preloaded all the necessary software—including Windows,™

Bit &

L. Pierce Riding Aca

dle-up

start when the
at 6:00 p.m.,
Madigan Hills
re will be a
experienced
ight vision)
ul world
e pierced
of
es of
ride over
moonlight
hts of
more
cook
d to
and
nk
ld west.

FOR
WBOYS

g 'uns. For
s, we'll be
the fun-
vseman.

THE COMPANY NEWSLETTER

All-Stars Clinch Division Title

Heavy hitting powers All-Stars to the top

Last night, the All-Stars won the game, the title and a chance to shuffle off to Buffalo next week for the championships.

The All-Stars powered themselves to the top with a 9-5 victory over the Grass Stains, who had been tied with the All-Stars for first place. The game was tight until the bottom of the fourth when the All-Stars blasted into the lead with two home runs to the upper deck in left field.

The game was highlighted by a most improbable play in which three All-Star players, each running at different speeds, ended up on third base at the same time. Fortunately, the third baseman missed the throw from centerfield, allowing two of the runners to score.

wanted to make headlines, publishing.

PageMaker® and PostScript®—on the system's hard disk. All you have to do is turn the system on, run through the tutorial program and begin composing your thoughts.

Because it's from IBM, you can transfer many of your IBM-compatible files directly into the Personal Publishing System—where you can then turn them into eye-catching documents.

There's good news for IBM PC XT™/286 and Personal Computer AT® owners, too. The IBM Personal Publishing Option offers you everything you

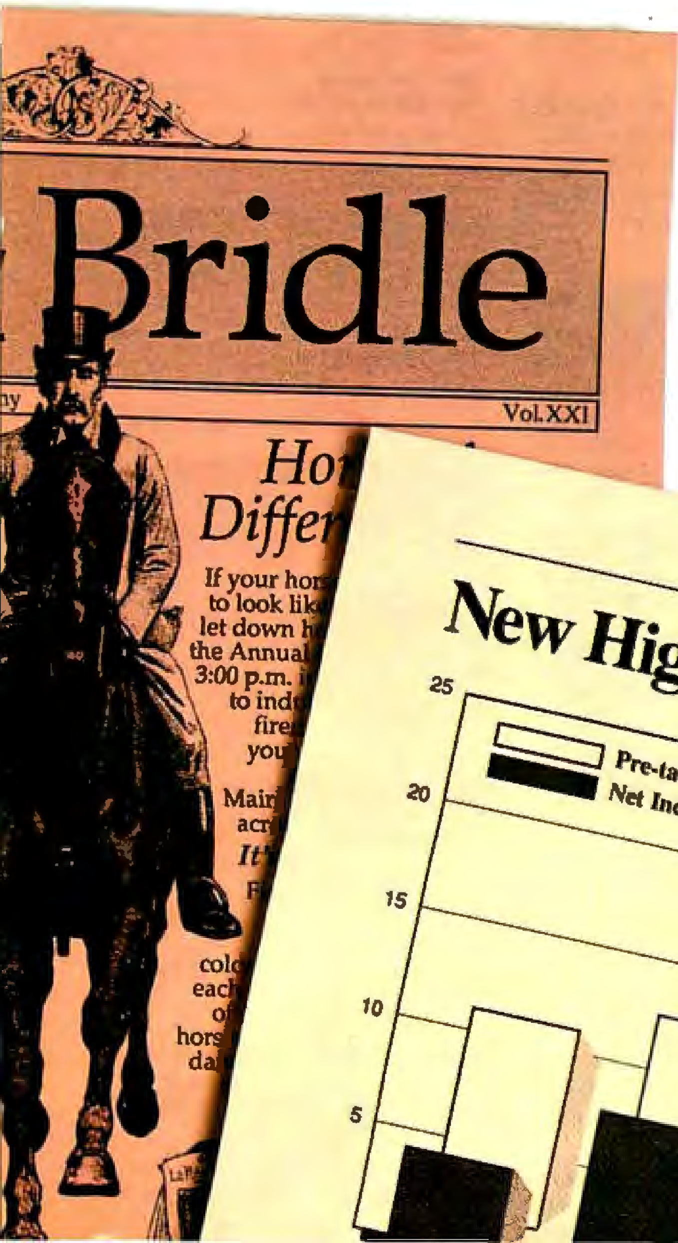
The IBM

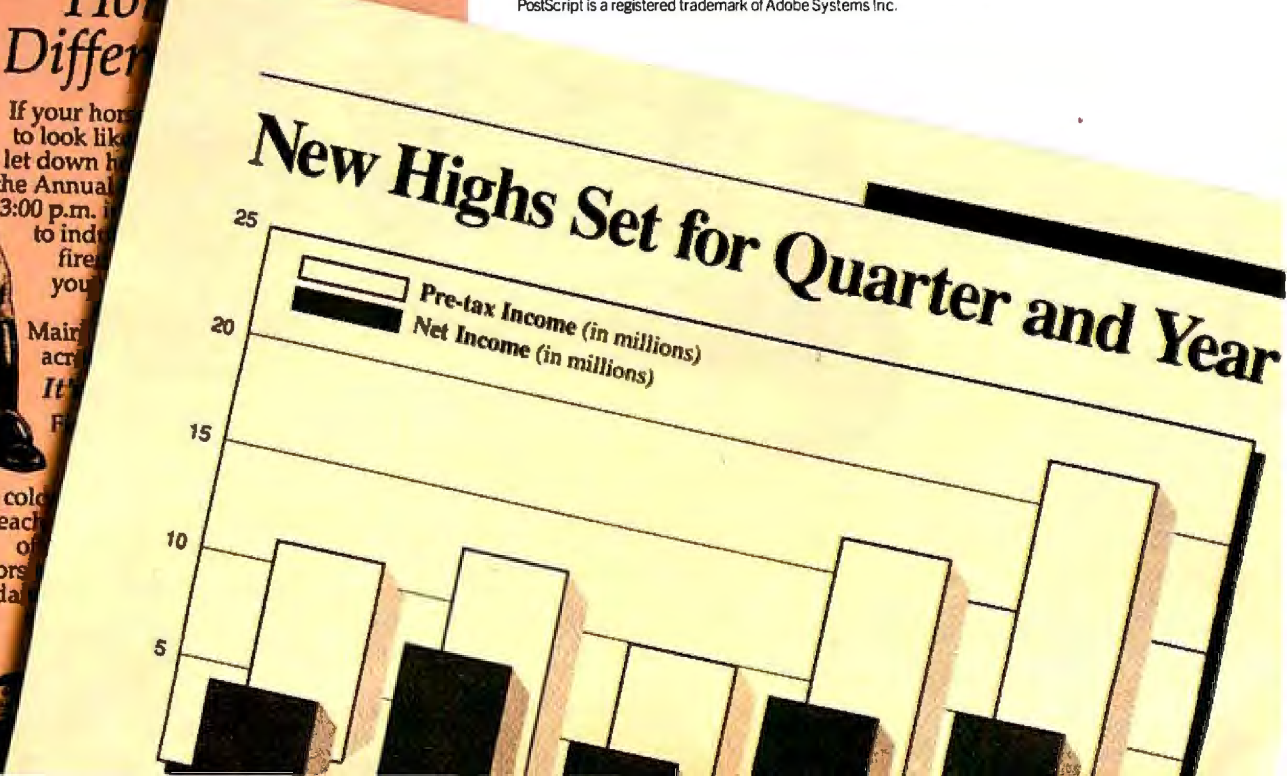
directly into the system.

into a cost-saving print shop.

To see that your newsworthy ideas are noteworthy, visit your IBM Authorized Advanced Products Dealer (ask about the IBM Credit Card), or call your IBM Marketing Representative.

For the certified dealer nearest you, call 1-800-447-4700, ext. 48. (In Canada, call 1-800-465-6600.) IBM®





Choose ALR's 16-MHz 386/2™ system for less than $2,000, or a blazing 20-MHz system for less than $2,500.

Now Advanced Logic Research announces second-generation 386 systems that do to the price barrier what our first generation did to the performance barrier. Annihilate it.

32-BIT $1990 MEMORY

Starting at $1990, ALR's totally new 386/2 systems couple the power of 32-bit processing with up to 2 MB of true 32-bit memory. What's more, unlike other companies, ALR 386/2 systems offer a choice of 16- or 20-MHz processor speeds, with sockets for 80287-10 and 80387 coprocessors. Which means your applications will run faster than on any other available PC. And ALR 386/2 systems let you use all the peripherals, graphics, enhancements, and applications developed for the most popular computer operating system in history.

How to run circles around the competition.

Sure, Compaq and IBM use the fastest hard disks and controllers with 1:1 interleaving, just like Advanced Logic Research. But instead of buffering

a full 17-sector track, they settle for single-sector buffering. Our way makes the fastest even faster where it counts—in the real world.

Naturally, all that speed coupled with EGA, enhanced EGA, or other high-performance graphics adapters makes the 386/2 ideal for CAD/CAM workstations and other graphics applications.

Plus, with Phoenix Control/386 software featuring 32-bit disk caching, Vdisk, and EMS/EEMS support, all enhanced 386/2 systems can run multiple applications without memory limitations.

Take your choice of a convenient desktop or space-saving floormount system configuration.

Speed to burn. Without having money to burn.

You won't find more power, flexibility, or quality at a more competitive price anywhere. And that reaffirms ALR's ability to deliver leading edge performance. At leading edge prices.

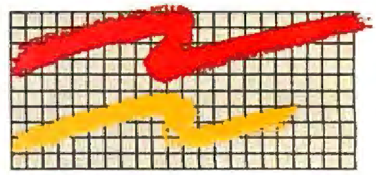
The ALR 386/2 Model 40 with EGA adapter is similar to the IBM Model 80-041 and Compaq Deskpro 386 Model 40 with EGA adapter. Except for a lower price and twice their standard RAM.

Of course, ALR also offers a full range of high-performance communications, memory, and storage enhancements.

Introducing ALR's 20-MHz 386/220,™ the fastest PC. Period.

If even more speed is what you need, you won't find a faster system than ALR's new 20-MHz 386/220. It scores a phenomenal 23.0 on Norton's SI benchmark, giving you nearly 25% more horsepower than Compaq's Deskpro 386. Yet it's priced many thousands less.

Take your pick of ALR's new 386/2 second-generation 386 systems: fast,



faster, or fastest. Call Advanced Logic Research today for the name of your local dealer. And discover the price of power isn't as high as it used to be.

Performance Specifications

386/2 Model 10 — $1990
386/220 Model 10 — $2485

- ALR-designed system board
- 16-MHz 80386 processor (20-MHz optional)
- Socketed for 80387 and 80287-10 support
- 1 MB 32-bit RAM, expandable to 2 MB on system board
- 1.2-MB floppy disk drive
- Serial and parallel ports
- Eight expansion slots
- 101-key keyboard

386/2 Model 40 — $3990
386/220 Model 40 — $4485

- ALR-designed system board
- 16-MHz 80386 processor (20-MHz optional)
- Socketed for 80387 and 80287-10 support
- 2 MB 32-bit RAM
- 40-MB, 30-ms or better average access time hard disk drive, 450-KB/s transfer rate
- EMS and EEMS software
- 1.2-MB floppy disk drive
- Serial and parallel ports
- Desktop or floormount
- Eight expansion slots
- 101-key keyboard
- 32-bit Vdisk and disk caching software

386/2 Model 80 — $4690
386/220 Model 80 — $5185

- ALR-designed system board
- 16-MHz 80386 processor (20-MHz optional)
- Socketed for 80387 and 80287-10 support
- 2 MB 32-bit RAM
- 70-MB, 30-ms or better average access time hard disk drive, 450-KB/s transfer rate
- EMS and EEMS software
- 1.2-MB floppy disk drive
- Serial and parallel ports
- Desktop or floormount
- Eight expansion slots
- 101-key keyboard
- 32-bit Vdisk and disk caching software

386/2 Model 130 — $7299
386/220 Model 130 — $7794

- ALR-designed system board
- 16-MHz 80386 processor (20-MHz optional)
- Socketed for 80387 and 80287-10 support
- 2 MB 32-bit RAM
- 130-MB, 30-ms or better average access time hard disk drive, 450-KB/s transfer rate
- EMS and EEMS software
- 1.2-MB floppy disk drive
- Serial and parallel ports
- Desktop or floormount
- Eight expansion slots
- 101-key keyboard
- 32-bit Vdisk and disk caching software

Enhancements

A complete range of enhancements, including 4 MB of 32-bit RAM, multifunction products, and additional I/O products are available.

See us at COMDEX Las Vegas Booth B910

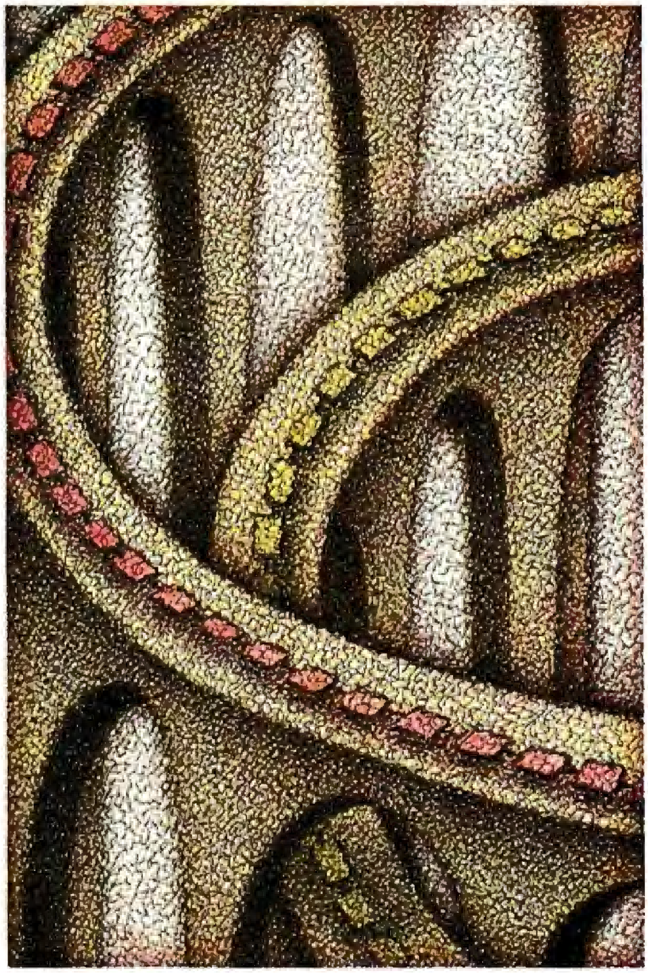
Advanced Logic Research, Inc.
10 Chrysler, Irvine, CA 92718 (714) 581-6770
FAX (714) 581-9240 Telex (510) 601-4525
Answer back Advanced Logic
From Asia or Europe, call Wearnes Technology, Phone (65) 259-2521 Telex RS38113 WRNTEC

Circle 6 on Reader Service Card (Dealers: 7)

Buses provide the base for the next-generation personal computer/workstation

Comparing IBM's Micro Channel and Apple's NuBus

Ciro Cornejo and Raymond Lee



The 32-bit bus has finally arrived for the personal computer in the form of Apple's Macintosh II and IBM's Personal System/2. Central to each of these machines is a 32-bit bus capable of high-speed operations at the bandwidth required for today's 16-megahertz processors.

As you might expect, though, Apple and IBM have adopted dissimilar bus architectures. NuBus, developed by MIT and Texas Instruments, has been adapted by Apple as the Macintosh II system bus. It supplements the Macintosh II's private 68020 processor bus, and six slots open the microcomputer for expansion. IBM, on the other hand, has revamped the older IBM PC and AT bus to handle higher speeds and 32-bit processing. The new IBM bus, the Micro Channel, serves as both a CPU bus and a system bus.

It's no accident that these new computers contain new bus architectures. Today's new microcomputers require more than just increased processor power and expanded memory. Investing in these products means committing to computing platforms that must be stable up to and perhaps through the mid-1990s. This requires a bus-based architecture capable of adapting to expanding processing rates, coprocessing or multiprocessing, and adapting to new peripherals such as advanced graphics terminals. I'll examine the two buses with regard to these capabilities.

Why a Bus?

Why have a special bus at all? Most processors actually define a bus structure of address and data paths called a *local*, or *CPU, bus*. The reason for this is straightforward: to build generality into systems.

A local bus is structured to optimize the processor-to-memory bandwidth. It is therefore highly processor-dependent: It is tightly linked to its processor, memory, and specific support peripherals. The cost of this performance is a loss of flexibility. A local bus might be unable to take advantage of newer technologies if they differ significantly from the local bus's design.

The existing IBM PC or AT buses are examples of an *expanded local bus*. An expanded local bus is a local bus with extensions that provide a set of generalized signals. These additional signals offer a general architecture that is easy to interface with. Since they use many of the processor's signals, expanded local buses are still processor-specific. For example, the IBM PC and AT buses are designed around the Intel 80x8x microprocessor architecture, and they have problems accommodating large memory expansions. The PC is limited to 1 megabyte of RAM (the 640K-byte limit is imposed by the layout of the PC BIOS), and the AT to 16 megabytes.

Unlike local buses, *system buses* are designed to maximize hardware subsystem-to-subsystem transfers. System buses offer a general protocol, or transfer method, for system CPUs or peripherals to interchange data. This is accomplished by treating the bus as a resource. To get control of the resource, a peripheral or processor must request its use formally, in competition with others. With this general approach, you can add peripherals, special functions, and even full computer subsystems easily

continued

Ciro Cornejo is an engineer with AST Research (2121 Alton Ave., Irvine, CA 92714). He was born in Chile, and his interests are nature, computers, math, and physics. Raymond Lee, a technical advisor at AST Research, is interested in computer architecture.

Illustration: Dave Ridley © 1987

to a system bus. The bus integrates hardware, cards, and subsystems into one smoothly running machine, much as an operating system integrates applications programs. The more general the integrating mechanism, the easier it is to add functionality and avoid obsolescence. Moreover, these buses are processor-independent. For example, NuBus defines a generalized address space that requires no processor-specific signals for peripheral or I/O accesses.

Overview of the Buses

The new IBM Micro Channel has evolved from the earlier PC and AT buses. Like them, it is a CPU or local bus once removed. As a local bus, it optimizes the host CPU-to-memory bandwidth, using a special transfer method termed "matched memory cycles," which I'll discuss later. It is also a system bus, in that it is treated as a system resource.

Taking an opposite tack, the Apple NuBus is a full system bus. It is independent of the Macintosh II's host processor; in fact, in the Mac II the motherboard is treated as a NuBus slot.

Both buses create a memory-mapped system. Each card or hardware entity is addressed within this bus address space. The 16-bit PS/2 systems, the Models 50 and 60, address a 24-bit space, or 16 megabytes; the PS/2 32-bit Model 80 and the Macintosh II NuBus address a full 32-bit space, or 4 gigabytes. Like its predecessors, the Micro Channel also has a 64K-byte I/O space.

Table 1: *A comparison of the two buses. Not all bus signals are included.*

	NuBus	Micro Channel
Utility signals		
Reset	RESET*	-CHRESET
Clock	CLOCK*	OSC
Interrupt	NMRQ* [Note 1]	-IRQ (3-7, 9-12, 14-15)
Audio	-----	AUDIO, AUDIO GND
Control signals		
Start bus cycle	START*	A0-A31, M/-IO, MADE24 [Note 2]
End bus cycle	ACK*	-CMD, or -MMC CMD
Cycle definition	TM0*, TM1*	M/-IO, -S0, -S1
Burst control	[Note 3]	-BURST, -TC
Address size	[Note 4]	-MADE 24
Data size	TM0*, TM1*, AD0*, AD1*	-BE0 through -BE3, TR32, -SBHE, -CD DS 16 [Note 1], -CD DS 32 [Note 1]
Matched memory cycle	-----	MMC, -MMCR
Signal returns		CHRDY RTN, -DS 16 RTN, -DS 32 RTN
Slot occupancy	[Note 5]	-CD SFDBK
Address/data		
Address	AD0* through AD31*	A0 through A31
Data	AD0* through AD31*	D0 through D31
Arbitration		
Request bus	RQST*	-PREEMPT
Arbitration lines	ARB0* through ARB3*	-ARB0 through -ARB3, ARB/-GNT
Slot ID	ID0* through ID3* [Note 6]	[Note 7]

Note 1: Separate line for each slot or card.
Note 2: For a memory refresh cycle, -REFRESH will also be used.
Note 3: Although a burst mode is defined in NuBus, it is not used in the Apple version of NuBus.
Note 4: All addresses on NuBus use 32 bits of address space.
Note 5: The declaration ROM must respond to a read at the top of the slot space.
Note 6: These lines are not bused.
Note 7: ID is stored on card but is not used for arbitration. A separate arbitration level is stored on the card when it is configured into the system.

Each bus entity can be defined as a *master* or a *slave*. A master entity can request and get control of the bus. A master must own the bus to send or receive data from another target entity on the bus, which can be another master or a slave unit. A bus slave unit cannot own the bus, but it can request service through an interrupt signal to one of the bus masters.

The masters contend for ownership of the bus resource via an arbitration protocol, which I'll describe later. Both buses allow multiple masters. However, only the Apple NuBus provides mechanisms for true multiprocessing: bus and resource locks. Bus locking allows a processor to lock a bus for exclusive access. With resource locking, a shared resource, such as RAM on a card with its own local processor, is locked so that the local processor can't access it. Both types of locks are necessary to prevent one processor from interfering with or corrupting memory that another processor is using.

While the IBM Micro Channel does permit multiple masters, there is not much to be gained in going to multiple host processors. This is because the Micro Channel is also an extension of the CPU bus. Processor memory operations tie up the Micro Channel, making the bus a bottleneck for the concurrent operation of two host processors. It should be noted that IBM, for efficiency, allows the host processor to access system motherboard memory without passing through the Micro Channel bus.

Also hindering multiprocessing on the Micro Channel is the absence of any direct provisions for bus or resource locking, although the 80386 in the IBM PS/2 Model 80 has hardware for bus locking. While not intended for true host-level multiprocessing, the Micro Channel does offer a general interface for drop-in coprocessing. The PS/2 host processors can be easily supplemented by powerful coprocessors, such as array and floating-point processors, or AI compute engines.

Timing the Critical Element

As a logic designer once said, "There are three important aspects of a digital design that must be carefully monitored: timing, timing, and timing." This is still true, especially for computer and bus designs.

Difficulties usually start when one block of logic has to talk to another block, especially if they each rely on different clock signals. This requires that the signals be synchronized to be passed from one logic block to another. A transmitting signal from a flip-flop strobed with one clock must be picked up and strobed into a receiving flip-flop using a second clock. The two clocks, transmitting and receiving, are asynchronous; no fixed relationship exists between them. Thus, it can take one receiving clock period to synch up to the transmitting data.

Buses, like logic, define synchronous or asynchronous interactions. In a synchronous bus, all interactions are defined in terms of a fixed bus clock or cycle. The bus clock edges define when data is valid and when to strobe it. Moreover, all transactions are in multiples of these bus cycles. The Apple NuBus is a synchronous bus.

Instead of relying on a fixed clock, an asynchronous bus is controlled by handshaking signals. A command signal is sent to a target adapter or card that responds with an acknowledge signal upon completion of a data transfer. All bus timing is dependent on the signals themselves. The IBM Micro Channel is an asynchronous bus, although it supports certain synchronous transfers.

Both the IBM Micro Channel and the Apple NuBus pass a common clock through the bus to minimize the synchronization problem among bus entities. However, there is a clock mismatch between the Macintosh II's local bus and NuBus, requiring synchronization before a transfer can occur.

The Macintosh II's 68020 runs with a 15.7-MHz clock, while NuBus runs with a 10-MHz clock. Synchronization delays between these bus clocks is minimized by using high-frequency clock signals. The NuBus 10-MHz clock is divided down from a 40-MHz crystal; the 68020 15.7-MHz clock is divided down from a 31.4-MHz crystal. The cost of clock synchronization is thus held to one clock period, either 25 or 31.5 nanoseconds. Clock synchronization is accomplished through the application-specific integrated circuit (ASIC)—the "GLU" custom gate array on the Mac II motherboard—and the NuBus timing control logic.

Synching up between the bus *processes* (i.e., bus reads or writes) also exacts a time penalty. The requesting bus must wait for the other bus to complete its current transaction cycle before it can attempt a transfer. All NuBus operations are defined with respect to its 10-MHz system clock. This clock has a 25 percent duty cycle: It is false (or high) for 75 ns and true (or low) for 25 ns.

Normally, a transfer from NuBus to the local bus takes a full 68020 instruction cycle (about 400 to 500 ns) to synch up. Going the other way, a Macintosh II request can take a typical NuBus transaction of 2 bus cycles (about 200 ns) to synch. It must be noted that this type of delay is not out of the ordinary; it is the time penalty paid by the communications protocol between the CPU bus and the system bus.

The IBM Micro Channel is an asynchronous bus, and all operations are gauged by the transmitted and returned signals. A common 14.3-MHz clock, OSC, is provided on the bus, eliminating the problem of signal synching. Moreover, a delayed signal will be picked up by the next clock, providing a built-in safety net for bus operations.

Micro Channel Timing

The IBM Micro Channel is an asynchronous bus. Handshake signals are used to initiate processes, signal availability of addresses and data, and completion of operations. It is the signal changes and the logic's response to them that drives this asynchronous bus.

The keys to this cycle are the -ADL and -CMD lines that define when the address is valid and when the data is valid. Their trailing edges can be used to strobe addresses and data as well. CD CHRDY is the mechanism for extending bus cycles. When this signal goes high, it triggers -CMD, which ends the bus cycle. See figure A for the sequence of a Micro Channel Basic Write cycle. Here is a short description of the sequence:

1. The cycle begins with the address and definition lines (-S0, -S1, M/-IO, MADE 24, TR32) defining the bus operation (read, write, memory, or I/O cycles), addressing mode (24-bit), and 32-bit transfers.
2. -ADL is asserted and defines a stable address.
3. The addressed card responds by asserting -CD 16, -CD 32, and -CD SFDBK. -CD 16 is asserted for 16- and 32-bit operations, -CD 32 for 32-bit operations. The card asserts its -CD SFDBK line to acknowledge being addressed. CD CHRDY is deasserted to extend the cycle, if necessary.
4. The data appears on the bus.
5. The -CMD line is asserted, indicating that valid data is on the bus. -ADL is deasserted.
6. If CD CHRDY was deasserted, the card drives CD CHRDY active after it has read the data.
7. -CMD is deasserted, ending the bus cycle.

Other masters can be contending for the bus ownership during the bus transaction. For more details on the Micro Channel, see the article "The 32-bit Micro Channel" by John Shiell on page 59.

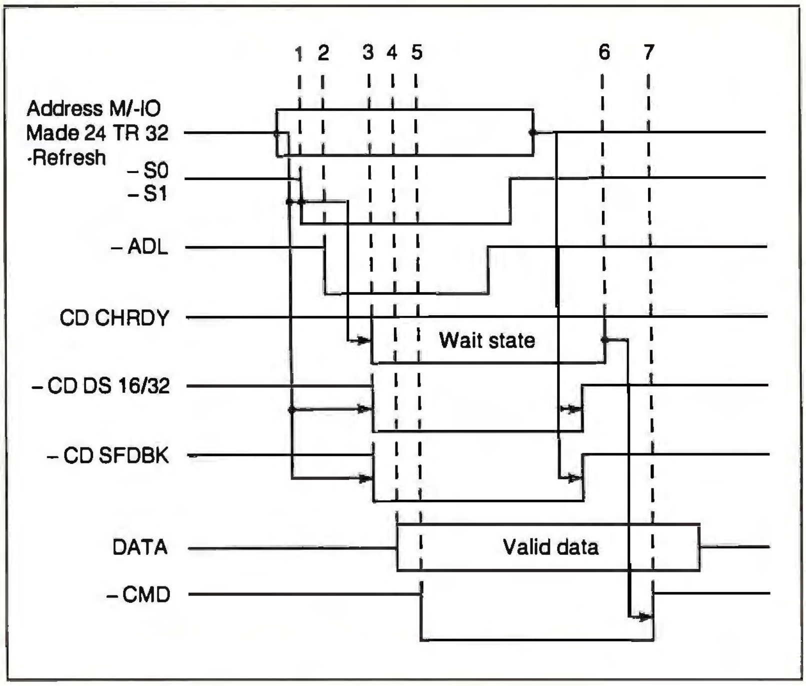


Figure A: *A Micro Channel Basic Write cycle.*

Bus to Bus

To distinguish between the two sets of bus signals, I'll stick to each bus's naming conventions. NuBus active low signals

continued

are labeled as signal_name*, while IBM uses its own convention for labeling an active low signal: -signal_name.

The NuBus is a simple and elegant bus that matches Apple's minimalist approach toward hardware. The NuBus has only 51 signals, including two parity signals not used by Apple. The IBM Micro Channel has 77 and 111 signals for the 16- and 32-bit versions, respectively. All Micro Channel signals are TTL-logic-compatible. Table 1 compares signals between the NuBus and the Micro Channel, and you can see a great deal of similarity between the two buses. The arbitration and utility signals almost match.

But there are differences. NuBus is multiplexed, sharing data and address on common lines, while the Micro Channel is nonmultiplexed, providing lines for both address and data. The IBM Micro Channel defines a number of discrete interrupts (-IRQ 3–7, 9–12, and 14–15) that can be shared among the boards. The Apple implementation, on the other hand, defines an interrupt (NMRQ*) per slot that is fed separately into the Macintosh II interrupt logic for processing.

The Micro Channel has a number of signals for coordinating asynchronous handshakes: The signals -ADL, -CMD, and -MMC CMD provide the basic bus handshake edges. Hardware signals are also used to delineate bus sizing (-BE0 through -BE3), 32-bit operation (-CD DS 32(n)), and 24-bit addressing (MADE 24). See the text box "Micro Channel Timing" on page 85 for more information on the bus cycles.

A special set of signals (-MMC, -MMCR, and -MMR CMD) is used in *matched memory cycles* to ensure fast CPU-to-memory accesses for the 80386. A matched memory cycle is started by the target slave returning an -MMCR request signal after being addressed by the system CPU. The 80386 responds by driving the faster -MMCR CMD handshake signal instead of the -CMD during a bus cycle. Matched memory cycles provide a bus read transaction in three clocks at 16 MHz, or 187.5 ns, while standard cycles using the -CMD handshake signal run four or more system clocks for a minimum of 250 ns. Matched memory cycles can be run with both 16- and 32-bit channel devices.

In contrast, the NuBus synchronous operations are relatively simple, requiring no special signals or exception processing. NuBus timing, however, is more stringent than the Micro Channel's, fitting sending and strobing of signals and data within 75 ns in the 100-ns clock cycle. See the text box "Apple NuBus Timing" below for more details on NuBus bus cycles.

NuBus defines a byte/word structure that matches the Intel 80x8x addressing schemes (byte order 0, 1, 2, 3), not the Macintosh's 68020 scheme (byte order 3, 2, 1, 0). The bus transceivers are wired to map the data from NuBus order into the Macintosh byte order. Bus sizing is handled automatically; the bus handles byte (8 bits), half-word (16 bits) and word (32-bits) sizes.

The NuBus specification defines a block, or burst mode, that can move up to sixteen 32-bit words in a transaction, but Apple has not implemented it in the Mac II NuBus design. IBM, however, has implemented a burst mode in the Micro Channel in conjunction with direct memory access. This DMA burst capability allows large blocks of data to be moved while minimizing bus overhead. In fact, each peripheral on the channel can be viewed as a DMA channel.

When accessed by the DMA controller, a card can assert -BURST, guaranteeing bus ownership for block transfers. Thereafter, data is transferred using only the -CMD signal to define data valid for both the read and write stages. The block transfer ends when the card deasserts the -BURST line for the last cycle. For predefined transfers, the DMA controller

continued

Apple NuBus Timing

The Apple NuBus is a synchronous bus; all operations are defined with respect to its basic clock cycle. The clock runs at 10 MHz, with a 100-ns period and a 25 percent duty cycle. Two edges of the clock serve the bus. The rising edge at the start of the period is the driving edge, strobing signals and address onto the bus, and the falling edge, 75 ns later, is the sampling edge for taking information off the bus.

Bus transactions are made up of bus cycles or clock periods. A transaction can be a single cycle or multiple cycles, especially if a slower peripheral is involved. Delays are added by inserting additional bus cycles. The timing diagram in figure B shows the basic write transaction, which consists of a START cycle, any intervening bus cycles, and an ACK cycle. Here is the sequence:

1. START* is asserted, indicating the start of a bus transaction. The master places addresses on the AD31* through AD0* lines; and the TM0*, TM1* lines define the type of transaction.
2. All cards read the addresses. The slave is identified by the address.
3. The master drives the data onto the AD31* through AD0* lines.
4. The slave reads the data off the bus.
5. The slave asserts ACK* to signal the end of the transaction and places the appropriate status codes onto TM0* and TM1*.
6. The master releases the AD31* through AD0* lines, and the slave releases the ACK* and status lines.

Other masters can be competing for the bus during the bus transaction.

For more information on NuBus, see "The Apple Macintosh II" by Gregg Williams and Tom Thompson in the April BYTE.

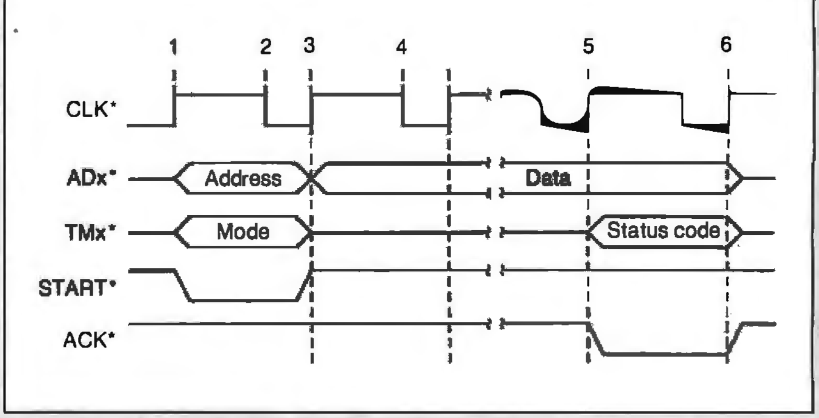


Figure B: *A Nubus write cycle.*

marks the last cycle by asserting the terminal count line (-TC).

A DMA controller can transfer 64K bytes of data between a peripheral and memory, the same as in an IBM PC. The PS/2 DMA controller can handle 24-bit read and write addresses, unlike the PC's 20-bit address limit. Unfortunately, this DMA capability is limited to transfers of 8- or 16-bit data.

Bus Address Space

Both the NuBus and the Micro Channel map bus addresses into a full bus address space that includes system memory and ROM, setup ROM, and device buffer space. Analogous to a CPU bus, these buses provide access to locations in that space.

The IBM implementation maps into a 16-megabyte or a 4-gigabyte address space. The bus address space is the same as the CPU address space. In this respect, the Micro Channel acts as a local CPU bus. The system board RAM, either 512K bytes or 640K bytes, starts at 00000 hexadecimal. The 128K-byte video RAM and channel ROM are mapped into the lower address pages. Topping off the memory space at E0000h through FFFFFh is the 128K bytes of system board ROM or RAM, depending upon how the computer's resources have been allocated. RAM memory mappings above address FFFFFh are managed in 1-megabyte chunks. See figure 1 for a memory map of an IBM PS/2 Model 80. Bits in a memory-encoding register and a split-address register determine how and where memory will be allocated.

Bus memory space for Apple NuBus implementation doesn't match the Macintosh II's 68020 processor address space. The upper one-sixteenth, or 256 megabytes, of the NuBus 4-gigabyte address space is called the *slot space.* This slot space is divided into 16 sections, one for each NuBus slot, and each slot owns 16 megabytes of the space. The top of each slot address space is reserved for a slot-declaration ROM that is accessed at that address. The slot a card occupies on NuBus determines its slot identification, which in turn determines its arbitration level and its location in the slot address space.

NuBus defines 16 slots, but the Macintosh II provides six. The six slots have IDs of 9h through Eh. Slot 0 is the Mac II motherboard, and slot F (which does not have a physical slot) is reserved. One slot becomes the video buffer for the machine, depending upon which slot the video card is placed in. Slots 1 through 8 are unused, because no room exists in the 24-bit address space for them. For this reason, the existing slots are limited to 1 megabyte of slot space instead of 16 megabytes.

Apple's implementation of NuBus allows a slot to own a "superslot" space of 256 megabytes, as well as its 16-megabyte slot space at the top of NuBus memory. We won't discuss superslots further, since they aren't accessible by the Mac II, although you should note that other cards on NuBus could use these areas. See figure 2 for a detailed look at the Macintosh II memory map and its arrangement in the NuBus address space.

The 24-bit address space for the Macintosh II starts at 0h with 8 megabytes of RAM, followed by 1 megabyte of ROM, then 6 megabytes of slot space, and topped by a 1-megabyte region of memory-mapped I/O devices. The Mac II's 24-bit address space is mapped into the 32-bit NuBus address space by placing the RAM, ROM, and I/O areas at the bottom of the NuBus address space. However, from the NuBus side, the Mac II's ROM appears at addresses F0800000h to F0FFFFFFh, and the I/O area maps to F0000000h through F07FFFFFh.

Under this scheme, the maximum RAM that can be accessed on the local bus is 8 megabytes, using 1-megabyte single in-line memory modules (SIMMs). The Mac II's motherboard RAM can be expanded to 128 megabytes if and when higher-density SIMMs are available. However, you can add more RAM to the system through the NuBus slots, and vendors are now supplying NuBus memory cards.

The Macintosh II is currently restricted to 24-bit addressing or 16 megabytes when running with the current operating system. An Apple Unix implementation (A/UX) is in the works that will handle 32-bit addressing and requires a memory-management unit for virtual-memory processing.

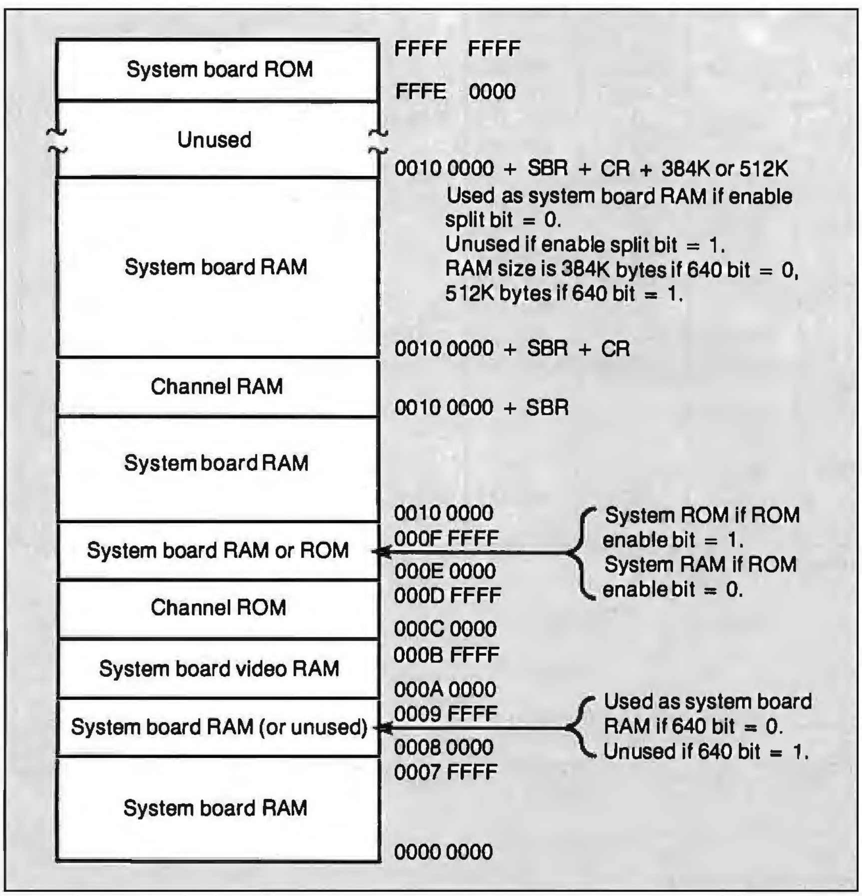


Figure 1: *The PS/2 Model 80 memory map. The memory arrangement is determined by the contents of the memory-encoding and split-address registers. SBR is system board RAM; CR is channel RAM. System board RAM and channel RAM are allocated in 1-megabyte chunks above address FFFFFh, with the exception of the split-system RAM. The system ROM at addresses E0000 through FFFFF is a copy of the system ROM at addresses FFFE0000 through FFFFFFFF.*

Bus Ownership

Both buses use arbitration to allocate ownership of the bus to a single master when several masters request use of the

continued

Why Does The Genius® Display Work So Well?

The Genius® Full-page Display:
66 lines in very high resolution - 100 dots/inch

Because All These Work So Well With The Genius.®

THE GENIUS Display System enables you to work with more programs in a full page format than any other monitor.

Desktop publishing, word processing and spread sheets are just a few of the applications that work better with THE GENIUS.

THE GENIUS Display System is a 15" high resolution monitor and adapter card.

With over 7,500 in use, THE GENIUS is available today to add a new dimension to office productivity.

See us at Booth #4316
COMDEX/Fall '87
November 2–6, 1987
Las Vegas Convention Center
Las Vegas, Nevada

Micro Display Systems, Inc.

1310 Vermillion St., P.O. Box 455, Hastings, MN 55033
Telex: 4310015 Fax: (612) 437-7325 Phone: (612) 437-2233
Toll Free: 800-328-9524

The Genius® Full-Page Display

bus. Arbitration typically takes place concurrently with bus transactions on both buses, but the Micro Channel allows a system configuration that restricts arbitration to nonconcurrent operation.

NuBus arbitrations take two full bus cycles, or 200 ns, to select the next bus owner. On the Micro Channel, arbitrations typically take 300 ns.

Each bus uses distributed arbitration to select the next bus owner; that is, logic on each card outputs the arbitration level on four arbitration lines (either ARB0* through ARB3*, or -ARB0 through -ARB3) and determines the winner of each arbitration contest based on the signals on these lines. The arbitration level is determined in NuBus by the card's slot ID, with 0 being the lowest priority and Fh being the highest. For the Micro Channel, the arbitration level is stored on the card when it is configured into the system. The highest priority a card can have is level 0, and the lowest is Fh. See table 2 for a comparison of the arbitration levels. The Micro Channel also has a Central Arbitration Control Point, which is some logic on the PS/2 motherboard, that controls the start and winner of an arbitration contest.

To compete for ownership, the master asserts its request line (RQST* for NuBus, -PREEMPT for Micro Channel). For the Micro Channel, the Central Arbitration Control Point drives the ARB/-GNT line to the arbitrate state, allowing the arbitration contest to begin. Each master then places its arbitration level onto the 4-bit arbitration bus. If a competing master has output a higher level, the master will cease to compete for ownership for the next bus transaction. It will, however, hold its asserted request line to compete for the following bus transaction. On NuBus, at this point, the winner of the contest owns the bus. On the Micro Channel, the Central Arbitration Control Point lowers the ARB/-GNT line to the -GNT state, allowing the winner to own the bus.

Both buses ensure fairness by preventing a higher-priority-level card or channel from continuously withholding ownership of the bus from lower-priority-level entities. Card or channel logic prevents the card just serviced from requesting bus ownership until all pending requests are honored. In a sense, there are no arbitration priority levels for NuBus cards, since the NuBus strictly enforces fair bus access. However, for special cases, a channel can be configured on the Micro Channel without fairness to ensure continued ownership of the bus.

The NuBus has explicit mechanisms

continued

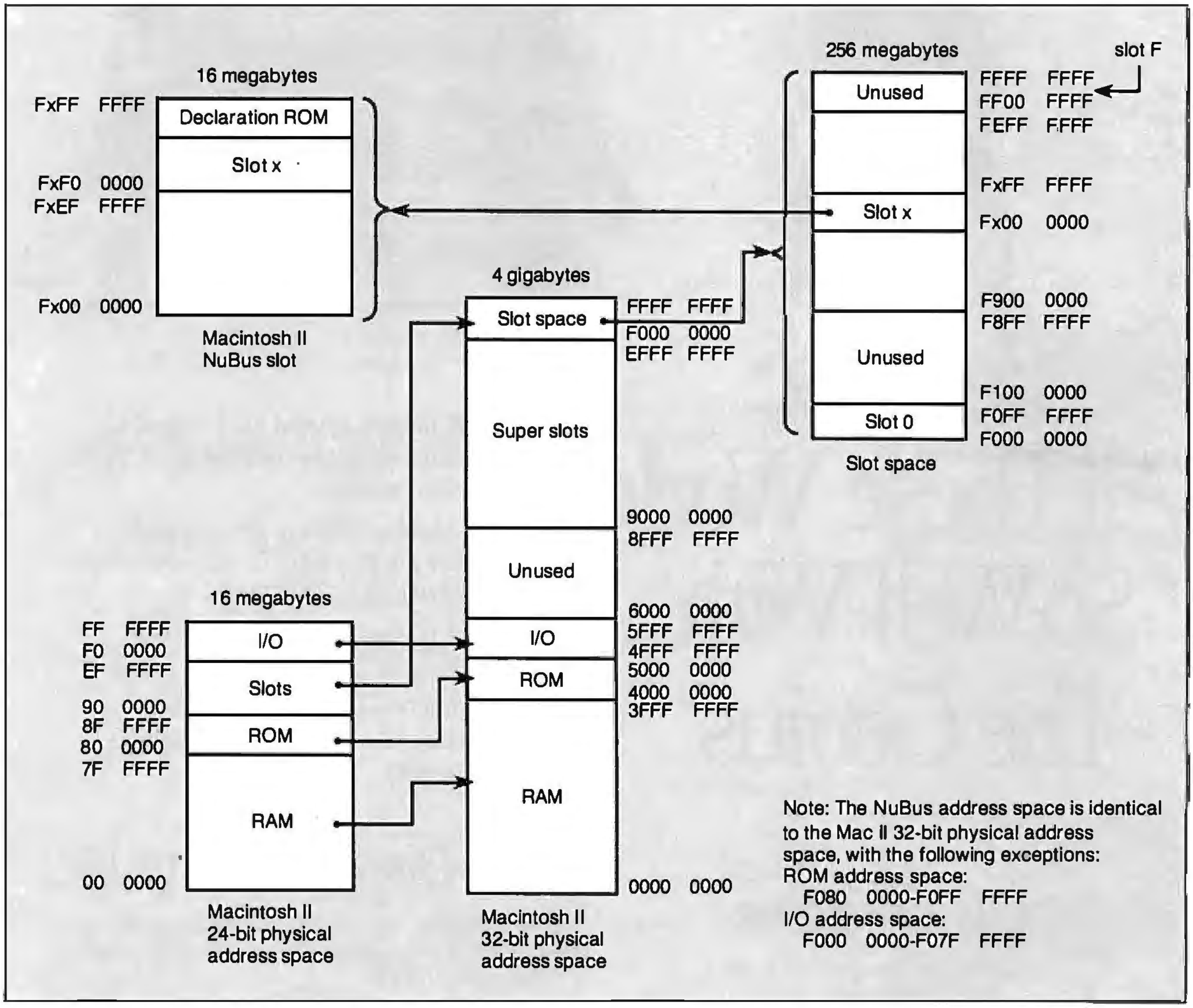


Figure 2: *The Macintosh II memory map. The memory in the superslot space and 15 megabytes of the slot space are not available to the Macintosh II, but NuBus cards can access these regions.*

for continued bus and resource ownership. Using an attention cycle (START* and ACK* both asserted), a master can request continuing bus ownership. It can also request a resource lock. A resource such as a memory card can be locked, denying access to any other master.

Both locks are extremely useful for multiprocessing; they allow a processor to do an uninterrupted test and set, as well as control access to a critical resource. For example, the Macintosh II motherboard uses bus locking to lock out the NuBus for critical local processing, including disk transfers and interrupt processing.

Card Configuration

Both the IBM Micro Channel and the Apple NuBus define high-level mechanisms to integrate cards or devices into the bus system. This eliminates the need for jumpers or switches to set either a card's interrupt level or its address space, which is the cause of a lot of bus problems on typical microcomputer systems.

The Micro Channel's Programmable Option Select (POS) eliminates switches from the system board and adapters by replacing them with programmable registers. Automatic configuration routines store the POS data into a battery-powered CMOS memory for system configuration and operations. The configuration utilities rely on adapter description files that contain the configuration data for a card. Configuration files define system operation, including system memory maps, video-processing options, and the individual adapter configurations.

At boot-up, the PS/2 Model 80 first validates the contents of the POS memory by examining a check character stored there. If the memory passes this test, the system then selects a card using the -CD SETUP lines. The card responds with its ID number. The system then loads the appropriate configuration data from CMOS memory into the card, as determined by the card's ID. This data sets the card's arbitration level and fairness, the address range of the card's I/O ROM, and the I/O address range. Cards that fail to configure properly are disabled by the system.

The Macintosh II relies on a slot manager to configure and maintain NuBus cards. Each card is required to have a special declaration ROM that holds the card-specific configuration information. Information in the declaration ROM includes byte lanes (which bytes of the NuBus data path are used), a test pattern, a revision level, a ROM cyclic redundancy check for validating the contents of the declaration ROM, and a resource directory. The resource directory points to various resource lists, such as the device icon, the device boot record, and the driver directory, which in turn points to blocks of code for the driver. The slot manager reads the declaration code at boot-up to configure the card into the system and installs any drivers or interrupt routines into system memory. The slot manager can also recognize a card as a bootable device and transfer control to the card when the system starts up. A card that fails to configure properly will be ignored, or a system error is posted.

Table 2: *The priority levels for the two buses and their device assignments. The priority levels are programmed into Micro Channel cards when they are configured into the system; NuBus priorities depend upon the slot the card is in.*

Arbitration level (Micro Channel)	Micro Channel Value	Micro Channel Device assignment	Apple NuBus Value	Apple NuBus Device assignment	Arbitration level (NuBus)
Highest	−2	Memory refresh	–	–	Lowest
	−1	NMI	–	–	
	0	DMA channel 0	0	Motherboard	
	1	DMA channel 1	1	No slot	
	2	DMA channel 2	2	No slot	
	3	DMA channel 3	3	No slot	
	4	DMA channel 4	4	No slot	
	5	DMA channel 5	5	No slot	
	6	DMA channel 6	6	No slot	
	7	DMA channel 7	7	No slot	
	8	Reserved	8	No slot	
	9	Reserved	9	Slot 9	
	A	Reserved	A	Slot A	
	B	Reserved	B	Slot B	
	C	Reserved	C	Slot C	
	D	Reserved	D	Slot D	
	E	Reserved	E	Slot E	
Lowest	F	System CPU	F	Reserved	Highest

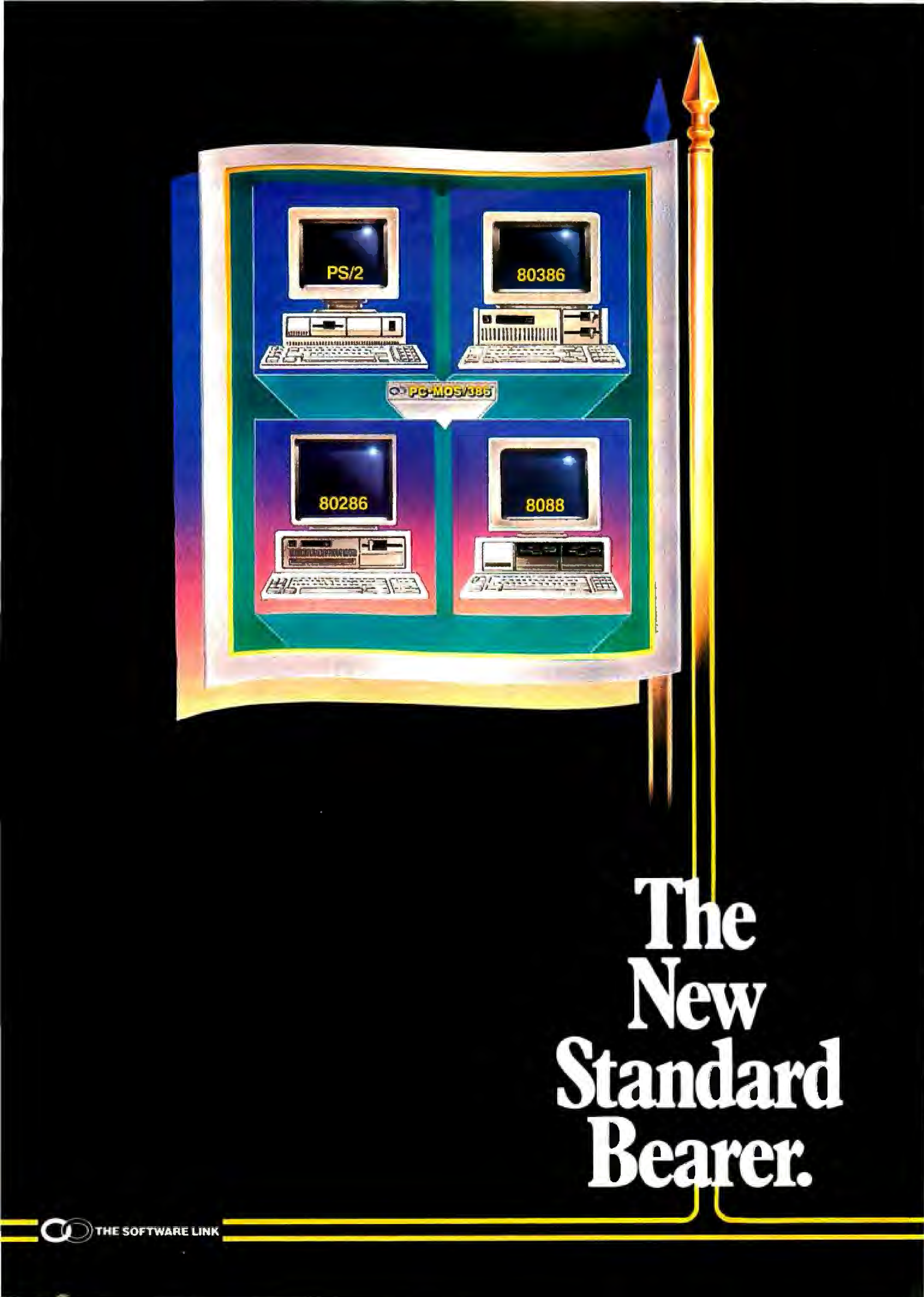
A Future with a Past

As you can see, both buses break new ground to optimize bus performance and minimize the user's effort to add a new card to the system. However, these buses must also deal with their past: providing compatibility with the existing market of software and hardware.

IBM faced the dilemma of maintaining compatibility with existing AT bus cards and limiting bus throughput to about 8 MHz, or redesigning the bus to optimize throughput at the expense of hardware compatibility. Looking toward a future of higher-speed processors and computing needs that require the handling of vast amounts of data, IBM chose to redesign the bus. However, the Micro Channel is, in a sense, still a CPU bus; throughput is optimized, since few bus clocks are lost synchronizing dissimilar components in the system. Its asynchronous nature allows future cards, operating at those higher speeds, to be installed with little to no change to the PS/2 system, while bus operations on NuBus are bound to its 10-MHz clock.

However, since the Micro Channel is a CPU bus, it's difficult to allow for multiple processors on the bus without interfering with the 80386's operation. NuBus, being a system bus, readily allows other processors to operate on it. Cards on NuBus can communicate and share data with one another without interfering with operations on the Mac II's local bus. In fact, AST Research offers a NuBus card that is essentially an IBM PC AT that runs independently in the Macintosh II but can share data with the 68020 CPU when necessary. Finally, the slot manager in the Mac II allows a NuBus card to be a boot device. You could drop a NuBus card with the next-generation CPU into a Mac II and let it take control of the machine—the ultimate in hardware expandability.

Both machines still have some of their past built into them. A look at the memory maps shows that both systems were designed to be compatible with their current operating systems, while providing a gateway to the next generation of software. The Macintosh II is the first machine in the Macintosh line to have slots, so Apple at least did not have to confront the problem of bus compatibility. But there's a certain irony in the fact that Apple must migrate from a 24-bit to a 32-bit operating system, similar to what IBM faces in the move to OS/2. ■


PS/2
80386
PC-MOS/386
80286
8088
The New Standard Bearer.
THE SOFTWARE LINK

# of Users Will Choose PC-MOS/386.™

6. Concurrently Supports Virtual 8086 and 80386 32-Bit Mode

80386-based PCs & PS/2s are dual-personality computers. To run DOS programs, they act as PCs with a 640K memory limit. But to take advantage of their minicomputer capacity, they operate in true 80386 mode which lets them address up to four gigabytes of memory. PC-MOS enables the 80386-host and its workstations to independently switch between these modes–making DOS compatibility and 80386 power simultaneously possible.

7. Multi-Tasking

While it's true you could look elsewhere for multi-tasking, why would you want to? The *other* multi-tasking operatinc system is not now, nor is it planned to be, multi-user. It won't even run multiple DC cations in multi-tasking mode.

Now consider PC-MOS/386.™ At the touch of a key, you can switch between up to 25 different tasks. And if you have workstations connected to a host, they get multi-tasking, too. Finally...a system that won't hold you back.

8. File/Record Locking and Security

When you decide to implement either a network or a multi-user system, there's a two-fold problem which must be solved: protecting your work from accidental misuse and securing it from intentional theft.

PC-MOS/38 5™ solves both aspects of this problem. Password protected security allows you to assign file, directory, and task access to each user. Plus, files and records are locked using either PC-MOS' proprietary system or NETBIOS emulation.

9. Remote Access

It's been said that information is power...which makes PC-MOS/386™ a deadly weapon to your competition. Imagine on-the-road salespeople being able to file call reports and access your latest inventory data. Picture executives being able to access your corporate database from across the country, or around the world–giving them the information they need, when they need it.

Visualize branch offices tapping time-critical data with nothing more than a modem and a workstation. Working at a home office in the evening or over the weekend suddenly gets awfully productive. And that makes good business sense. The kind of sense you can't afford to be without.

10. The Price...

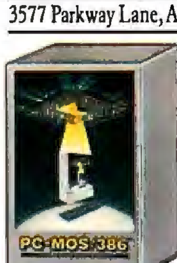
As you evaluate operating systems, ask yourself if it's reasons you're considering...or rhyme. Ask if you're getting a system for tomorrow, or one that was made for yesterday. See if you're being forced to buy new hardware because of *their* software.

And consider this.

Only one operating system in the world can give you the raw power, features, and functionality that you demand. Its name is PC-MOS/386.™ And it's immediately available in one, five and 25-user versions starting at $195.

For the dealer nearest you,	In Georgia:	International/OEM Sales:	Resellers/VARs:
CALL: 800/451-LINK	404/441-2580	404/263-1006	404/448-5465

3577 Parkway Lane, Atlanta, GA 30092 Telex 4996147 SWLINK FAX 404/263-6474



The Software Link/Canada CALL: 800/387-0453

DEALER INQUIRIES INVITED

PC-MOS/386™

MODULAR OPERATING SYSTEM

THE SOFTWARE LINK

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS MAIL PERMIT NO. 201 PISCATAWAY, NJ

POSTAGE WILL BE PAID BY ADDRESSEE

BYTE

Subscription Dept.
P.O. Box 6807
Piscataway, NJ 08855-9940

A simple Windows program for investigating and debugging other Windows programs

Spying on Windows

Michael Geary

Getting started with Microsoft Windows programming can be a daunting experience. Many programmers have reported steep learning curves. Things are getting better, however; Microsoft has greatly improved the documentation in its Software Development Kit for Windows 2.0, and books on Windows programming are starting to appear. In addition, developers are taking a renewed interest, since OS/2 will incorporate a new version of Windows as its presentation manager.

Once you learn the ropes, Windows is a virtual treasure chest of user-interface, graphics, and system management functions, all designed to let you write MS-DOS applications with a Macintosh-like user interface. In fact, Macintosh experience is a great help in Windows programming. The user interfaces are similar, as well as the entire philosophy of application style. As a result, the way you organize a Windows application is a lot like a Mac application rather than a traditional DOS application.

To quote *Inside Macintosh* (whose authors were quoting Firesign Theater), "Everything you know is wrong." Windows applications are so different that experience in writing DOS applications can work against you. To write a good Windows application, you have to do things the way Windows likes them done. A good way to learn that is by seeing how existing Windows applications are put together.

As you might guess, one of the most important features of Windows programming is a window. Nearly every Windows application creates at least one window, its top-level application window, and most applications create and destroy a number of different windows as they execute. Some windows appear as such to the user; others are a programming convenience. For example, every dialog box is a window, and each item (e.g., edit fields and push buttons) inside a dialog box is itself a window. Windows predefines these particular kinds of windows for you; others you build in your own code. Just choosing how to set up this variety of windows can be a challenge.

Since I would rather borrow a good idea from someone than invent everything from scratch, I wrote a program called Spy that sneaks a look at existing applications. Windows maintains a linked list of all the windows that currently exist; Spy scans through that list and gathers up all the information it can find about every window, regardless of the application that created it. Then it displays all this in its own window, in either a summary or a detailed format.

Spy manages to uncover some rather personal information about another application's windows, including the address of its window function, which is the actual program code that manages a window. Yet Spy is a well-behaved Windows application. It doesn't use any undocumented Windows features.

I've used Spy both to investigate techniques used in other Windows applications and to help track down bugs in my own applications. Figures 1 and 2 show Spy on a typical mission, displaying the list of windows being used by several Windows applications.

What Spy Displays

Figure 1 shows Spy's summary view—one line per window with the basic infor-

continued

Michael Geary (P.O. Box 1479, Los Gatos, CA 95031) is a software engineer for Gupta Technologies Inc. and is moderator of the BIX CD-ROM conference.

mation about that window. This includes:

• The type of window: top-level, icon, pop-up, or child. This category determines how a window uses screen space. Top-level and pop-up windows share the entire screen, as do icons, and they can all overlap each other. Child windows exist inside a parent window and are clipped off at the edges of the parent.

• The window handle, a 16-bit value displayed in hexadecimal. When you create a window, Windows assigns it a window handle, and every function in Windows that manipulates a window takes the window handle as a parameter. A window handle is analogous to a file handle in DOS, except the values are not small sequential numbers.

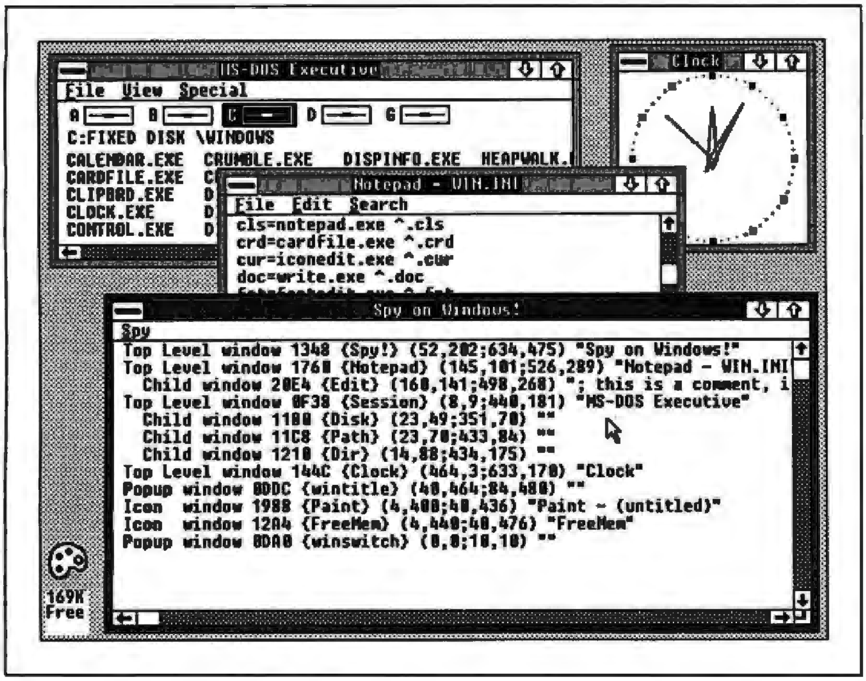


Figure 1: *Spy's summary view, showing each window's handle, class name, location, size, and title.*

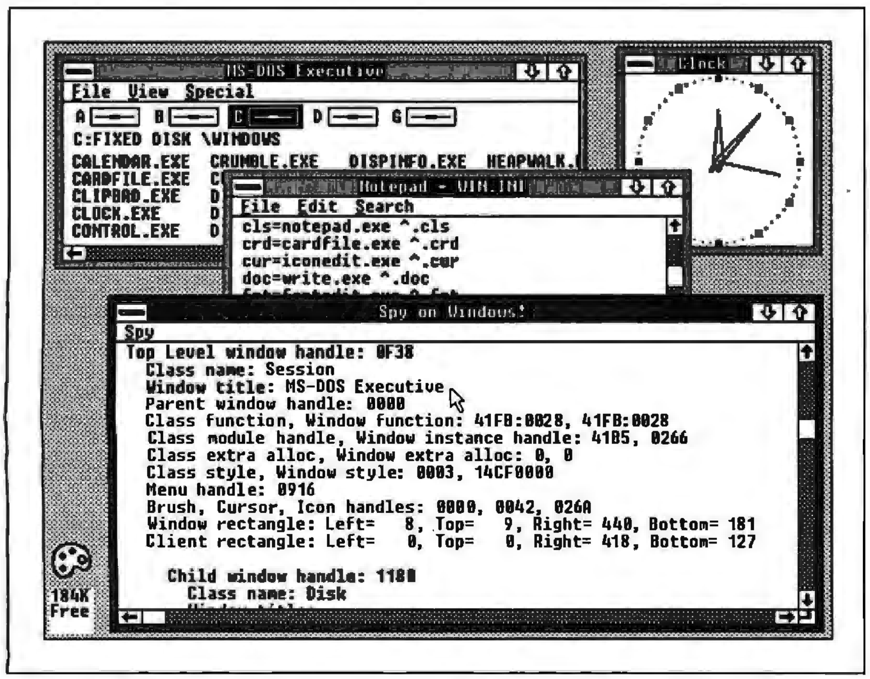


Figure 2: *Spy's detail view. This shows all the information Spy has gathered about each window.*

• The window class name, shown in figure 1 inside curly braces. Every window belongs to one window class or another, and window classes are identified by name. Many windows can belong to the same class, giving you an easy way to create multiple windows that operate similarly. For example, every editable text field in a dialog box is a child window belonging to the Edit window class.

• The window rectangle, four decimal values that specify the left, top, right, and bottom corners of the window. These values represent pixels and are in absolute screen coordinates; for example, (0,0) is the top left corner of the screen.

• The window title. Generally, this is the title that appears at the top of the window. For an edit-control window, it's the text inside the window.

Spy's detail view gives much more information about each window, which I'll discuss later. First, I'll take a look at some of the basic philosophy underlying the structure of a Windows application.

Breaking Out of the Mode

One of the goals of Windows is to provide a modeless, visually oriented user interface for applications programs. A "mode" is a condition a program enters that limits the user's options or changes the meaning of user input. A classic example of a modal system is a nested-menu user interface, where you type a number or letter at the top-level menu to choose another menu, then type another number to get to the next menu, and so on. The meaning of your keystrokes changes at each level of the menu tree, and instead of having all your choices available, you have to navigate to the right place in the menu tree before you can do anything.

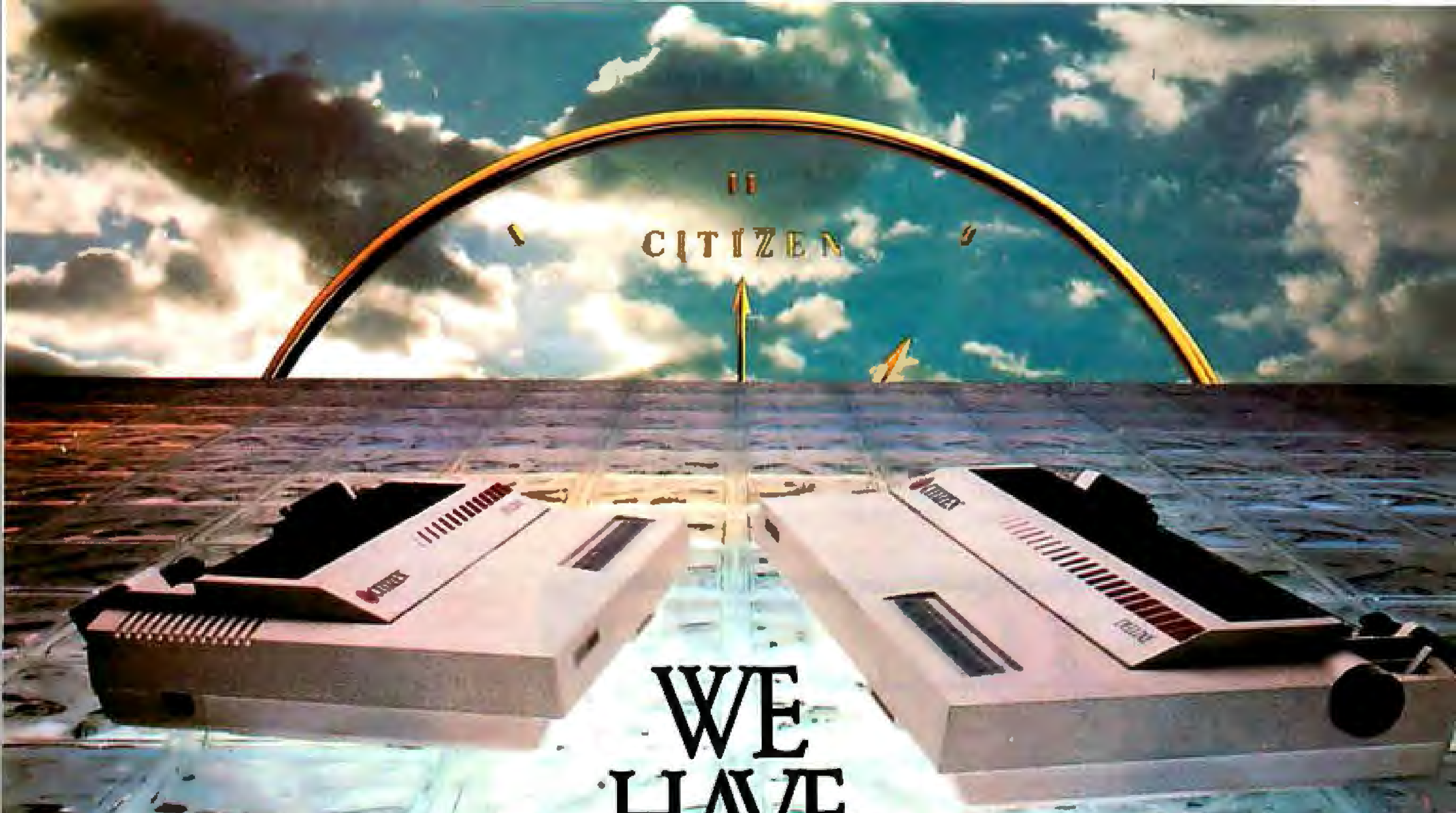
Modal interfaces like this can be tedious, annoying, and confusing. They make the user feel that the program is in control. Well-written applications, for Windows or any other environment, avoid modes as much as possible.

Unfortunately, the structure of a conventional DOS application easily leads to the use of modes. In a conventional program, flow of control is the governing factor. The program is always executing in one portion of the code or another, and when it needs user input, the program will run some particular subroutine that requests that input. Once the program is in this particular input mode, the user's choices are likely to be limited to whatever has been coded in this part of the program.

Don't Call Us; We'll Call You

To avoid modes, a Windows application is built from a collection of windows and

continued



WE HAVE MET THE COMPETITION. AND IT IS US.

Some nine-wire dot matrix printers hide from the competition. Not the new Citizen™ MSP-50.

With the ability to handle all types of paper, top or bottom feed, and a convertible push or pull tractor, it figured it was the best. However, another printer had designs on the title.

So the MSP-50 relied on its high speed, up to 300 cps, to crush the competition. But it wasn't enough. Feature by feature these two contenders battled. Quiet mode. 240 dpi graphics. Color capability. Choice of fonts. Front panel feature controls. Even compatibility with most major software. They were equal. Finally, the MSP-50 put its 80-column print width on the line. Unfortunately, the other machine offered 136.

What was the other machine? The new Citizen MSP-55. Which just goes to prove that if you want to be considered one of the best, you have to be a Citizen in good standing.

For more information call 1-800-556-1234, Ext. 34. In California call 1-800-441-2345, Ext. 34.

CITIZEN™
Printers that run like clockwork.

their associated window functions. The job of a window function is to process messages that are sent to its window. All keyboard and mouse input comes in the form of messages, and Windows sends messages to notify the window function of other system events that affect a window's operation. Windows has over 100 different messages that it sends for different events, and applications can define additional messages.

For example, when you open up several application windows, each window function receives messages to notify it of the current window position and size (WM_MOVE and WM_SIZE). Then, as you move the mouse across these windows, Windows sends WM_MOUSEMOVE messages to the window functions, passing to them the current mouse position.

Windows also has messages for the mouse buttons (e.g., WM_LBUTTONDOWN and WM_LBUTTONUP) and the keyboard (WM_KEYDOWN, WM_KEYUP, and WM_CHAR). In addition, as you move the windows around and bring different ones into view, the window functions receive WM_PAINT messages to notify them to repaint part or all of the screen display. Menu selections send a WM_COMMAND message along with the menu identification code you assigned to the menu item.

There are many more messages. Windows sends a message for just about every event that can affect your application, and your entire application is built around the window functions that process these messages. Once you understand how window functions and messages work, you're well on your way toward success with Windows.

In a way, this is the opposite of traditional DOS programming. Instead of the program calling for user input, writing output to the screen, and generally being in charge, the window function is at the mercy of the messages thrown at it. It doesn't get to decide what message it's going to receive next; it has to process each one and return until the next one comes to it. Since user input generates most messages, this puts the Windows user in charge of the software.

Window Rectangles

Besides the window function and its messages, the most important attributes of a window are its window rectangle and client rectangle. The window rectangle, shown in Spy's summary view, describes exactly where the window is located (at the time that you invoke Spy) in absolute screen coordinates. The window rectangle can be partially or completely off the screen, and the window display is clipped at the screen edges. The window rectangle includes the entire window.

There are two main areas of a window rectangle: the client area and the nonclient area. The nonclient area includes the border, title bar, menu, and scroll bars, and it is generally taken care of by Windows, although your application can take over control of this area if you wish. The remaining portion of the window is the client area, which is under the application's control. The client area is defined by the client rectangle. As a convenience to the application, the top left corner of the client rectangle is always (0,0), and any drawing done with these client coordinates is automatically converted to the proper screen coordinates.

Types of Window Flavors

As I mentioned earlier, top-level and pop-up windows share the full screen as their display space and are defined in terms of absolute screen coordinates. They are clipped at the edges of the screen, and they are also clipped relative to each other wherever they overlap.

Child windows, on the other hand, are displayed inside the parent window's cli-

continued

INTRODUCING High Performance Graphics for the IBM PS/2®

ARTIST 10/16 VGA, the first 1024 x 768 non-interlaced display controller for the IBM PS/2

The ARTIST 10/16 VGA is the first graphic controller that enhances the standard VGA graphic capabilities of the IBM PS/2 models 60 and 80. It displays instantaneous CAD drawings at 65,000 vectors per second.

Our ARTIST 10/16 VGA is the high performer for use with AutoCAD. ARTIST is the most popular board for CAD applications. It is supported by more than 90 software products.

100% Compatible with VGA Software
You get full screen VGA in single or dual monitor configurations. ARTIST 10/16 VGA is a fully authorized IBM third party product with an official IBM board ID number.

For the dealer nearest you call (800) 826-4281.
In Minnesota, call (612) 631-7800.

GT Now Available for ARTIST 10/16 VGA

For AutoCAD applications, our ARTIST boards feature high level editing options and instantaneous zoom and redraw. The ARTIST boards team with display list processing to give you:

- New multiple view windows speed the design process. Each window can be zoomed independently anywhere on the drawing. All views can be edited simultaneously.
- 30 levels of zoom give you unlimited drawing space. Image resolution is enlarged to 16,000 x 12,000 pixels.
- Reduces your AutoCAD zoom and redraw time by more than 90%.
- A Bird's-Eye-View window shows your exact location within a drawing.

Control Systems

2675 Patton Road, P.O. Box 64750, St. Paul, MN 55164
Telex 756 601

ARTIST Distributors: CADsource, CAD Distributors, Corporate Computer Systems, Edwards C.P.E., ICT Technologies, Microsouth, MicroSystems Distribution ParityPlus, and Price Electronics. Also carried by ComputerLand and Entre Computer Centers that specialize in CAD.

ent area. They can overlap each other, but they are always clipped at the edge of the parent's client area. If you move a window, all its child windows move along with it. Child windows are defined in their parents' client coordinates; that is, a child window with an origin of (0,0) would be at the top left corner of its parent's client area.

Child windows are used extensively in Windows applications, as figure 1 shows. They are a handy way to provide different kinds of behavior in different areas of a single application window, since each child window has a window function. For example, the MS-DOS Executive (a file-handling utility that comes with Windows) uses three separate child windows for the disk icons, the current directory path, and the list of files. With this approach, each child window's window function can handle its own operations without worrying about the others—you simply create the windows and let them run.

The child windows inside the MS-DOS Executive window are not readily apparent to the user; in fact, it takes a program like Spy to reveal them. They are a coding convenience for the MSDOS.EXE program. The exact same application could be written with a single window and explicit code to handle things like mouse hit-testing (i.e., determining which area of the window the mouse was clicked on). Using child windows lets Windows do some of the work for you.

You can also create child windows that the end user can manipulate directly. Child windows can have caption bars, a size box (and in Windows 2.0, fat borders), and the like, which let the user move and resize the window. In fact, child windows can look and operate a lot like top-level windows, except for being located inside the parent's client area. The Multiple Document Interface in Windows 2.0 uses this capability.

Window Classes

Often you want to have several different windows that operate in a similar way. Suppose you have a dialog box with several edit fields. Each one is a window (in fact, every item in a dialog box is a child window). Although you need to be able to refer to each item individually, and each one has unique data (such as its text), the actual program code for each item is the same. This suggests that they should share a common window function, and perhaps some common data, along with a block of data unique to each window.

That's exactly what a window class is. A window class defines a collection of windows that generally have a common window function and similar behavior. Windows comes with a number of predefined window classes, most of which are the various "controls," or child windows used inside dialog boxes. These are named according to their function: Edit, Static (for background text), Button (for all kinds of buttons), ScrollBar (which includes size boxes), and List (a list box).

Windows also has many predefined window classes that it uses behind-the-scenes, such as a class for the dialog box window itself, and one for pop-up menus. Applications programs create windows from these classes indirectly through functions like `CreateDialog` (which uses its own class).

Spy's Detail View

Spy gathers much more information than is shown in its summary view. Spy's `Show Detail` menu item toggles back and forth between the summary view and the detail view, which shows everything Spy has uncovered. Besides what's in the summary view, the detail view shows the following for each window:

- The parent-window handle. This is

continued

always zero for top-level windows, but for pop-up and child windows, this line shows the handle of the window's parent. As discussed earlier, a child window is physically located inside its parent. A pop-up window is not limited in this way, but nevertheless it usually has a parent window to which it "belongs." For example, the parent of a dialog box is usually the top-level window that created it.

• The window-function address for both the window class and its particular window. Usually these are the same, since all the windows in a class share a common window function. However, an application can set up a unique window function for any particular window.

• The instance handle and the module handle. You can run more than one copy (or instance) of a Windows application at a time, and each one creates its own windows. So, individual windows are tracked by an instance handle, which is a way to refer to each individual instance of the application. In contrast, features like the window class "belong" to *all* instances of the application, so Windows keeps track of them with the module handle, which refers to the application as a whole and not as any particular instance.

• The class and window "extra" allocation. When you create a window class, you can specify extra storage to be allocated within the class data structure and each window data structure. Windows has functions to access this data, and the application can use the storage space to keep track of anything related to the window or the window class.

• The class style and window style. These are collections of bits that define things like whether the window has a border or title bar. For simplicity, Spy displays these as hexadecimal values, although the bits have symbolic names you would use when programming.

• The menu handle. Top-level and pop-up windows can have menus, and the menu handle lets you refer to the menu in a program. Child windows can't have menus, so the menu handle for a child window is used (mainly in dialog boxes) as another way to refer to it.

• The background brush handle, cursor handle, and icon handle. These tell Windows what color to use for the window's background; what kind of cursor to display in the window; and what icon to display when the window is closed, or iconized. The application can either use these handles or set them to null (0) and determine the background color, cursor, and icon through explicit programming. For example, the Clock program that comes with Windows has a null icon handle so that when the program is iconized, it draws a working clock face with Windows calls instead of a default icon.

• The client rectangle. Spy's summary view shows just the window rectangle; the detail view shows both the window and client rectangles.

Using Spy

Spy isn't just a demonstration Windows application, but a useful tool for finding out how other applications work. For example, those who have done any Windows programming know that you can put scroll bars on a window by setting a couple of style bits when you call the `CreateWindow` function. These standard scroll bars run along the entire width and height of the window, on the bottom and right edges. Spy's own scroll bars work like this. Windows Write has a similar pair of scroll bars, but the horizontal scroll is shorter and has a page number next to it. How does it do that?

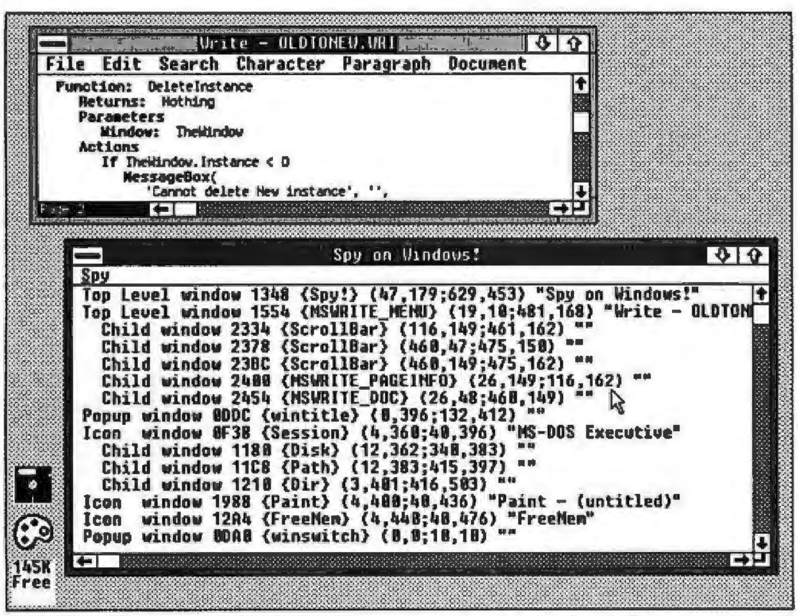


Figure 3: *How does Windows Write display the page number in the lower left corner? Spy uncovers the answer.*

Time for a little industrial espionage, as shown in figure 3. Spy shows that Write's window contains several child windows. One of them has the suspicious class name of `MSWRITE_PAGEINFO`; sure enough, that child window is located right where the page number appears. Write creates a child window for the page number.

Why doesn't the horizontal scroll bar run into that child window, though? Spy's display shows that Write's top-level window contains three child windows of class `ScrollBar`. One is the horizontal scroll bar, one is the vertical scroll bar, and the third is the size box in the lower right corner. The size box also belongs to the `ScrollBar` class, even though it operates differently. Write calculates the positions for the windows itself rather than relying on a default, so it can put them where it wants, and they won't run into each other.

Contrast this with Spy's one and only window, as shown in figure 1. (Yes, in the best double-agent tradition, Spy even spies on itself.) Spy's window has no child windows at all. I set the style bits to have the standard scroll bars when Spy draws its window; Write doesn't do this. Since Write's scroll bars are created explicitly as child windows, the program doesn't have to use the standard scroll-bar location and size; it can place them anywhere it likes.

I've also used Spy several times to help locate bugs in other programs. For example, during development of another program, I found that a particular window was sometimes painted wrong; the screen background and parts of other windows showed through. A quick Spy mission showed that I had inadvertently set the background brush to zero, so the window background was not erased properly (see figures 4 and 5). Since Spy showed me what to look for, finding the problem in my code was much easier.

How Spy Works

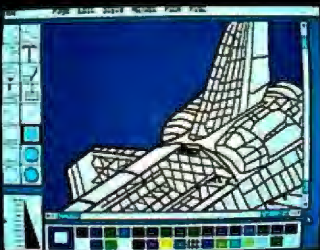
I discussed earlier how Windows sends messages to a window function. There are actually two different ways a message

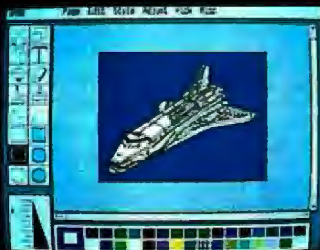
continued

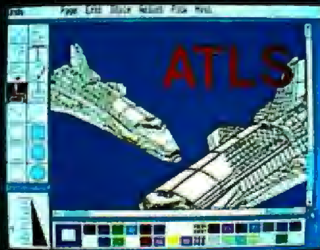
Unlock your desktop with Publisher's Paintbrush.™

Now Supporting IBM® Personal System/2!

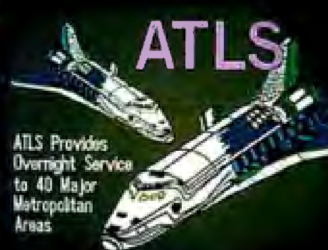
Finally, a paint package designed for desktop publishing! When you create or scan a 300 dpi page, you'll get a 64-screen computer image. That's why Publisher's Paintbrush lets you zoom out and work on the big picture. So you get ultra-sharp resolution without ultra-tedious labor.


One screen of a 64-screen image.


Shrink down to a manageable size.

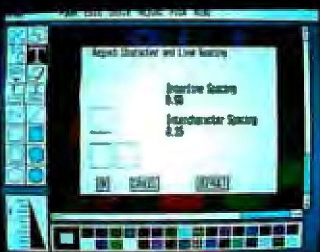


Next, cut-and-paste . . .



A few edits later . . . VOILA!

New typography frontiers: continuously adjustable point sizes, text slope, line and character spacing, extended and condensed type, and letter slant.

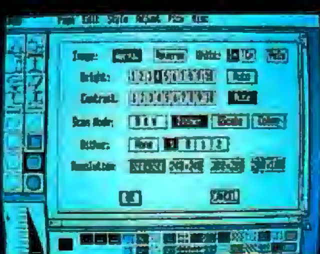

Easy-to-use menus.

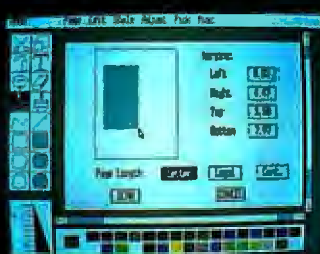

Slant and angle text 1° – 359°.

4- to 250-point type.

Adjustable everything!

Publisher's Paintbrush adapts to most scanners with absolute simplicity. Scan in existing art, logos and diagrams to save many hours of work!


Full use of scanner features.


Scan all or part of a page.

Combine art and type . . .

. . . from several sources!

Publisher's Paintbrush is a quantum leap in power beyond our top-selling PC Paintbrush®. In fact, it's the dawn of a new era for desktop publishing. With it, you can produce pictures of super-high resolution (many times sharper than your screen!) and marry them into text pages. With it, you can take full advantage of desktop publishing packages, laser printers and image scanners. Without it, you're stuck with ragged edges and tedious multi-screen editing.

Publisher's Paintbrush supports major-brand image scanners and hundreds of printers and video display boards. And most desktop publishing packages have built-in links to Publisher's Paintbrush.

Say goodbye to the old boundaries. Say "Show me" to your ZSOFT dealer. Or call our corporate offices for more information.

Publisher's Paintbrush is a trademark and PC Paintbrush is a registered trademark of ZSoft Corporation. Dover Clip-Art Series is a registered trademark of Dover Publications, Inc.

Z-SOFT

1950 Spectrum Circle, Suite A495, Marietta, Georgia 30067, (404) 980-1950 can get to a window function. One is with a SendMessage call, which takes a window handle as a parameter; and Windows then finds the window function for that window. Many of the messages that originate from inside Windows are sent this way, such as the WM_SIZE message, which tells a window function when the user has resized a window.

Other messages, such as keyboard and mouse input, aren't sent directly to a window function but are placed in a message queue. The application's main program pulls messages off the queue and dispatches them to the appropriate window functions. This is performed by WinMain, which basically sits in a loop, calling GetMessage to pick up the next message, and DispatchMessage to send the message to the appropriate window function. WinMain also initializes and terminates the application.

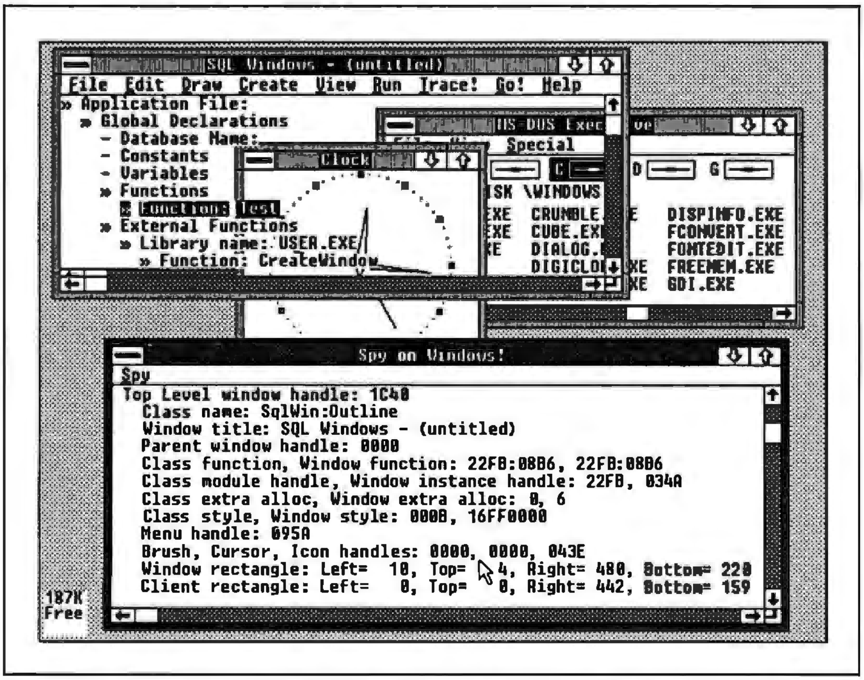


Figure 4: *Tracking down a bug with Spy. Why does my outline window look so strange? (Hint: Spy shows the background brush handle to be 0000.)*

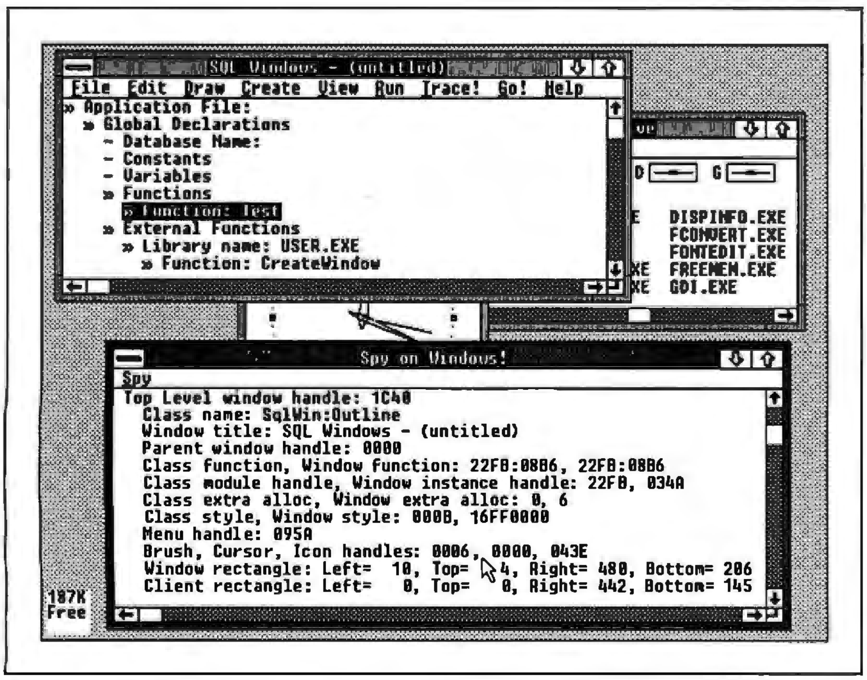


Figure 5: *This looks better. Sure enough, a different value, 0006, is in the brush handle.*

Initializing the Program

Spy's Initialize function performs many tasks. The most important ones are to set up Spy's window class using the RegisterClass function and to create Spy's window with a CreateWindow call. In the process, Initialize checks whether this is the first instance of the Spy program being run or a subsequent instance, because the window class must be registered only once. Each instance has to create its own window, however.

Initialize also calls GlobalAlloc to preallocate a data structure that stores the data Spy gathers on the various windows. At first, however, it allocates only 1 byte in the structure. Later, when Spy scans through the windows, it reallocates this structure to the actual size needed. Windows' memory management is nice this way: Your program can resize any block of memory at any time you want, and the memory never gets fragmented.

Windows pulls off this trick by giving you a handle to the memory block instead of an absolute pointer. Windows can move the data around as needed to accomplish resizing because when you want to get to the data, you call GlobalLock to lock down the data and get the physical address, and GlobalUnlock as soon as you're done. As long as the data isn't locked, Windows can move it around.

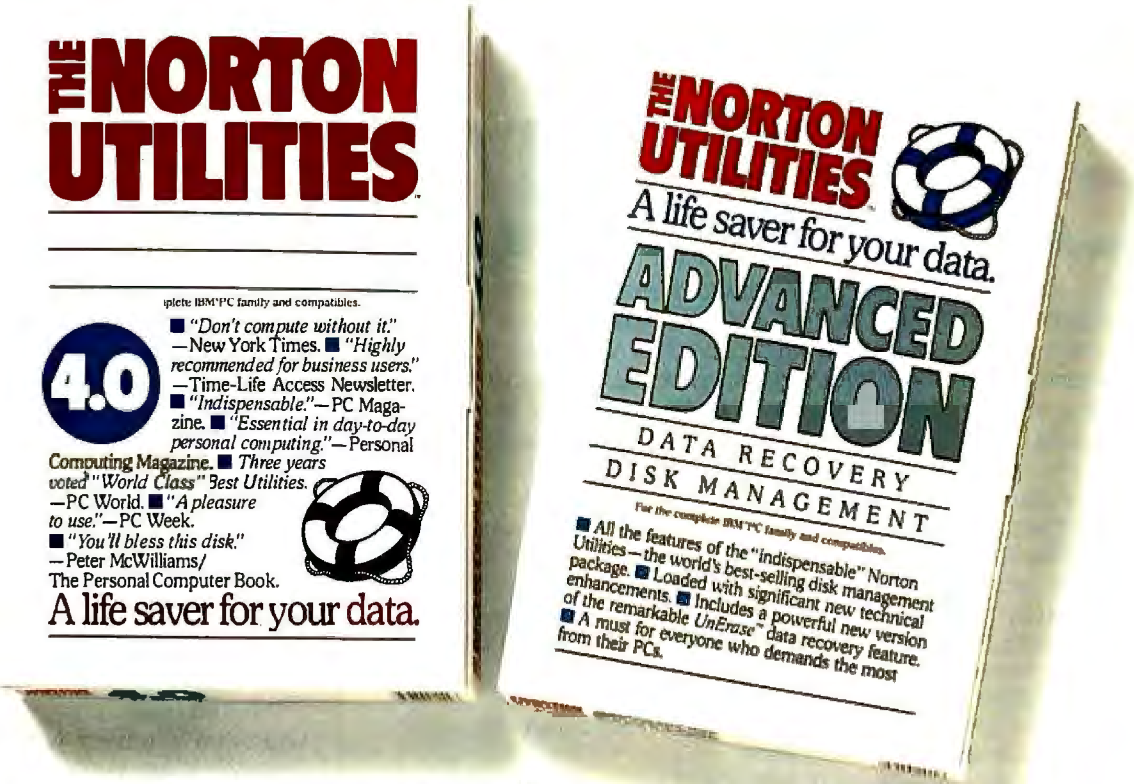
Calling GlobalLock and GlobalUnlock all the time involves extra work, but these functions are fast and provide a real benefit in flexibility of memory use. For situations where the overhead of calling those functions is unacceptable, you can allocate memory with a "fixed" option; then Windows won't move it around, and you can hang on to the absolute address.

Spy's Window Functions

The window functions are the heart of any Windows application, and Spy is no exception. A window function (e.g., Spy's SpyWndProc function) is generally just a switch/case statement that takes care of the messages that the application wants to handle itself. Any message not handled by SpyWndProc is passed through to a default window function, called DefWindowProc.

This trick is the source of much of Windows' flexibility. Windows does very little behind the scenes without first giving your application a crack at it. For example, when the nonclient area of your

continued



Now it's not so lonely at the top.

At last.

After five long years alone at the top of the best-seller lists, the Norton Utilities™ is being challenged by another software package.

Which is no small feat, considering that Version 4.0 is selling even faster than its predecessors did.

Unfair competition.

Of course, in order to compete with the Utilities, this upstart new package first had to copy it.

It had to include the famous *UnErase,*™ which has rescued the derrières of thousands of grateful PC users.

As well as the unique *File Info,* which lets you attach descriptions of up to 65 characters to your files.

Not to mention the *Norton Integrator,* which lets you control everything from a single program and gives you on-line help for each function.

In fact, The Advanced Edition of the Norton Utilities includes every single one of the features, functions and enhancements that have made Version 4.0 so popular.

Right down to the user interface which, according to the *InfoWorld* Review Board makes the Advanced Edition "as easy to use as possible."

And that's not all.

The more the merrier.

Because the Advanced Edition goes on to include a wish list of brand new technical features and functions.

Like *Speed Disk,* for tightening up disk space and optimizing access.

And *Format Recover,* for unformatting your accidentally reformatted hard disk.

As well as a FAT Editor, a Partition Table Editor, a Directory Editor and access to absolute disk sectors.

All of which explains why *InfoWorld* called the Advanced Edition "a great program that has gotten even better."

We just call it good company.

Peter Norton
COMPUTING

Designed for the IBM® PC, PC-AT and DOS compatibles. Available at most software dealers, or direct from Peter Norton Computing, Inc., 2210 Wilshire Blvd. #186, Santa Monica, CA 90403. To order, call 213-453-2361 (VISA and MasterCard welcome). MCI Mail: PNCI. Fax 213-453-6398.

window (e.g., the title bar and borders) needs to be repainted, Windows sends the window function a WM_NCPAINT message. If you want to create some special kind of border instead of the default, you can intercept this message and draw whatever is needed. Most window functions ignore the WM_NCPAINT message and pass it through to DefWindowProc, which paints the nonclient area according to the standard window style.

Now let's take a look at the messages that SpyWndProc handles. For example, WM_COMMAND tells the window function that the user has selected a menu command (among other things). Spy has only two menu commands: New Spy Mission, which tells Spy to make another pass through all the windows and update its display accordingly, and Show Detail, which toggles between the summary and detail views.

Windows sends a WM_DESTROY message when a window is being closed. Since this is Spy's only window, receipt of this message calls the PostQuitMessage function, which puts a WM_QUIT message in the message queue. WinMain calls GetMessage, which pulls the WM_QUIT from the queue and returns FALSE, terminating the main application loop.

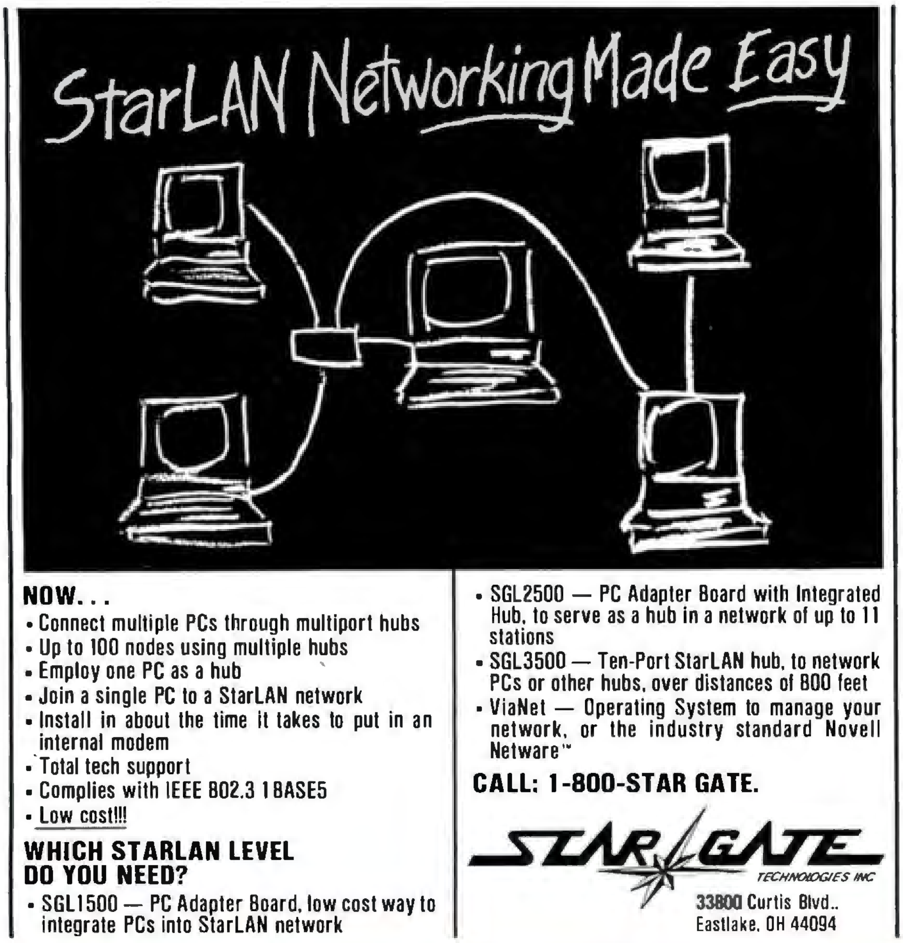
Spy uses WM_HSCROLL or WM_VSCROLL when the user manipulates the horizontal or vertical scroll bar with the mouse. Windows passes these messages along with codes indicating what action has been taken, such as clicking on the scroll bar or the page-up/page-down area.

Since Spy uses the keyboard only for scrolling, it checks for the WM_KEYDOWN message to see when a cursor key is pressed, and simulates the equivalent scroll-bar message. By the way, if you have ever been frustrated with the way DOS and the BIOS handle the keyboard, you'll love Windows, because it gives you complete control over keyboard information. Every key press sends a WM_KEYDOWN message, and every key release sends a WM_KEYUP message. When a key auto-repeats, it sends additional WM_KEYDOWN messages, but there's an indicator to let you know it's an auto-repeat and not an actual key press. These messages are sent for *every* key on the keyboard—even Shift keys. To get this kind of control under DOS, you have to write your own keyboard-interrupt routine. When you just want ordinary ASCII characters in Windows, you can use a WM_CHAR message that is like conventional DOS/BIOS keyboard access.

A special kind of message handled by SpyWndProc is WM_PAINT. Windows considers screen painting to be the lowest pri-

continued


The data that can fit in your
pocket today can have an
impact on what goes into
your pocket tomorrow.
BASF
BASF QUALIMETRIC™ 3.5" MICRO FLEXYDISKS. CERTIFIED 100% ERROR-FREE AND WARRANTED...
FOREVER



ority of all its operations. Whenever part of your window needs to be repainted, Windows sends the window function a WM_PAINT message. However, this message goes to the end of the queue. If there are any other messages for any window in the system, they are processed first. Then, when there is nothing else to do, the WM_PAINT message comes through.

In the meantime, Windows keeps track of which areas of your window need painting. It accumulates these into an update region, which describes the portion of the window to be painted. Suppose, for example, that your entire window needs to be painted, but a couple of pop-up windows overlap part of the window. The update region would include the entire client area, except the portions covered by those other windows.

Finally, whenever the user changes a window's size, Windows sends SpyWndProc a message called WM_SIZE, along with the new size. Spy uses this information to recalculate the maximum ranges of its horizontal and vertical scroll bars.

I Spy; You Can Too

There isn't room here to go into the rest of the details of how Spy works, but I've covered the most important points. Spy illustrates how closely one Windows application can take a look at another application and its windows. It's no accident that Windows works this way; this capability helps applications work together instead of being islands unto themselves.

Windows applications can easily swap data using the window handles and messages I've discussed. A protocol called DDE (dynamic data exchange) describes how to use these features and shared memory to implement "hot links" between applications. For example, Microsoft's new Excel program can exchange data with other Windows applications under program control. They don't all have to be part of a single application.

Spy is a useful tool for Windows developers, both for finding examples of how other programs do tasks and for helping track down bugs. It also shows the basic structure of a Windows application, which is quite unlike a conventional DOS application. Although getting used to this structure can take a while, the payoff is great—Windows provides a rich set of tools for the applications developer. ■

[Editor's note: *The source code for Spy is available on disk, in print, and on BIX. See the insert card following page 208 for details. Listings are also available on BYTEnet. See page 4. You will need the Microsoft C compiler and the Microsoft Windows Software Development Kit to use the source code.*]

In a direct comparison between printer sharing devices, our competition would like to remain anonymous.

- *2, 4 or 6 input ports*
- *Serial or parallel output*
- *0Kb to 1MB of RAM — often soldered, not adaptable for memory expansion*
- *Sequential scanning or concurrent input*
- *Warranty — 90 days to one year*
- *Compatible with most printers*

See us at
COMDEX®/Fall '87
November 2–6, 1987
Las Vegas Convention Center
Las Vegas, Nevada
Booth #813

- *Up to eight serial input ports*
- *Serial and parallel outputs*
- *Concurrent input and output for all ports*
- *Easy, menu-driven configuration*
- *Separate baud rate, flow control, parity and messages for each port*
- *Multiple copy command*
- *Priority print command*
- *User-definable print direction*
- *Field expandable memory up to 2MB*
- *Printer emulation codes and prompt/response messages stored in RAM*
- *Automatic top-of-form generation*
- *Self-testing features*
- *Buffer clear, pause and top-of-form controls*
- *LEDs display buffer status, current port activity and printer ready*
- *Optional parallel/serial converter*
- *Compatible with virtually all printers and plotters*
- *Quality and reliability backed by a one-year warranty; lifetime warranty optional*
- *Unlimited 800# hotline support*
- *UL approved*

LASERNET™ allows up to eight users to quickly and easily share any two printers or plotters.

For the advanced features you demand from a printer sharing device, at an attractive price, LASERNET is your only choice — for general business, CAD/CAM and desktop publishing. And, after 23 years of market leadership, you know that Western Telematic will continue to support its products and customers for years to come. So, don't get left holding the bag. Call us now for a FREE brochure at **(800) 854-7226**, in California (714) 586-9950.

LASERNET
A Product of Western Telematic Inc.

western telematic inc. *5 sterling, irvine, california 92718*
(800) 854-7226 · in california (714) 586-9950 · telex: 467741

Circle 243 on Reader Service Card (Dealers: 244)

The difference is compatibility.

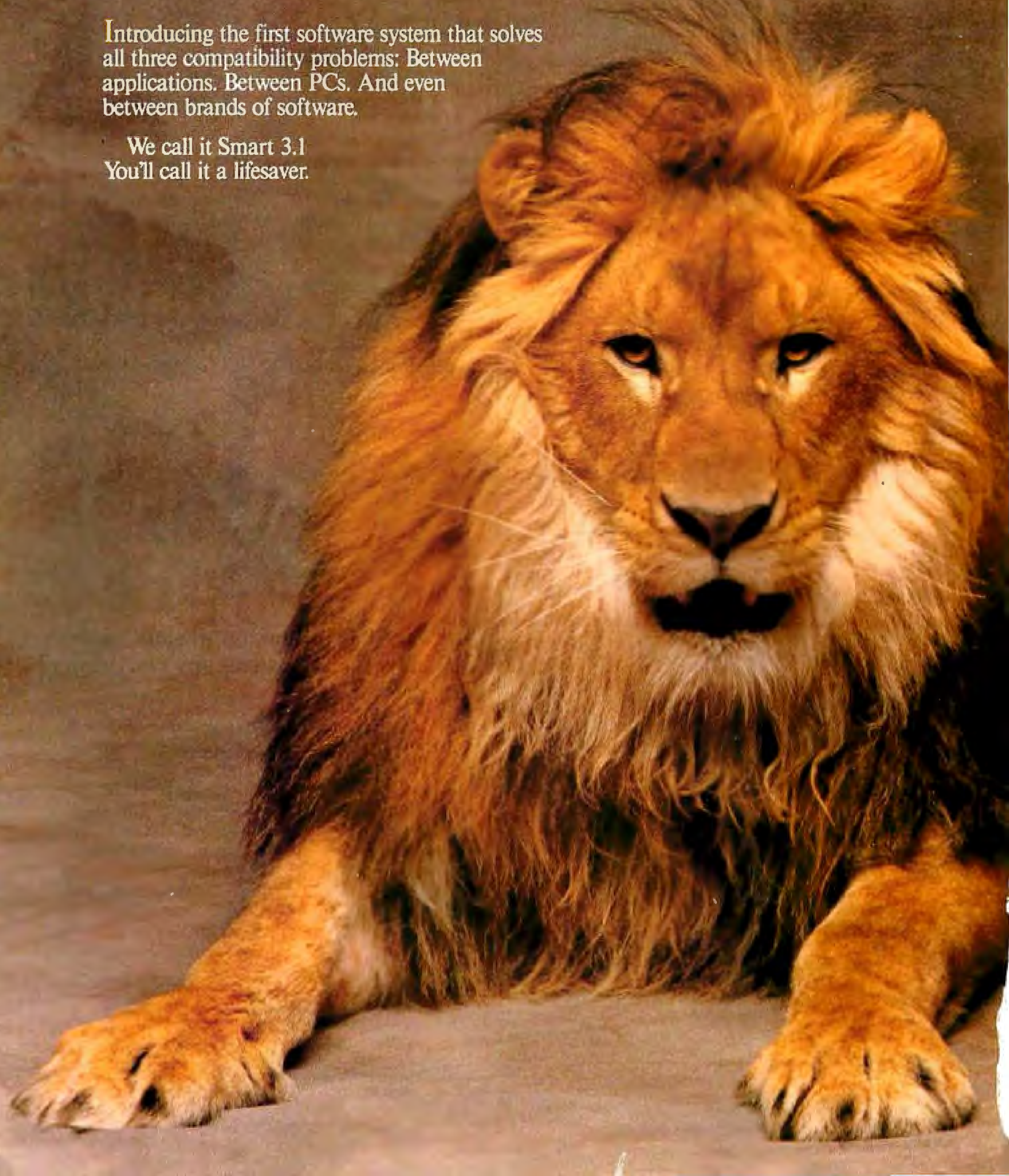
Introducing the first software system that solves all three compatibility problems: Between applications. Between PCs. And even between brands of software.

We call it Smart 3.1
You'll call it a lifesaver.

Between Applications

Smart includes the full range of productivity applications your office needs: Word processing, data base management, spreadsheet analysis, business graphics, and communications.

All perfectly integrated for easy transfer of data from one application to the next.



Send my FREE demo disk.

NAME: ______________________________

COMPANY: ______________________________

ADDRESS: ______________________________

CITY: ______________________ STATE: ________ ZIP: ________

TELEPHONE: ______________________

Check only one of the following:

- ☐ individual computer user
- ☐ work in an organization with many computers
- ☐ computer store retailer
- ☐ VAR, VAD, consultant or developer

CURRENT OR PLANNED ENVIRONMENTS

Check all that apply:

CURRENT	PLAN		CURRENT	PLAN	
☐	☐	NCR Tower Family	☐	☐	Convergent
☐	☐	AT&T 3B Series	☐	☐	IBM PS/2
☐	☐	Microport UNIX System V	☐	☐	DOS single user
☐	☐	SCO XENIX System V	☐	☐	DOS LAN
☐	☐	Altos	☐	☐	Macintosh
☐	☐	Apollo	☐	☐	DEC VMS
☐	☐	Sun	☐	☐	other ________
☐	☐	Plexus			________

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

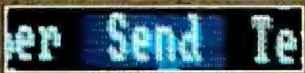
FIRST CLASS PERMIT NO. 2502 SHAWNEE MISSION, KS

POSTAGE WILL BE PAID BY ADDRESSEE

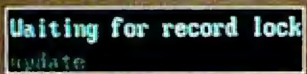
INNOVATIVE SOFTWARE INC.
DEPT. 70010BYT
P.O. BOX 15998
LENEXA, KS 66215-0998

And only Smart lets you create custom programs that actually jump between applications. So, for instance, a single keystroke can transform a worksheet into a 3-D pie chart, paste the chart into a document, and send the document via modem to the home office.

Smart's "REMEMBER" command lets non-programmers create custom programs.



Smart's "SEND" command makes it easy to move data between applications.



Smart automatically protects your data in multiuser environments.

Between Workstations

Even if your PC isn't on a local area network (LAN) now, there's a good chance it will be soon.

And with Smart, you'll be ready.

Every Smart application includes a fully automatic File or Record-Locking scheme for shared data. So right out of the box, Smart can be used either on a single user PC or on a LAN workstation.

There are even versions of Smart for multiuser UNIX® and XENIX®

Which, according to Information Week, makes Smart "the first to provide DOS-UNIX/XENIX LAN connectivity without sacrificing power and features."

Between Software Brands

Old software can be replaced, but old data can't. So you'll be pleased to learn that Smart is fully compatible with files already created with programs like Lotus 1-2-3 and dBase III Plus.

Smart also reads and writes ASCII, SYLK, DIF, and DCA.

Oh by the way, there's something else Smart is perfectly compatible with: You. Fact is, nearly every top computing magazine has praised Smart for its ease of use, on-line help, extensive tutorials, and clearly written manuals.

But why not see for yourself?

Just return the attached card, or call toll-free (800) 331-1763 (in Kansas, Alaska or Canada call (913) 492-3800) and we'll rush you a free Smart demo disk and information kit.

SmartWare®
from Innovative Software

MICROPROCESSOR MASTERY!

The Development System that Supports 150 Different Microprocessors.

A complete solution

Here, at last, is the working environment of the future for developing error-free and efficient microprocessor code. Save time and money with UniLab II's seamlessly integrated toolset:

An 8/16-Bit Universal Emulator – With UniLab's full selection of symbolic debug commands you can quickly display and change all registers, memory, and ports, *plus* set software and hardware breakpoints.

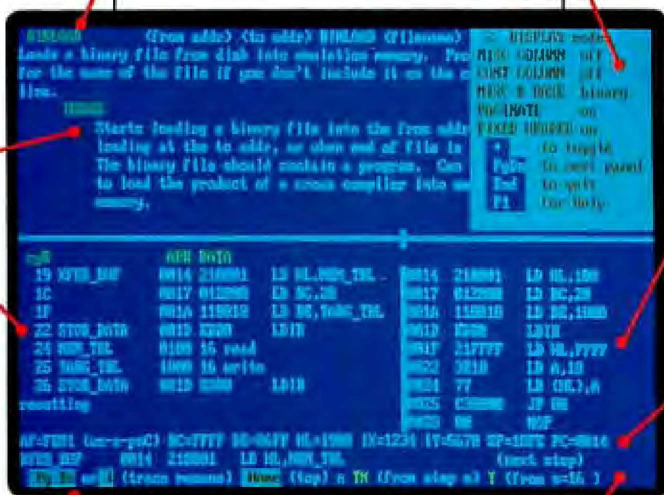
An Advanced 48-Channel Analyzer – Most other development systems are dead in the water if there is a hardware fault, such as a simple bus short. Now, you can use the power of UniLab to home in on both software *and* hardware problems quickly.

An Input Stimulus Generator – You conveniently specify system inputs and observe the results.

A Built-In EPROM Programmer – helps finish the job!

Find Bugs Fast

Searching for bugs by single-stepping through suspect code can take forever. Now, with UniLab, just specify the bug symptom you are looking for as a trigger spec and let UniLab catch the bug for you as your program runs in real time.



UniLab trace filtering eliminates extraneous information and shows you only the program steps of interest.

Get Running Fast

You probably won't use your development system every day. You do need a system that's easy to learn, and easy to come back to. That's UniLab. It lets you use commands or menus – or a mixture of both. The same commands work for all MPUs. Useful help screens, an on-line manual & glossary, instant pop-up mode panels, a quick command and parameter reference, are at the ready.

Affordable, Expandable

At less than $5,000 UniLab costs less than our less-able competitors. You can add our new Program Performance Analysis option to help you optimize your software. If you don't need UniLab's power, other models are available from $2,995. Get the story on UniLab II and how it can revolutionize your software design efficiency, as it has for thousands. Universities, ask about our Education Outreach Discount Program.

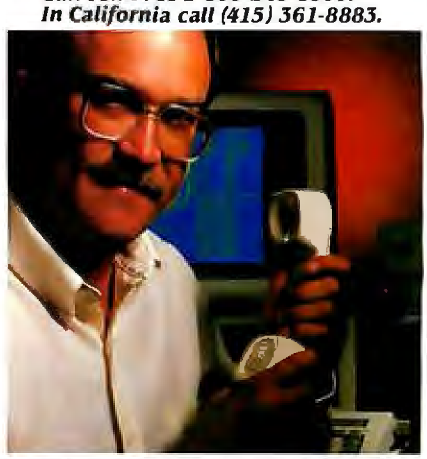
Call Toll-Free 1-800-245-8500. In California call (415) 361-8883.

Development Dreams Come True

Use Unilab's advanced windows to set up your screen the way you want to... view multiple items of interest. Imagine being able to automatically compare a current trace with previous trace data to instantly determine differences. You can set breakpoints, single-step, then go back to the analyzer without missing a beat. If you make a change in your code, use UniLab's built-in line-by-line assembler to instantly patch the fix and test the results. Think of the time savings.

Set up for any 8– or 16–bit processor in seconds!

Thanks to our innovative emulation approach, moving between processor types requires only a new emulation module and software change. You save both money and time: Our inexpensive Personality Paks™ (only about $500) include all the items you need for fast hook-up. Orion MicroTargets,™ functioning target circuits for popular MPUs, let you test your software before your own hardware is ready.



When you own or rent a UniLab II, you get access to Orion's team of Applications Engineers.

ORION
INSTRUMENTS, INC.
702 Marshall St.
Redwood City, CA
94063
TELEX 530942

Circle 160 on Reader Service Card

A report on the performance of the latest generation of floating-point coprocessors

The State of Numerics

Stephen S. Fried

As the line separating personal computers and minicomputers/workstations blurs, a new acronym pops up more and more frequently when comparing machine performance. Alongside MIPS (millions of instructions per second), we must contend with MFLOPS, or megaflops (millions of floating-point operations per second).

MFLOPS become an important measure of performance when you compare machines that are intended to perform image processing, graphics-intensive CAD, weather modeling, chemical simulations, and other applications that intensively manipulate all floating-point numbers.

Until recently, we dealt exclusively in the realm of fractional MFLOPS. Today, however, several devices can deliver a million or more FLOPS (see the text box "Available Floating-Point Solutions" on the next page). This article conveys my early impressions of several of these new devices.

Beyond the 8087

There has been an incredible jump forward in numerics since the advent of the 8087 in 1981. The 5-megahertz 8087 takes 80 to 120 cycles to perform inter-register 80-bit operations. That translates into 16 to 25 microseconds per operation, or 40,000 to 60,000 FLOPS. For the sake of argument, we will call the 5-MHz 8087 a 50,000-FLOPS processor.

In the last several years, the rapid development of silicon compilers and CAE tools has resulted in very fast (i.e., wide) combinatorial arrays that function as accumulators, multipliers, and barrel shifters. These devices are now available to designers as cells in sophisticated CAE systems, and they form the basic building blocks of floating-point devices, including coprocessors. Using these tools and custom design techniques, it is possible to build single-chip floating-point accumulators or multipliers that perform up to 60 million single-real instructions per second (i.e., these devices are up to 1000 times faster than an 8087).

Several companies offer floating-point chip sets and/or cells, including LSI Logic, AMD, Analog Devices, Wietek, and Bipolar Integrated Technologies. These chip sets are most frequently built into specialty data-flow machines that do digital signal processing. In these machines, the numbers flow along a pipeline that contains 10 or more processors. The total throughput of one of these pipelined machines can be in the range of a supercomputer—100 to 1000 MFLOPS.

Until recently, these devices have been accessible only to microprocessor-based systems that have array processors installed. This has changed with the advent of the Wietek/Intel 1163, an application-specific integrated circuit (ASIC) that interfaces the Wietek 1164/65 accumulator/multiplier chip set to the 80386 data bus.

The WTL 1163 glue chip contains an 80386 bus interface unit, a microcode sequencer, microcode ROMs, and a register file arranged as 32 single-real (single-precision) registers. These registers can be used to store 32 single-real operands or 16 long-real (double-precision) operands. The Weitek 1163/64/65 chip set, referred to as the WTL 1167,

continued

Stephen S. Fried is best known for his work in chemical lasers and the use of numeric coprocessors in the IBM PC. He is vice president of research and development at MicroWay (P.O. Box 79, Kingston, MA 02364).

F

The Weitek WTL 1167
Weitek offers a three-chip set in on a small daughterboard that signed to plug into an 80387 socke WTL 1167 features thirty-one registers for storing operands and mediate results to minimize data fers with the 80386. The Weitek which conforms to the IEEE-754 floating-point arithmetic standa said to yield over 3.5 million stones at 16 MHz (see table 1).

The WTL 1167, which costs $ does not provide on-chip transce tals or trigonometric functions.

The Motorola 68881/68882
The Motorola floating-point cop sors use eight 80-bit internal regis perform IEEE-754 standard aritl operations. The 68881 and 6888 tain an on-chip ROM that makes able 22 constants, including pi, powers of 10.

The Motorola coprocessors ar similar to the Intel coprocessors, that they have registers instead of a and were the first coprocessors w line transcendentals, a feature picl by the 80387.

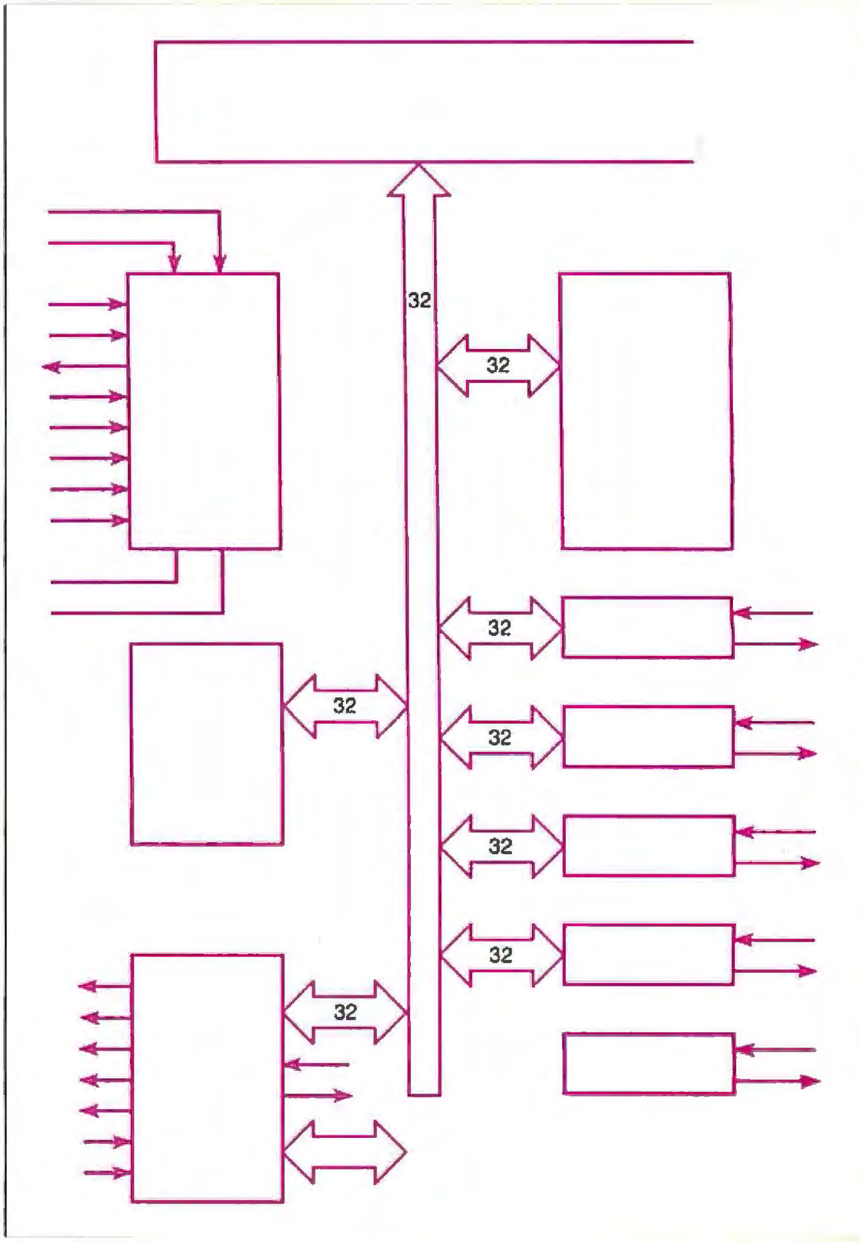
The 68882 achieves higher p mance than the 68881 with the ac of a conversion control unit (CCU improves the performance of the instruction and most arithmetic tions by speeding up the conversic tween the external 16-, 32-, and data formats and the floating unit's (FPU's) internal 80-bit forn

A dual-ported floating-point da ister adds further to performance b mitting concurrent execution of store, and compute operations.

Motorola claims that the 688 perform at two to four times the the 68881. Peak benchmarks of t MHz 68881 have yielded 1.2 n Whetstones, while the 16.67 68882 processor has yielded 3.5 r Whetstones.

Both processors can be used Mac II, and they retail in the $ $800 range.

The Intel 80387
The 80387 from Intel operates in dently of the real or protected mo the 80386 microprocessor. Using



80-bit registers, the 80387 performs all operations in extended real format and automatically converts 16-, 32-, or 64-bit integers or 32- and 64-bit floating-point numbers into this format. The dynamic range of the IEEE-754 standard numerics used by the 80387 is $10^{\pm 4932}$ for extended precision. The 80387 can also accept 18-bit binary-coded decimal operands.

The 80387 has built-in function support for trigonometric, logarithmic, exponential, and transcendental operations. The 80387 can be driven by the microprocessor's clock (synchronous mode) or by a separate clock (asynchronous mode). The 80387 executes 80287 code a factor of 3 faster than the 80287. However, its transcendentals run a factor of 8 faster when properly encoded. You can purchase 80387s for $500 to $800 depending on speed, source, and quantity.

Intel claims "4 to 6 times greater performance" for the 80387 over the previous-generation 80287. The 80387 costs $500 in quantities of 100.

The Inmos Transputer

The Inmos T800 Transputer, from Cambridge, England, was preceded by 16- and 32-bit devices, neither of which had an FPU. Each Transputer has four link interfaces for hooking it up to four nearby Transputers when built into a Transputer network.

The T800 (see figure 1) contains a 32-bit T414B, to which an FPU has been added (on-chip). The 64-bit FPU conforms to the IEEE-754 standard for floating-point arithmetic. The T800 provides no support for transcendentals or trigonometric functions.

Without an FPU, the 10-MIPS T414b Transputer (the 32-bit predecessor of the T800) costs about $2000 packaged on an IBM PC card with 2 megabytes of high-speed dynamic RAM. If you need floating-point performance, the T800 adds $1000 per node to the expense. The onboard FPU delivers 1.5 MFLOPS (at 20 MHz) to 2.25 MFLOPS (at 30 MHz). This is about 15 times the performance of a 10-MHz 8087, or twice the performance of a 24-MHz 68882.

The cost of floating-point performance for the T800 is less than $1000 per MFLOP. True supercomputers run between 100 and 1000 MFLOPS, indicating that a true "personal supercomputer" could be built using Transputers for between $100,000 and $1,000,000 —not cheap, but still a factor of 5 less than current prices of Crays and machines of their ilk.

performs only the four basic operations (addition, subtraction, multiplication, and division), along with conversions and compares. However, it performs them so fast that all the other numeric coprocessors, including the 80387, look underpowered beside it.

The best- and worst-case timings for the combined chip set are 7 cycles (for single-real) to 11 cycles (for long-real) for an interregister operation. At 16 MHz (the nominal speed available as of June 1987), that equates to overall processor timings of 440 to 700 nanoseconds. Expressed as rates, that results in throughputs of 1.4 million (long-real) to 2.2 million (single-real) operations per second. While not quite the 60 million operations claimed by the fastest of the floating-point chip sets, it's still 30 to 50 times the speed of an 8087 and points us in the direction of things to come.

The Problems

The problem with achieving high floating-point throughput in a microcomputer is data-bus bandwidth. For example, suppose our goal is to perform a long-real dot product—multiply two long-real vectors together to create a new long-real vector. Each element will require us to read in two long-real numbers (each 8 bytes wide), perform a multiply, and then store a long-real result. The total transaction involves reading or writing 24 bytes and performing a multiply. Assuming we want to do 1 million operations per second, that specification dictates a system data-bus bandwidth of 24 megabytes per second. A 16-MHz 80386 has a data-bus bandwidth of 32 megabytes per second. This means it is capable of sustaining the I/O associated with 1 million dot product multiplies per second, but would fail at 2 million (which requires a data-bus bandwidth of 48 megabytes per second).

This example reveals the Achilles tendon—vector operations—of coprocessors in general. However, in situations where the numbers being processed are stored and stay in registers, the data bus is not in the computational loop, and it becomes possible to keep the coprocessor fed with operands.

Scalar operations, where the operands are stored in the registers, are the strong point of the WTL 1167 and coprocessors in general.

To achieve the best vector performance, the WTL 1167 was designed in an unorthodox manner. The key to good vector speed is minimizing the I/O associated with passing instructions to the coprocessor. Reducing the data-bus bandwidth required for instructions makes more bandwidth available for operands (data). The designers of the WTL 1167 accomplished this by memory-mapping the processor. In the WTL 1167, the location to which you pass a piece of data specifies what you want the processor to do with the data. The 80386 has a 32-bit data bus and a 32-bit address bus. During I/O, the 80386 actually emits 64 bits of information. By dedicating a 64K-byte block of addresses to the 1167, 16 of the 32 address bits end up getting used to transfer instructions, while the 32 bits of data end up doing their original job, transferring data. By old standards, 64K bytes is a lot of memory to waste on a memory-mapped peripheral, but it is inconsequential in the 4-gigabyte real address space of the 80386.

The WTL 1167 is not the only processor that has benefited from the 100- to 300-ns floating-point times that are easy to achieve with CMOS technology. The Inmos T800 Transputer also achieves overall floating-point speeds that range from 233 ns (single- and long-real adds) to 700 ns (long-real multiplies).

The main reason for the performance differences I uncovered between the T800 and the WTL 1167 (see table 1) is their design philosophy. The WTL 1167 has a reduced-instruction-set-computer-like large register file that contains 32 single-real registers. This makes the WTL 1167 very good for solving problems that are scalar-bound (i.e., a small number of local variables interact with each other). It turns out that even supercomputers are not much better than a WTL 1167 at solving these types of problems. Al Cameron of the Harvard College Observatory finds that a Wietek-equipped Sun workstation provides 20 percent of the throughput of a Cray for performing some astronomical simulations. He attributes this to the fact that the most efficient algorithms for these problems cannot make use of the vector or parallel facilities of the Cray. The WTL 1167 shines in his application because it can keep its variables in the numeric data processor (NDP) and take complete advantage of its throughput (i.e., his problem does not get bogged down in moving numbers from memory to the NDP).

The T800 Transputer architecture is not at all like that of the WTL 1167: The T800 has only three floating-point registers arranged as a stack and an interface to off-chip memory that is less efficient than that employed by the 80386. To get the T800 to hum, you have to take advantage of its strong points—the 4K bytes of very fast on-chip memory (50-ns static RAM) and the fact that the CPU and the NDP are on the same chip and are interfaced in such a manner that it is possible to get a lot of concurrency between the two units

continued

(see figure 1 on page 116).

In the 80386/WTL 1167 duo, the 80386 is actually a little underpowered. As a result, it often acts as a handmaiden for the WTL 1167, rarely getting far enough ahead to make it possible to achieve the kind of overlapping that we are used to with the 8086/87.

On the other hand, the two computational units of the T800 were designed from the start as a matched pair. Consequently, they achieve a high level of concurrency in situations where the real numbers being processed come from the high-speed on-chip storage. As a result, the T800 shines in situations where every floating-point operation also requires a prior 32-bit operation to compute the location of the operands being processed. It also turns out that the deep stack of the WTL 1167 is now considered a hindrance by operating-systems people who are concerned about the time required to save the state of the processor during a context switch. The bottom line for really taking advantage of the scalar speed of the WTL 1167 is to run it on single-threaded operating systems (like MS-DOS) that let you take full advantage of the RISC-like register file of the 1167.

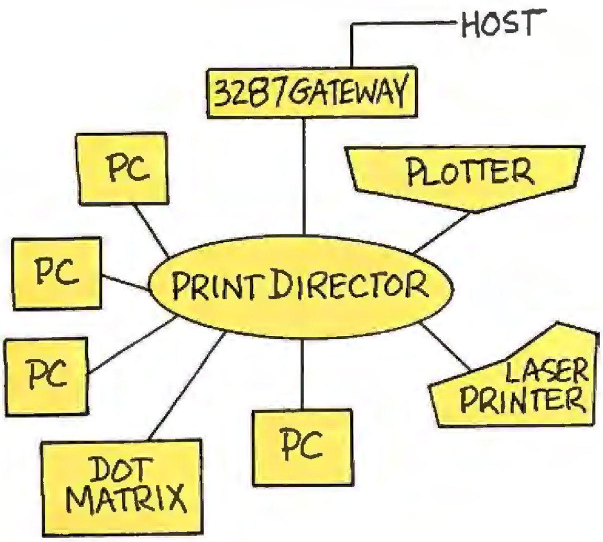
The Rest of the Story

The other factor that complicates processor evaluations is the role of compilers and software tricks in benchmarks. Some benchmarks measure the overall quality of a processor or compiler. A good compiler can improve some benchmarks by a factor of 10 by simply moving loop-invariant code out of loops. The same trick can also be done by the implementer of a benchmark like the Whetstone when translating the Whetstone to a new language like C or OCCAM. Another software factor is library quality: Two of the products I compared are still in the beta phase of their developments (the WTL 1167 and the T800) and are affected by the immaturity of their numerics libraries. In the case of the 8087, it took three years to really refine some of the 8087 libraries.

I ran the first benchmarks (R/M/287) on an AT running at 8 MHz with a 10-MHz 80287 (see table 1). The benchmarks were compiled with R/M FORTRAN. The Savage is 10,000 iterations of the well-known Savage benchmark, while Whet refers to the standard double-precision Whetstone benchmark. The second column (R/M/387) also used R/M FORTRAN, but this time I ran it on an IBM PC AT equipped with an Intel Inboard (16-MHz 80386/80387). The increase in speed of 2 to 2½ times tells us that the main effect in moving from an 8-

continued

Share Any Printer



PrintDirector provides the ultimate in printer sharing.

- ☐ Share any type of printer: laser, plotter, dot matrix, letter quality, and FAX.
- ☐ Connect any type of PC running any application to serial or parallel printers, using proven, reliable asynchronous technology.
- ☐ Print any document without waiting, using concurrent I/O and automatic print buffering.
- ☐ Plain and simple—the Auto-Install program and understandable, pop up menus make set up and operation fast and easy.

Call: **1-800-243-2333; in California: 1-800-247-2006**

PrintDirector

Digital Products, Inc.
108 Water Street
Watertown, MA 02172
617-924-1680

PrintDirector is a trademark of Digital Products Inc.

MHz 80286 to a 16-MHz 80386 running the same code is on clock speed (which doubles).

The next column in table 1 (NDP/387) shows what happens when you recompile the same benchmarks with an 80386 native-code FORTRAN. The typical speedup when you recompile a 16-bit application with a 32-bit compiler is a factor of 3 to 5. There are two reasons for this. First, 32-bit operands can now be stored in registers instead of in memory (this normally speeds things up by a factor of 3). Second, the 32-bit operations are taking place in registers using in-line instructions instead of subroutine calls (another factor of 2).

The last benchmark reinforces the importance of using systems in the manner in which they were intended. In other words, 32-bit processors run best executing 32-bit code.

The fourth column (NDP/1167) was generated by an Avalon accelerator board for the Digital Equipment MicroVAX, a board that gives the MicroVAX the ability to run MS-DOS or Unix applications on an 80386/WTL 1167 accelerator faster than it can run its native VMS applications. The interesting thing here is not the much higher Whetstone achieved by the 1167, but that the WTL 1167 is only slightly better in the Savage than the 80387-equipped AT. I attribute this to the fact that the libraries used were "universal" C libraries, and I expect this to change when hand-coded Wietek libraries are used. However, it also points out that the built-in transcendentals of the 80387 are now very good.

The last benchmarks were taken on a 20-MHz Transputer board running a prototype Inmos T800 Transputer. The code was generated by a stand-alone OCCAM II compiler. The benchmarks were enigmatic at first. The problem turned out to be a beta release of OCCAM that was placing many floating-point scalars in off-chip memory. I had similar problems with the 80387 until I used in-line 80387 operations. As table 1 shows, the Transputer generates the best times but also uses the fastest clock.

The Way to Go?

The bottom line for top-end performance is a toss-up between the WTL 1167 and the T800. If you're doing scalar operations and have an 80386-based machine, the 1167 is the clear winner. If you plan to stay with an XT or AT, then a Transputer-based coprocessor card is a possible solution, especially if you are capable of recompiling in OCCAM and running your application on a network of T800s, in which case there is no upper limit on CPU throughput. ■

Introducing PC AT Version 3

AdaNow

The Serious Ada Compiler For Serious Ada Programmers

If you're a serious Ada programmer today, or expect to be one—if you're serious about DoD business and those new DoD directives —you look for a lot more in your Ada compiler than just validation. And the new fourth-generation Alsys Ada compiler Version 3 for the PC AT and compatibles gives it to you.

Take code quality, for example. Version 3 generates the highest quality code of any Ada compiler on any machine! Check the PIWG benchmarks, and those of U-Michigan. Compare the quality with code from the mature C and Pascal compilers you're used to. Check especially where you might need quality most, and where Version 3 shines—in the implementation of procedure calls and exceptions. The elimination of code associated with unused subprograms gives you large reductions in code size in many applications.

Consider the Runtime Executive. True production quality. No exception-handling overhead is incurred unless an exception is actually raised. The Runtime is optimized for programs running in limited memory, or running for a long time. That's serious.

Consider the Ada-specific high level optimizer technology (not derived from Fortran or C) that removes redundant constraints checks and does so much more for code quality.

Consider robustness and reliability. Version 3 is written in Ada and bootstrapped through itself. That's proof right there that it will compile 400,000 lines of Ada code. Beyond validation, we test our compilers on hundreds of thousands of lines of extra code—from the ACEC tests, and from Ada Repository programs, and from our own specially designed code that breaks most other compilers.

There's a lot more that's serious about Version 3. The new Multi-Library environment, for example, that maximizes the efficiency of programming teams; error messages that correct, instruct, and speed the programming process; human engineering; superb

Send for our free brochure
The Many Facets of Quality.

alsys

In the US: Alsys Inc., 1432 Main St., Waltham, MA 02154 Tel: (617) 890-0030
In the UK: Alsys Ltd., Partridge House, Newtown Rd., Henley-on-Thames, Oxon RG9 1EN Tel: 44 (491) 579090
In the rest of the world: Alsys SA, 29 Avenue de Versailles, 78170 La Celle St. Cloud, France Tel: 33 (1) 3918.12.44

_______Send me *The Many Facets of Quality*.
_______Send me more information on the PC AT Version 3 Compiler.
_______Call me.

Name_______________________________
Company_______________________________
Address_______________________________
City_______________State_______Zip_______
Phone_______________________________

BY IBM

Alsys, Inc. • 1432 Main Street • Waltham, MA 02154

The Many Facets of
Quality

*Ada is a registered trademark of the U.S. Government (AJPO).

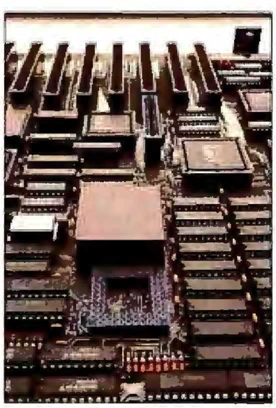
SAVE YOUR PC FROM EARLY RETIREMENT.

GET HAUPPAUGE'S NEW 386 MOTHERBOARD.

386 SPEED–ONLY $1,495

Give your PC a new lease on life! With our industry first 386 MotherBoard, your PC, PC/XT or compatible will revel in speeds equal to the Compaq DeskPRO 386. And faster. Because we've built in 1 Megabyte of high speed RAM and a 387 math coprocessor socket for speeds that will knock you off your rocker.

To keep retirement at bay, our 386 MotherBoard is compatible with the PC/AT (BIOS and I/O) — allowing you to run the new generation of DOS, OS/2. We've also included a 16-bit expansion slot that accommodates the latest I/O expansion card. No accelerator card can give you so much versatility.



With 386 power and true AT software compatibility, your business, desktop publishing and engineering applications will get a boost to boast about! **Technical Features** ▪ 16 MHz 80386 ▪ 1 Megabyte of 100 nsec 4-way interleaved RAM ▪ PC/AT compatible I/O and BIOS for support of OS/2 ▪ Seven 8-bit expansion slots ▪ Two 16-bit expansion slots ▪ One 32-bit RAM expansion slot ▪ Optional 16 MHz 80387 math coprocessor ($695)

Put the power of the 386 into your IBM PC for 1/4 the cost of a 386 computer. And put off your PC's retirement. For more information on our easy-to-install Motherboard, call **1 (800) 443-6284.** In New York, call **(516) 360-3827.**

Hauppauge Computer Works, Inc.
358 Veterans Memorial Highway,
Commack, New York 11725

Hauppauge!

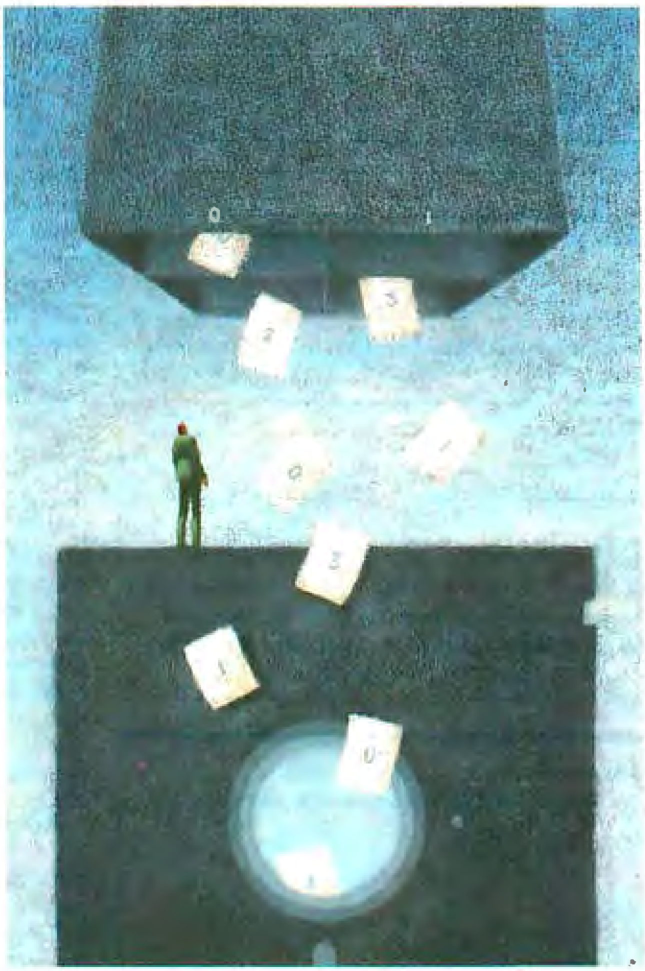
Circle 95 on Reader Service Card

Switching from real mode to protected mode may finally teach us to write well-behaved programs

286/386 Protected-Mode Programming

Joel Barnum

There's no such thing as a free lunch. Some pitfalls await you when you try to update real-mode programs to protected mode. With the pending release of OS/2, a protected-mode operating system for 80286- and 80386-based computers, you might want to rewrite existing real-mode programs or write new programs that execute under protected mode. While it is generally more reliable, protected mode is not totally compatible with real mode and places additional restrictions on programs. This article deals with those restrictions, but first I will discuss the various modes and how they work.



Modes of Operation

Real mode is the "power-on" mode of the 80286 and 80386 processors, in which the processor emulates the 8086 and 8088 microprocessors. Real mode has no memory protection, and the maximum amount of physical address space is 1 megabyte. Currently, PC-DOS and MS-DOS operate in real mode on the IBM PC AT and compatibles.

To switch to protected mode, the operating system sets the least-significant bit in the machine-status word (MSW). In protected mode, the processor verifies every memory access so that one program can't corrupt memory belonging to another. Protected mode also enables multitasking support and virtual memory. The maximum amount of physical memory that an 80286 operating in protected mode can address is 16 megabytes; on an 80386, this number increases to 4 gigabytes.

The 80386 also includes virtual 8086 mode (VM86), which will enable future 80386 protected-mode operating systems to execute unmodified 8086 programs as guest tasks.

The Segment Descriptor

Regardless of mode, well-behaved programs use symbolic names to address segments. For example, listing 1 uses the name of a segment, `DATA`, to initialize the DS (data segment) register. In real mode, `DATA` refers to the segment's address, and the program loader fills in the correct value at load time. In protected mode, however, the program loader fills in a 16-bit number, called a *selector*, that points to the segment indirectly via an 8-byte data structure called a *descriptor*.

The program loader creates the descriptor, which contains information describing the segment. Figure 1 shows a segment descriptor. The *base address* is the segment's starting address; that is, where the program loader put it. The processor uses this address to locate the segment; it is comparable to the segment address in real mode.

The *limit* is the maximum offset allowed in that segment. If a program tries to use an offset greater than the limit, the processor prevents the instruction from executing. This ensures that one program is unable to modify another program's memory.

The *access-rights* byte contains bit fields that indicate the type of the segment; it includes the descriptor privilege level (DPL). Each descriptor has a DPL ranging from 0 to 3, with 0 the most privileged. One segment cannot access another segment that has a higher privilege. These levels help protect system integrity; for example, they prevent appli-

continued

Joel Barnum is a partner at Descriptor Systems (P.O. Box 461, Marion, IA 52302), a training company that presents technical workshops in such subjects as assembly language programming and 286 and 386 system architecture. Joel has a BSEE from the University of Iowa.

cations programs from corrupting the operating system.

The access-rights byte also tells you what kind of access is allowed on the segment. For data-segment descriptors, the access rights indicate that the segment is either readable and writable, or readable only. For code-segment descriptors, the segment can be either executable and readable, or executable only.

The operating system organizes the descriptors into groups called descriptor tables, which contain a maximum of 8192 descriptors each, numbered from 0 to 8191. These tables are of two types: the global-descriptor table (GDT), which contains those descriptors that are available to all programs, and the local-descriptor tables (LDTs), which contain the descriptors for each task's own segments. While each task has an LDT, the system has only one GDT.

To access a segment, you load a segment register with the selector. Its index and table indicator (TI) fields (see figure 2) point to a descriptor from the GDT or an LDT. During this load, the processor locates the descriptor and copies its base address, limit, and access rights into an extended version of the segment register. From then on, the extended segment register contains all the pertinent information about the segment.

Thus, when a protected-mode program loads a segment register, the result is the same as in real mode—the segment register contains the segment's address. The difference is that the protected-mode segment register also contains the segment's limit and access rights so the processor can enforce memory protection. The example in listing 1 should work the same in real or protected mode, because it is well-behaved and uses a symbolic name to refer to its data segment.

When a protected-mode program violates one of the protection rules, the processor generates an interrupt called an exception. An operating system can handle exceptions in various ways, but the most probable action is to terminate the faulting program.

Behavioral Problems

Real-mode programs that are poorly behaved (i.e., that use direct addressing) may not work properly in protected mode because a selector's value doesn't correspond to a segment's address. For example, consider the real-mode program in listing 2, which writes to the IBM PC's color display memory. In real mode, 0B800 hexadecimal is the segment address of the color card's memory. In protected mode, however, a selector with value 0B800h refers to the GDT descriptor located at index equals 1700h. It's unlikely that there's a valid descriptor at that large an index, but even if there is, it's improbable that it's the right one. At best, the program won't display a character. At worst, it will incur a protection exception during the segment-load instruction, because the selector refers to an invalid descriptor.

Well-behaved programs avoid writing to fixed memory locations and use operating-system I/O services instead. You could upgrade listing 2 more easily if it used the DOS display-character function call instead of writing directly to the color card. If you need to write directly to memory for performance reasons, you should restrict such accesses to a single procedure. Then you need to modify only one procedure, not your mainline code.

Another problem related to poor behavior arises if a program performs arithmetic on segment values. For example, a real-mode program with two contiguous 64K-byte segments might add 1000h to the segment address of the first in order to point to the second. Or, a program might calculate its load size by subtracting the segment address of the first segment from that of the second. In either case, the program would have problems in protected mode, because a selector doesn't correspond to the segment's base address. A protected-mode program should not rely on segment arithmetic.

In protected mode, the access-rights byte has enough bits to delineate two types of code segments: executable and readable, and execute only. However,

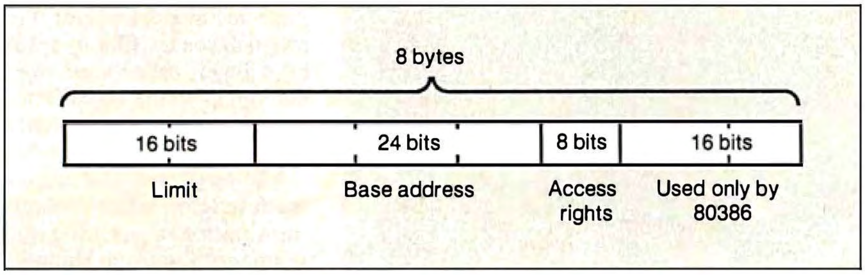


Figure 1: *The 80286 segment descriptor contains the offset limit allowed and the segment's base address and access rights.*

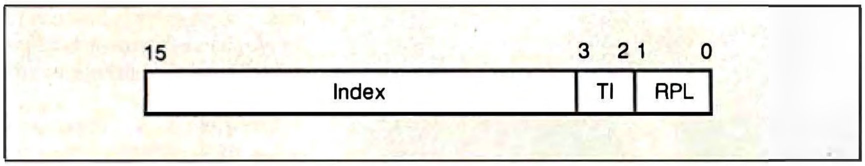


Figure 2: *The selector format. The index can range from 0 to 8191, and it points to the descriptor. The TI equals 0 if the selector refers to the GDT, or 1 if it refers to an LDT. The privilege level (RPL here) is described in figure 3.*

Listing 1: *If you use symbolic segment names to create well-behaved programs, your code should work equally well in real or protected mode.*

```
DATA SEGMENT
  var1 DB ?
DATA ENDS
CODE SEGMENT
  ASSUME CS:CODE,DS:DATA
  MOV AX,DATA          ;symbolic segment name
  MOV DS,AX            ;point to data segment
  MOV var1,1           ;sample access
CODE ENDS
```

Listing 2: *If you use direct addressing to create poorly behaved programs, your code will have problems in protected mode.*

```
  MOV AX,0B800H              ;segment address of color card
  MOV ES,AX                  ;ES to display memory
  MOV BYTE PTR ES:[0],'A'    ;display a character A
```

there is no way to make code segments writable. Thus, a protected-mode program can never use a CS segment override as an instruction's destination.

For example, in both of the `MOV` instructions in listing 3, the program accesses a variable, *var1*, inside the code segment. `MOV AL, var1` works fine as long as the code segment's descriptor says that the segment is readable. However, `MOV var1, AL` fails regardless of the descriptor type, because code segments are never writable in protected mode. To avoid this problem, you should define all variables within a data or stack segment.

The interrupt-vector table is the source of another potential conversion problem. In real mode, the operating system stores the addresses of interrupt-service routines in the interrupt-vector table located at memory address 0000. In protected mode, the interrupt-descriptor table has a different format and doesn't have to reside at any specific address. Therefore, programs that directly manipulate the real-mode interrupt-vector table won't work in protected mode. You can avoid this problem by using DOS functions 25h and 35h to read and write entries in the interrupt table.

Listing 3: *Accessing code-segment variables.*

```
CODE SEGMENT
  ASSUME CS:CODE
  var1 DB 1       ;code segment variable
  start:
  MOV  AL,var1    ;works if execute and read access rights
  MOV  var1,AL    ;never works, can't write
CODE ENDS
```

Table 1: *Protected-mode sensitive instructions.*

IN	Read a port
OUT	Write to a port
INS	Read a string from a port
OUTS	Write a string to a port
CLI	Disable interrupts
STI	Enable interrupts

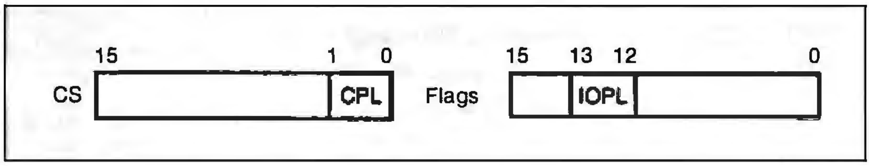


Figure 3: *The CPL is located in the last two bits of the code segment's selector. The IOPL is found in the program's flags. You can modify the IOPL only when CPL=0; you can execute sensitive instructions only if CPL < = IOPL.*

Sensitive and Privileged

To make a system as reliable as possible, the operating system can prevent you from executing so-called sensitive instructions (see table 1) while in protected mode. If you use these instructions incorrectly, your program can crash the system. For example, if you issued an invalid `OUT` instruction, you could turn off the PC's direct-memory-access controller and cause a memory parity error.

The operating system controls who can execute sensitive instructions via the I/O privilege level (IOPL) bits in the program's flags. (Each program has its own set of flags.) To be allowed to execute sensitive instructions, the current code segment's privilege level, or CPL, must be higher (i.e., numerically lower) than the IOPL; otherwise, the processor will generate an exception (see figure 3).

Existing real-mode programs that use sensitive instructions may fault in protected mode, depending on how the operating system assigns CPL and IOPL. To avoid such faults, you can compare these values programmatically to determine whether a particular program can execute sensitive instructions. The procedure `check_sensitive` in listing 4 performs this comparison.

Listing 4: *This routine determines whether you can execute sensitive instructions. It takes no inputs. After execution, the carry flag will be 0 if you can use sensitive instructions at the segment's current privilege level, or 1 if their use will generate an exception.*

```
check_sensitive  PROC
                 PUBLIC check_sensitive
  PUSH  AX
  PUSH  BX
  PUSHF                      ;save flags (IOPL) on stack
  POP   AX                   ;copy flags to AX
  AND   AX,3000H             ;mask all but IOPL
  SHR   AX,12                ;right-justify IOPL
  MOV   BX,CS                ;CPL resides in CS
  AND   BX,3                 ;mask all but CPL
  CMP   BX,AX                ;compare CPL and IOPL
  JA    no_sensitive         ;jump if CPL > IOPL
  CLC                        ;sensitive instructions OK
  JMP   SHORT cs_exit
no_sensitive:
  STC                        ;exception occurs on sensitive
cs_exit:
  POP   BX
  POP   AX
  RET
check_sensitive   ENDP
```

Only the operating system or other highly privileged programs with a CPL equal to 0 can execute privileged instructions; they would pose too great a risk in applications programs. Table 2 contains a list of privileged instructions. Of these, only the `HLT` instruction exists in real mode, so it's the only one you have to look for; you won't have upgrade problems with any of the others. New protected-mode programs can use the procedure `check_privileged` in listing 5 to

continued

Table 2: *Protected-mode privileged instructions.*

HLT	Halt the processor
LGDT	Load the GDT register
LIDT	Load the interrupt-descriptor-table register
LLDT	Load the LDT register
CLTS	Clear the task-switched flag
LMSW	Load the MSW
LTR	Load the task register

The following privileged instructions exist only on the 80386:
MOV to/from control registers
MOV to/from test registers
MOV to/from debug registers

Listing 5: *This routine determines whether you can execute privileged instructions. It takes no input. After execution, the carry flag will be 0 if you can use privileged instructions at the segment's current privilege level, or 1 if their use will generate an exception.*

```
check_privileged    PROC
                    PUBLIC check_privileged
   PUSH  AX
   MOV   AX,CS              ;CPL resides in CS
   AND   AX,3               ;mask all but CPL
   JNZ   no_privileged      ;jump if CPL <> 0
   CLC                      ;privileged instructions OK
   JMP   SHORT cp_exit
no_privileged:
   STC                      ;exception occurs on privileged
cp_exit:
   POP   AX
   RET
check_privileged ENDP
```

Listing 6: *Routine to determine if an offset is usable within a segment. The selector for the segment goes in BX, and the offset goes in CX. After execution, the carry flag will be 0 if the selector:offset is fine, or 1 if its use will generate an exception.*

```
check_segment_limit    PROC
                       PUBLIC check_segment_limit
   PUSH  DX
   LSL   DX,BX                  ;obtain the segment's limit
   JNZ   exception_return       ;exit if bad selector
   CMP   CX,DX                  ;compare offset and limit
   JA    exception_return       ;jump if offset above limit
   CLC                          ;selector:offset OK
   JMP   SHORT csl_exit         ;return to caller
exception_return:
   STC                          ;bad selector or offset
csl_exit:
   POP   DX
   RET
check_segment_limit ENDP
```

SAVE $20

United States ☐ One Year $22 ☐ 2 Years $40
Canada/Mexico ☐ One Year U.S. $25
Europe ☐ $69 (air delivery), U.S. Funds enclosed
☐ $37 (surface mail), U.S. Funds enclosed

☐ BILL ME. If I'm not completely satisfied with my first copy, I'll simply write "cancel" across your invoice, mail it back, and my subscription will be cancelled.
☐ Check Enclosed ☐ Bill VISA ☐ Bill Mastercard
Please allow 6-8 weeks for processing your subscription.

47ITNU-3
Name ______
Address ______
City/State/Zip ______
Acct.# ______ Expires ______
Signature ______

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS MAIL PERMIT NO. 201 PISCATAWAY, NJ

POSTAGE WILL BE PAID BY ADDRESSEE

BYTE

Subscription Dept.
P.O. Box 6807
Piscataway, NJ 08855-9940

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS MAIL PERMIT NO. 201 PISCATAWAY, NJ

POSTAGE WILL BE PAID BY ADDRESSEE

BYTE

Subscription Dept.
P.O. Box 6807
Piscataway, NJ 08855-9940

determine whether they can execute privileged instructions.

Living Within Limits

In real-mode programs, you can freely use offsets greater than the size of a segment. For example, if two small segments are contiguous, you can set a segment register to the first and use a large offset to access the second. However, if you try to access beyond a segment's limit in protected mode, the processor will generate an exception interrupt. To prevent these faults, a program—particularly a procedure to which an offset is passed—can use the load segment limit (`LSL`) protected-mode instruction to verify that the offset lies within the segment's limit.

`LSL` requires two operands: a destination 16-bit register and a source selector in a register or memory location. The instruction verifies that the source selector is valid; if it is not, `LSL` clears the zero flag. A selector might be invalid for a variety of reasons, the most common of which is a privilege-level violation. If the selector is valid, the processor copies the limit from the segment's descriptor into the destination register. The `check_segment_limit` procedure in listing 6 uses `LSL` to determine whether a particular selector:offset address is valid. [Editor's note: *Listings 4 through 6 are available as PMODE.ASM in Microsoft MASM 4.0 assembly language source code in print, on disk, and on BIX. See the card following page 208. This file is also available on BYTEnet. See page 4.*]

Time to Pay the Piper

Protected-mode programs are more robust and reliable than real-mode programs and can take advantage of larger memory addressability, but some problem areas exist in converting programs from real mode to protected mode. These areas are contained in the selector and the segment descriptor. The privilege level in the selector controls the use of sensitive and privileged instructions. The protection afforded by access rights, making sure that you don't write to a segment if you're not supposed to, or even read it unless you have the right to, can cause problems where you least expect them. The use of limits ensures that you are not jumping unintentionally (or intentionally) into another segment with offsets that exceed the size of the current segment.

Finally, one subject comes up again and again. It is what anyone and everyone with anything to say about programming will tell you: Write well-behaved programs. While performance considerations in the past may have dictated that you not always follow that advice, the time has come to pay the piper. ■

Introducing the VGA card

It's VEGA VGA,™ just what you'd expect from the company that brought you the award-winning VEGA Deluxe.

This short card gives you VGA compatibility without having to buy a new IBM PS/2 machine.

And at the same time, it gives you capabilities that even IBM can't match.

To begin with, VEGA VGA provides a graphic resolution of 640 x 480. And gives you 16 spectacular on-screen colors from a palette of 64, with a variable frequency monitor.

Or if you're using an analog monitor, you can select 256 simultaneous colors with 320 x 200 resolution from a palette of 262,144 or 16 colors with a resolution of 640 x 480.

But this isn't just a BIOS level VGA card. VEGA VGA offers complete software compatibility for all VGA programs. And that's not all.

VEGA VGA is backwards compatible, working with your favorite software programs for EGA, CGA, MDA and Hercules. Guaranteed.

that'll knock your socks off.

And due to our innovative design, VEGA VGA supports both analog and digital monitors. Which means it will work with the monitor you're currently using.

And it works up to 400% faster than an EGA card.

VEGA VGA also incorporates surface mount technology with our custom IC chips—for less heat, more reliability and durability.

VEGA VGA provides drivers for an additional high resolution of 800 x 600.

And finally, VEGA VGA is the only card that comes with a full three year warranty.

See Software Compatibility Guarantee for full details. Video Seven reserves the right to change specifications without notice.

All this for a fraction of the price of a new IBM PS/2. Plus a new pair of socks.

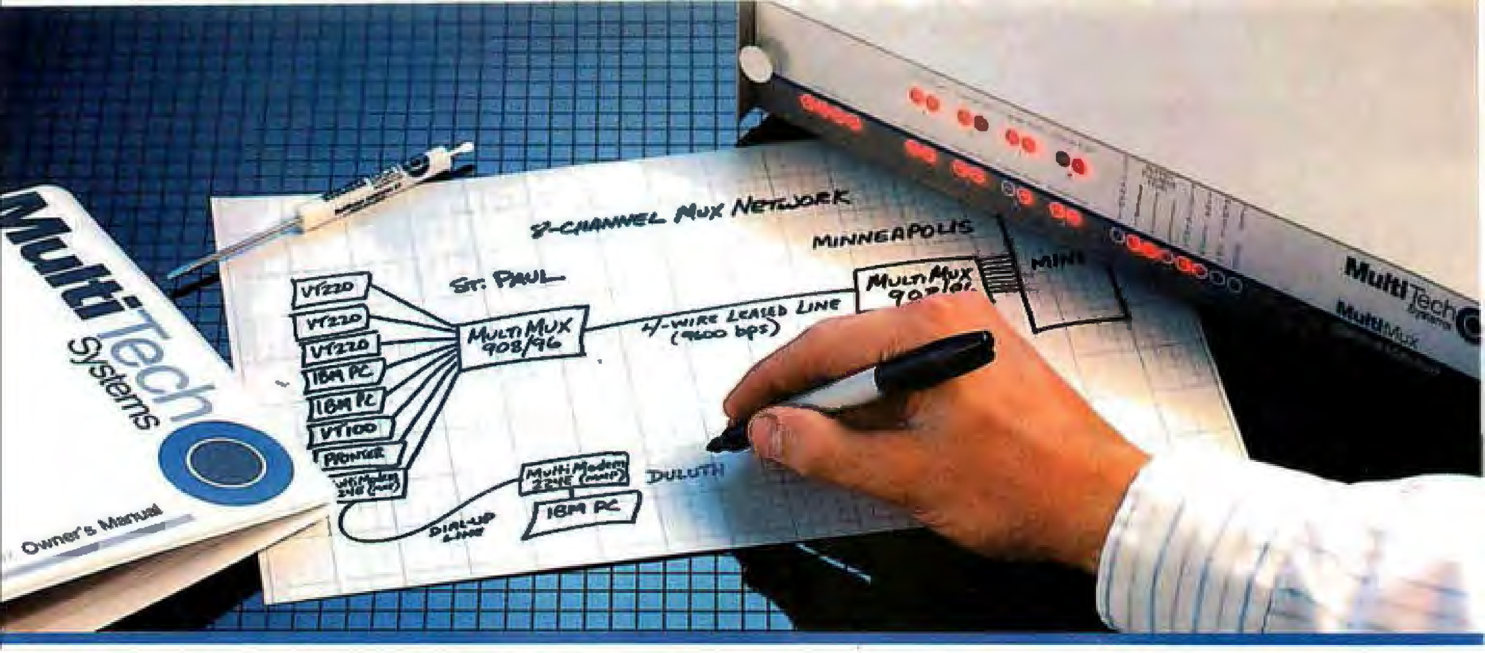
For more information, visit your nearest Video Seven dealer.

Video Seven Inc., 46335 Landing Parkway, Fremont, CA 94538. 1-800-238-0101 (in California, 1-800-962-5700.)

VIDEO SEVEN

We make a clear difference

Circle 238 on Reader Service Card



Four- and Eight-Channel Statistical Multiplexers from Multi-Tech Systems:

When it has to be as simple as it is smart

- There are a lot of good reasons to buy a pair of statistical multiplexers. Reduced line expenses, network security and the elimination of transmission errors are the three main ones.
- Unfortunately, most multiplexers come with a big reason not to buy them: They're nearly impossible for mere mortals to install and understand. In other words, they're just not worth the trouble.
- With that in mind, we'd like to introduce you to the new MultiMux,™ from Multi-Tech Systems. The biggest difference between the MultiMux and other muxes is that you can easily install a pair of them yourself, without the excessive time, expense and frustration you've avoided in the past.
- You may know us best for our MultiModem™ line of 1200 and 2400 bps dial-up modems. The same simple approach we use for our modems is now used for our MultiMux. We even use similar "AT" type commands.
- The MultiMux is offered in four- and eight-channel versions, with or without an integral 9600 bps V.29 modem. Instead of paying for up to eight dial-up lines for your asynch terminals, PCs or printers, you can use a pair of MultiMuxes to combine them all into a single leased line, for much less money.
- Convinced? If not, consider that the MultiMux is completely command driven, with a supervisory port, help menus, downline loading, full LED display, statistical reporting, and an integral 1200 bps dial-up modem for remote configurations and diagnostics. Also, consider that the MultiMux (like all of our products) is manufactured and supported by us in Minnesota, and that we've been doing so since 1970. Finally, consider our price. You'll find that to be a pleasant surprise too.
- Please call us toll-free at **1-800-328-9717**, for additional information...get a multiplexer that's as simple as it is smart!

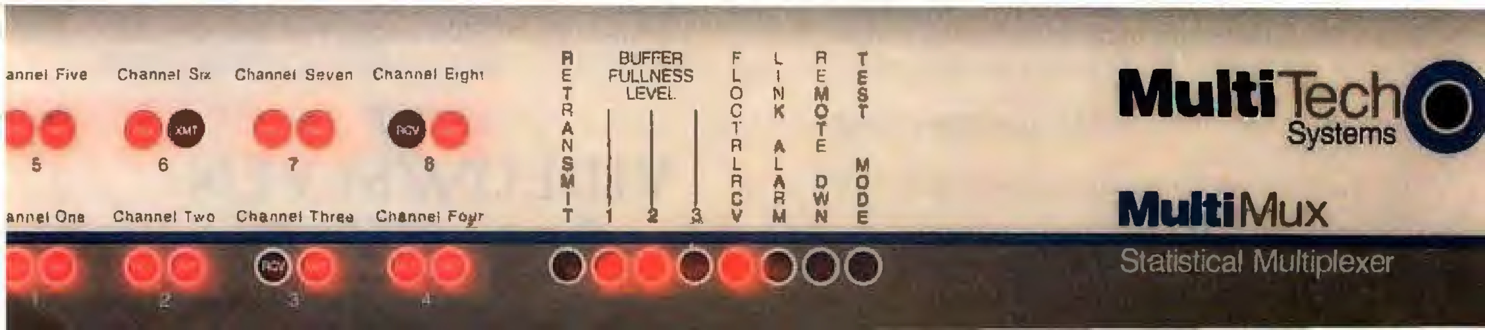
MultiMux, MultiModem and the Multi-Tech Systems logo are trademarks of Multi-Tech Systems, Inc.

MultiTech Systems

Circle 144 on Reader Service Card (Dealers: 145)

The right answer every time.

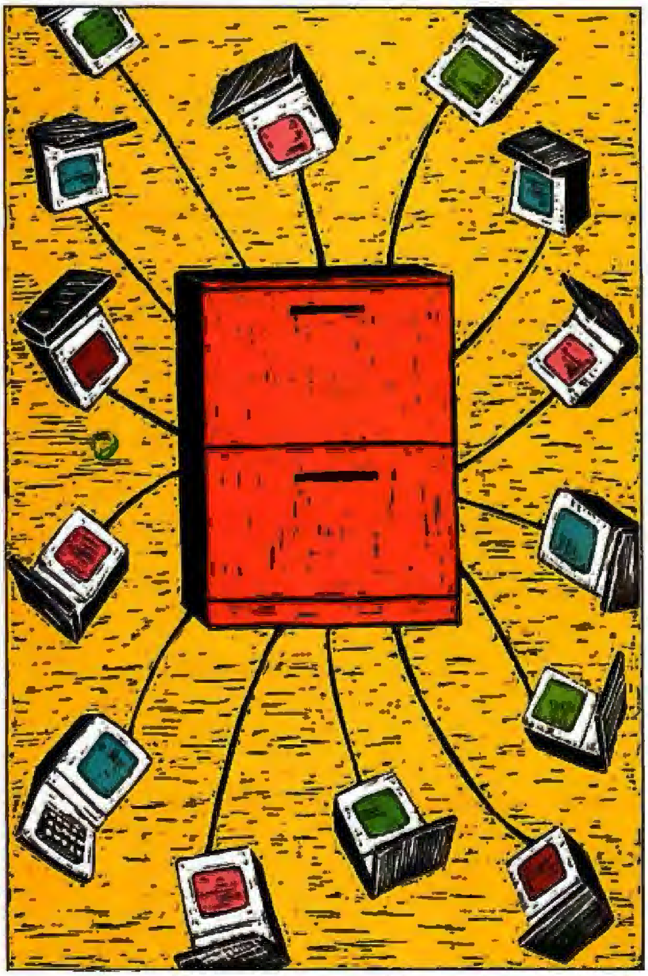
Multi-Tech Systems, Inc. • 82 Second Avenue S.E. • New Brighton, Minnesota 55112 U.S.A.
1-800-328-9717 • 1-612-631-3550 • FAX 612-631-3575 • TWX 910-563-3610 (Domestic) • Telex 4998372 MLTTC (International)



Solving the RT's network problem

The IBM RT Gets Connected

Jason Levitt



IBM's recent foray into the technical workstation market has left little doubt that the Unix operating system is the incontestable standard in that field. The IBM RT PC, IBM's initial entry in this market, simply joined a host of other computers whose companies were already in a flurry of development to equip their latest workstations with the fastest CPU, memory, and networking options available. The initial RT offering, seen in this light, was interesting, yet ineffectual. [Editor's note: *For a description of the IBM RT PC, see Richard O. Simpson's article "The IBM RT Personal Computer" in BYTE's* Inside the IBM PCs, *Fall 1986.*]

The RT was a relatively fast (about 2 million instructions per second) machine using a proprietary reduced instruction set computer architecture, but with poor floating-point performance and no distributed processing capabilities except for limited PC Network support and the prehistoric Unix standby, uucp (Unix-to-Unix copy).

The latest IBM offerings, however, place IBM systems competitively against such established distributed systems as Apollo's Domain system and Sun's Network File System (NFS) as a potentially heterogeneous distributed computing environment for technical workstations. The key to this new computing environment is an extended version of AIX, IBM's port of Unix System V for the RT, incorporating a set of facilities called Distributed Services (DS). DS lets a group of networked RTs share files and directories. It also lets processes on different machines communicate using a form of distributed interprocess communication.

Distributed File System

A distributed file system should let a user on one machine (the client) have access to files and directories on another machine (the server). DS provides this service to the user through a modified Unix `mount` command. The normal command allows only the mounting of file systems; the DS version, in addition, accepts mount combinations of files or directories that are local and remote. For example, you can mount a remote directory so that it appears as a local directory on your system.

Unlike NFS, where the system administrator must explicitly mark a file or directory before a client can mount it, DS allows the mounting of any file or directory that a client can name locally or on a server, provided the client satisfies the authorization criteria listed in table 1. The advantage of such casual mounting is increased user interaction across the network and close adherence to Unix system semantics.

Figure 1 shows a four-node RT network before and after two users—`clienta` and `clientb`—have issued the series of remote `mount` commands shown in table 2. The clients share the source code tree from `src_mach` and the file `/etc/passwd`. Access to these remote files and directories is transparent to users and programmers. Thus, users on `clienta` and `clientb` can access files in the source tree without realizing that the files exist on another machine. System calls such as `write`, `read`, and `open` perform as they do on local files. The exceptions to this are that DS does not let a user reference a remote device driver—hence, you cannot directly use printers or terminals on a remote ma-

continued

Jason Levitt is a Unix consultant with a B.S. in computer science from Indiana University. He can be reached at P.O. Box 49860, Austin, TX 78765.

Illustration: Claudia Tantillo © 1987

Table 1: *Mount authorization criteria in Distributed Services.*

1. Superuser can issue any mount.
2. Members of system group can issue local device mounts defined in `/etc/file` systems.
3. Other users/groups are allowed to perform directory/file (but not device) mounts if these processes have search permission for the requested directory/file, own the mounted-upon object, and have write permission in the parent directory of that object.

Table 2: *Mounting remote files and directories with the* `mount` *command. See figure 1.*

Mounts issued by user `clienta`:

```
mount -n src_mach /etc/passwd/etc/passwd
mount -n src_mach /src/mysrc/machsrc
```

Mounts issued by user `clientb`:

```
mount -n src_mach /etc/passwd/etc/passwd
mount -n src_mach /src/updates
```

chine—nor does it allow the use of shared memory on a remote machine with the `shmat` system call.

Vnodes

The ability to mount remote files makes it possible for files from several different machines to coexist on the same machine; you can even mount several copies of the same file at various points on the same machine. The traditional view of a Unix file as represented by an i-node structure is not sufficient for remote representation.

DS uses the NFS concept of Virtual File Systems (VFS) and vnodes to represent files and directories. As figure 2 illustrates, the VFS and vnodes place a layer between the Unix system calls and the i-node structure of the Unix file system. Each time a remote mount occurs, a new VFS structure is created, with the mounted file or directory becoming the root of the VFS. Each file and directory in the new VFS is represented by a vnode structure. The information contained in

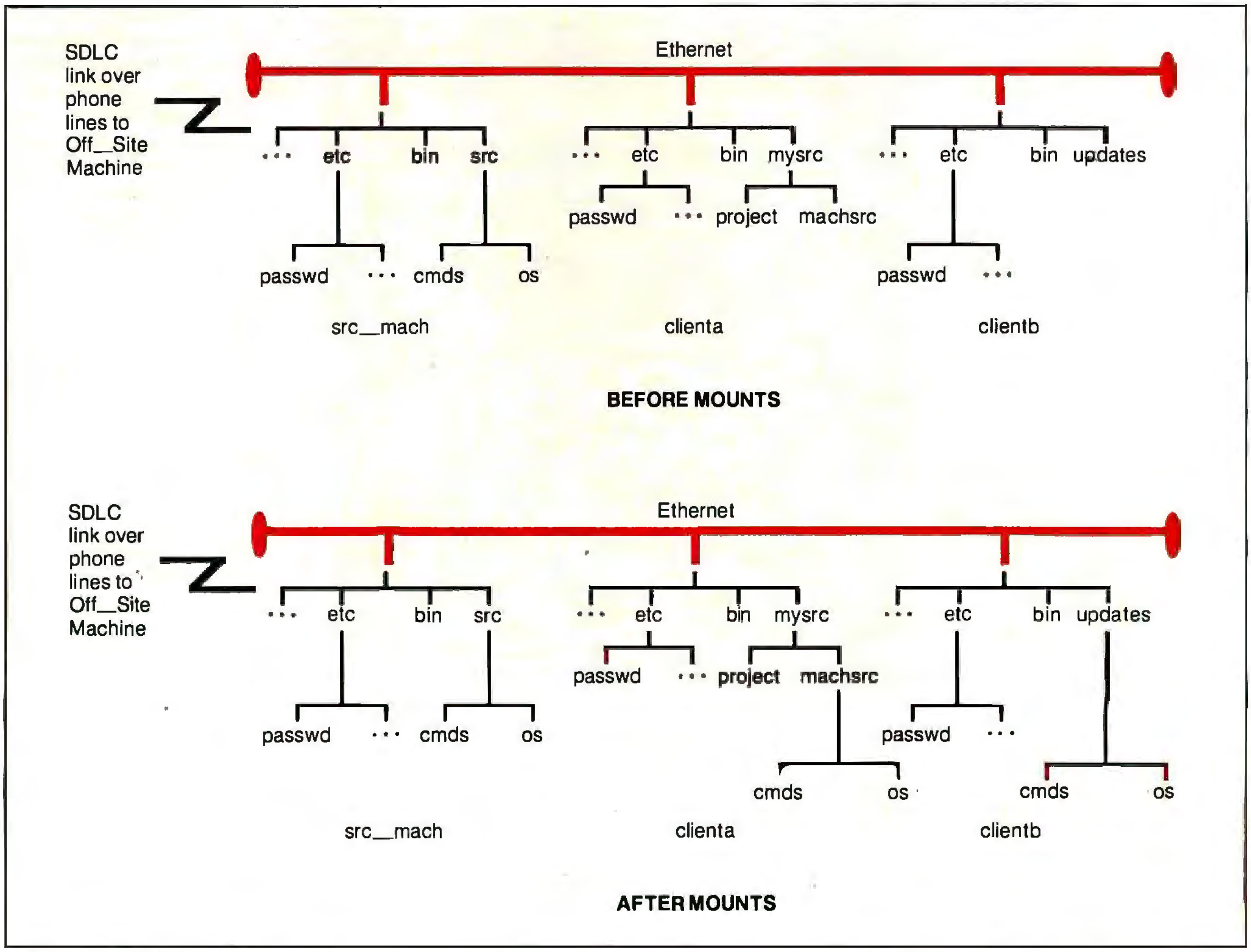


Figure 1: *A four-machine RT PC network before and after users of the network have issued commands shown in table 2. The highlighted lines in the second half of the figure indicate paths that have been changed or added due to the commands.*

the VFS structure describes the file system's characteristics. Likewise, the vnode structure describes the type of entity it is associated with. By accessing files via these higher-level data structures, mounting a foreign file system—such as, a DOS file system—is much simpler because the kernel needn't be aware of that file system's structure.

File Access

With several machines sharing a group of files, it is sometimes the case that files are accessed by users on different machines, with the additional hazard that the network may fail between two or more machines. Suppose that, in our sample network, `clienta` has finished editing a file, `/mysrc/machsrc/cmds/cmp.c`, located on `src_mach`, and the editor is writing the changes back to the file. If the network connection fails during the `write` system call, the client receives an error return from the `write` call in the form of an error message from the editor, and the server (`src_mach`) will close the file, as well as perform other cleanup operations so that the file's contents reflect the last update the system was able to perform. Because the server maintains knowledge about the file's state (i.e., who accessed it last, was it closed improperly, and so on), DS is considered "stateful" in this aspect of its implementation. In contrast, NFS is stateless because servers do not store any state information about remote files.

DS maintains directory caches on client and server so that, in the above example, network access need not occur each time the directories in the path `/mysrc/machsrc/cmds/cmp.c` are resolved.

DS also supports both *read ahead* and *write behind* by maintaining client and server data caches. Data-caching on both client and server optimizes reading and writing to and from files across the network, since caching effects can occur simultaneously on two machines. Since the server is not aware of what files and directories clients have mounted from it, DS is stateless with respect to remote mounts.

The advantage of statefulness is its ability to adhere closely to Unix file-access semantics, particularly in areas such as file locking. However, this also makes it harder to support non-Unix file systems. DS allows remote read and write locks in both enforced and advisory mode.

Network Support

DS uses a finely tuned version of SNA LU6.2 as its network protocol. Because of SNA LU6.2's wide vendor support and its importance as a connectability medium in the IBM architecture world, it has emerged as one of the standard networking mechanisms and is a likely choice for integrating heterogeneous distributed systems. SNA implements a virtual circuit between two machines by providing the transport layer with an error-free channel and ensuring ordered delivery of packets. The transport layer can be Ethernet or synchronous data-link control; although SDLC links are too slow for most purposes, they are useful for using DS across existing media, such as phone lines. In the sample network of figure 1, `src_mach` runs DS over a leased phone line with the machine `Off_Site` to distribute its source code.

A Remote Procedure Call (RPC) layer insulates DS from the network code. IBM calls the RPC layer a virtual circuit interface (VCI), probably because NFS calls its layer RPC (see figure 2). The VCI makes it easier to port DS to a different set of network protocols.

Remote Process Support

Unix System V message queues let processes communicate with each other by reading and writing data to and from lists (queues) of arbitrary data structures (messages). These lists of data are similar to files—they can be created, read, and written—except they reside in memory instead of on disk and are consequently much faster. Each message queue is indexed with a queue id (`qid`) and key that uniquely identifies it. DS extends this idea to let processes on one machine communicate with processes on other machines via message queues.

Each machine maintains a message queue translation table to accomplish the mapping of local message queues to queues on other machines. Table 3 illustrates a translation table that might exist on `src_mach`. `Src_mach` has two real DS message queues, called `clientaq` and `clientbq`, that map into surrogate or stub queues on `clienta` and `clientb`. You can think of the stub queues as remote files; the queues actually exist on `src_mach`, but the user on `clienta` or `clientb` can use the stub queue as if it were a local message queue.

Also, a TCP/IP (Transmission Control Protocol/Internet Protocol) implementation is available under DS that operates concurrently on Ethernet with SNA LU6.2, and this TCP/IP software makes possible another popular form of interprocess communications: sockets.

Security

DS defines a set of network `uids` and `gids` that are translated independently from machine to machine through the use

continued

Table 3: *An IPC message queue translation table for machine* `src_mach`.

Queue name	Local key	Remote key	Remote ID/nickname
clientaq	f777777	f777777	clienta
clientbq	f777776	f777776	clientb

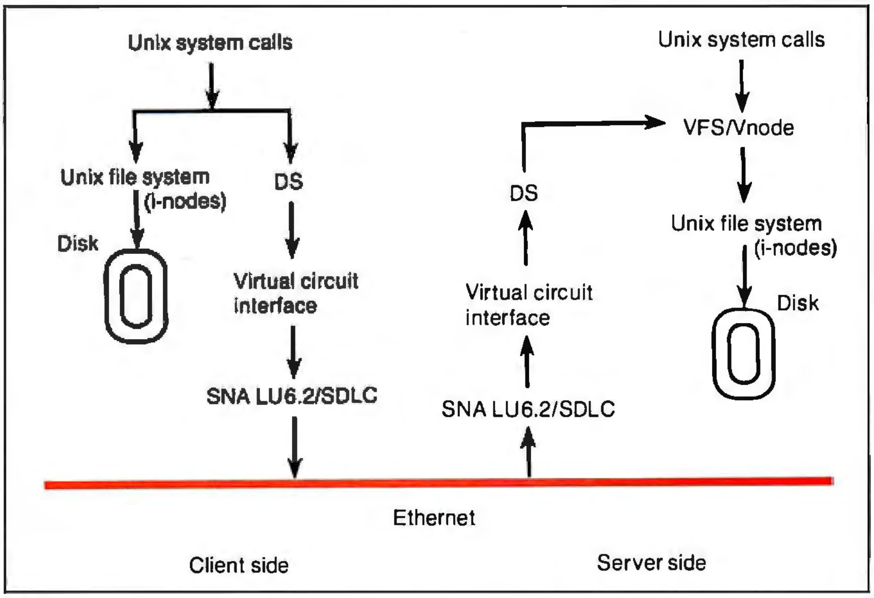


Figure 2: *The structure of the Distributed Services file system.*

Table 4: *A partial* uid/gid *translation table for (a)* clienta *and (b)* src_mach.

(a)

User/Group name	U/G	Local ID	Network ID Outbound	Network ID Inbound	Originating node name/nickname
root	U	0	100	200	src_mach
system	G	0	300	310	src_mach
staff	G	25	111	622	src_mach
larry	U	23	400	410	src_mach
rudy	U	22	400	411	src_mach
clay	U	31	650	660	src_mach
guest	U	8	1000	*	*

(b)

User/Group name	U/G	Local ID	Network ID Outbound	Network ID Inbound	Originating node name/nickname
root	U	0	200	100	clienta
system	G	0	310	300	clienta
clay	U	31	660	650	clienta
bin	G	2	600	111	*
src	U	20	3000	400	clienta
guest	U	8	1000	*	*

Table 5: *A partial node ID table for system* src_mach.

Remote nickname	Remote node ID	Node security	Data link type	Connection profile	Attachment profile
clienta	10811CA1	None	Ethernet	10811CA1	10811CA1
clientb	20FFFFFFF	Secure	Ethernet	20FFFFFFF	20FFFFFFF
Off_Site	20811838	Secure	SDLC	20811838	Slaved

of memory-resident tables on each machine. Table 4 shows translation tables that systems clienta and src_mach might use in our network. A request by user larry leaving clienta is translated to network uid 400, network gid 111 (larry is a member of group staff on clienta). When the request reaches src_mach, the uid/gid pair is translated according to src_mach's tables into user src, group bin, which are then used on src_mach to validate the request.

Maintaining the tables requires effort on the part of the network administrator but has the advantage of allowing greater machine autonomy, since individual machines may vary resource access according to their own preferences.

DS networks may also choose to limit access at the machine level by using the secure node identification service offered in SNA LU6.2. Whenever a session is established between two machines, and at random intervals thereafter, SNA will exchange encrypted passwords to authenticate the remote machine. Each machine in the network has a memory-resident node table listing the other machines in the network that are known to it and whether secure node identification (node security) is desired. Table 5 shows src_mach's node table, indicating that it requires authentication from clientb and Off_Site; a bogus machine masquerading as clienta or Off_Site would have to guess the password before access to DS would be allowed.

Transparency

DS is a true distributed computing environment with transparent access of remote files and message queues. DS represents IBM's first attempt to bridge the Unix-to-Unix communication gap—an area previously filled with 3270 terminal emulators, uucp, and other ancient communication mechanisms.

Although the initial release of DS works only with networks of RT PCs, the choice of industry-standard SNA LU6.2 with its VCI and the VFS/vnode concept of file-system description should make movement toward a heterogeneous computing environment straightforward.

BUY THE NUMBERS.

There's only one way to make sure that you're buying a genuinely high-performance system and that's to evaluate the competition by the real numbers.

And when you compare Tandon's numbers against our major competition there's no doubt who's really selling the systems of the future.

		TARGA 20	PS/2 MODEL 30	TARGA 40 PLUS	PS/2 MODEL 50
1	PROCESSOR:	80286 6/8 Mhz dual speed	8086 8 Mhz	80286 8/10 Mhz dual speed	80286 10 Mhz
2	MEMORY: Standard Memory Management	 1 MB Yes	 640 KB No	 1 MB Yes	 1 MB No
3	DISK STORAGE: Capacity Effective access time	 20 MB 85 ms	 20 MB 85 ms	 40 MB 35 ms	 20 MB 85 ms
4	COMPATIBILITY: 5¼" floppy Runs OS/2	 Yes Yes	 No No	 Yes Yes	 No Yes
5	PRICE:	$1,999	$2,295	$2,995	$3,595

Take the excitingly priced Targa 20. Thanks to its powerful 80286 processor it dramatically outperforms the PS/2 model 30. And it supports Microsoft's OS/2, the operating system standard of the future.

Or put the Targa 40 up against their PS/2 model 50. Double the storage capacity, innovative disk cache technology, and a Tandon low price.

So whether you need a powerful system to help you manage your business, or else a high-performance file server at the heart of your network, the Tandon Targa is the ideal fit.

For more details on the powerful Tandon Targa family call: National 1-800-556-1234 ext. 171, California 1-800-441-2345 ext. 171.

Tandon

Please send me your Tandon Fact Pac, a comprehensive set of literature and product reviews:

Name ______________________

Company ______________________

Address ______________________

City/State/Zip ______________________

Telephone ______________________

Tandon Computer Corporation
405 Science Drive
Moorpark, CA 93021
(805) 378-6104

BYTE IBM

puting environment straightforward.

Network performance is adequate with the original Romp processor card, but it is significantly enhanced with the new Advanced Processor Card, boosting the RT to a 4.5-MIPS machine with improved floating-point capabilities due to the on-board Motorola 68881 math coprocessor. The stateful aspects of DS (directory and data-caching) also enhance network performance and maintain Unix file-access semantics in multiple-access situations. Finally, DS's excellent network security and mounting capability satisfy the fundamental goal of distributed services: A user is more likely to use the remote file capabilities if he or she trusts the distributed-services mechanism. ■

BIBLIOGRAPHY

Bach, Maurice J. *The Design of the Unix Operating System.* Englewood Cliffs, NJ: Prentice-Hall, 1986.

Hatch, Mark J., Michael Katz, and Jim Rees. "AT&T's RFS and Sun's NFS: A Comparison of Heterogeneous Distributed File Systems." *Unix World*, vol. 2, no. 11, 1985.

Sager, G. R., and R. B. Lyon. "Distributed File System Strategies." *Unix Review*, vol. 3, no. 5, 1985.

Sandberg, R., D. Goldberg, S. Kleinman, Dan Walsh, and B. Lyon. "Design and Implementation of the Sun Network File System." *Usenix Conference Proceedings*, Portland, ME, June 1985.

Sauer, Charles H., et al. "RT PC Distributed Services." *IBM Industry Systems Products*, 1986.

Tanenbaum, Andrew S. *Computer Networks.* Englewood Cliffs, NJ: Prentice-Hall, 1981.

Walsh, Dan, Bob Lyon, Gary Sager, J. M. Chang, D. Goldberg, S. Kleinman, T. Lyon, R. Sandberg, and P. Weiss. "Overview of the Sun Network File System." *Usenix Conference Proceedings*, Dallas, TX, January 1985.

Glossary

• File locking: This enables a process to control access by other processes to a region of a file. If a process creates an enforced lock on a section of a file, then other processes attempting to read and write that section will fail. An advisory lock requires more cooperation among processes. If an advisory lock is placed on a region of a file, other processes can still read and write that region freely if they choose; they become aware of the existence of the advisory lock only when and if they attempt to create a lock themselves.

• i-node: In Unix, an i-node is a data structure that resides on disk and contains all the specifics of a file; for example, where the various blocks containing the file are on the disk, how big the file is, its owner, and when the file was last modified. The i-nodes for a given file system are stored in a contiguous region known as the i-list, and an i-node's position in the i-list is given by an i-number. Each entry in the disk's directory consists of a filename and an i-number, so that when you reference a file, the system can locate the i-node defining that particular file.

• `mount`: You execute this Unix command to make a Unix file system available for use, typically on a hard disk storage device. You can partition the hard disk into formatted sections called file systems, and each section is assigned a device filename that resides in the directory `/dev`. To attach a file system to the current directory hierarchy, you issue a `mount` command (e.g., `mount/dev/hd3/usr2/auug`). This example makes available the file system on hard disk partition 3, starting with the path `/usr2/auug`.

• Read ahead: Even though you might read a single character from a file (e.g., using the `getc()` function under Unix), the system reads data from the disk in blocks, where the size of a given block is some multiple of the disk's sector size. The blocks are kept in memory, so that if your program reads another character from the file—often the next sequential character—the operating system can fetch the character from memory rather than read the disk again. Thus, the system "reads ahead."

• `shmat`: This system call attaches a shared memory region to the data segment of an executing application. (The `shmget` system call creates the shared memory segment.)

• SNA LU6.2: SNA (System Network Architecture) was introduced by IBM in 1974 as the specification of the means by which that manufacturer's diverse computer products (primarily mainframes and minicomputers) would be networked. The low-level software modules that actually provide communication services over the network are referred to as LUs (logical units), and LU6.2 is a particular logical unit that is seeing wide use. (LU6.2 has better support of distributed transaction processing than its predecessors. LU6.2 is also known as APPC—Advanced Program-to-Program Communications—for which IBM has recently introduced a PC version.)

• Socket: Used in BSD (Berkeley System Distribution) Unix, sockets are a form of interprocess communication that let tasks talk to one another across a network. A socket is a set of software routines that appears to an application program as one end of a two-way communication path.

• TCP/IP: Transmission Control Protocol/Internet Protocol. Developed by ARPAnet researchers, TCP/IP is a set of communications protocols that allows computers to share resources across a network, and often across dissimilar networks.

• User identifier (`uid`): Every user on a given Unix system is assigned a user identification number (stored in the `/etc/passwd` file, which the system reads at log-in time). This number is attached to any file the user creates, so that the system can keep track of file ownership (particularly important if the user decides to make the file accessible only to himself or herself). Users are also assigned group identification numbers (`gid`), the distinction being that each user has a distinct `uid`, but multiple users can share the same `gid`.

• `uucp`: An acronym for "Unix-to-Unix copy." The command `cp` is the Unix file-copy utility; you can think of `uucp` as a modified version of `cp` that extends the source and destination file path names to include system name prefixes so you can copy files from one system to another in a networked environment.

• Write behind: On many computer systems, you can often type commands faster than the system can execute them; your keyboard input is buffered. Many Unix systems offer a similar feature when writing to files; the system will buffer the data in memory until it has a chance to actually write it to the disk so that the program isn't being held up waiting for a disk.

HAYES ANN TECHNOLOGICAL MODEMS THAT IM

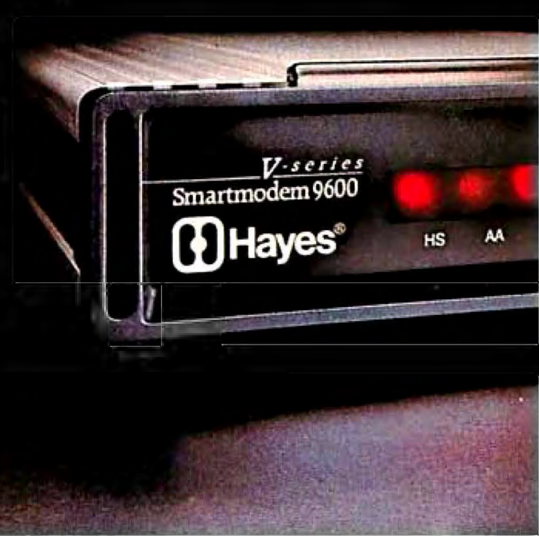
It's long been thought that even the best of technology eventually becomes obsolete. A notion that we at Hayes could never really understand. And certainly never accept. So in defiance of it, we created the V-series Smartmodem 9600™ and V-series Smartmodem 2400.™ Modems that actually get better as they get older. Because they not only incorporate the most intelligent features found in modems today, they also possess the capability to provide a long-term growth path into the communications environments of tomorrow.

V-SERIES SMARTMODEM 9600

This is the fastest modem we've ever made. It can send and receive data at 9600 bps and with adaptive data compression achieve an effective throughput of up to 19,200 bps. Point-to-point error control, forward error correction and data flow control ensure that data gets there accurately.

The V-series Smartmodem 9600 also comes with automatic feature negotiation, a self-operating capability that analyzes all options for modem link and then selects the optimum feature set with any Hayes modem for the most efficient transmission at the highest shared speed.

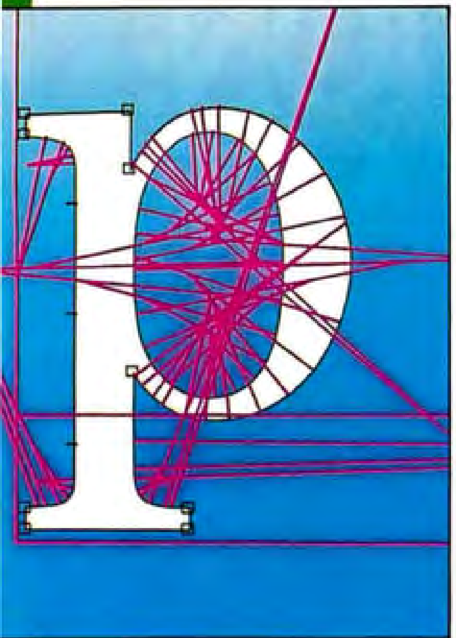
Synchronous and asynchronous communications modes as well as simulated full-duplex employing advanced CCITT V.32 trellis code modulation and fast turnaround ping-pong technology are also part of the package. Plus you'll get the capability to link up with a range of networks, including SNA. And soon V-series technology will offer an X.25 PAD option to further accommodate network environments of today. And the future.



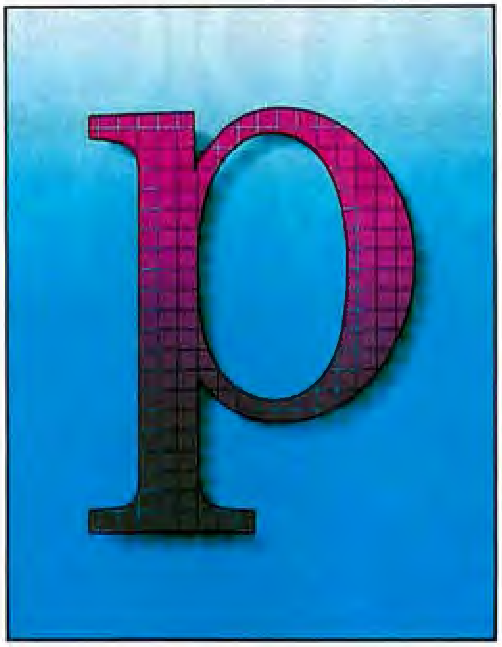
Hayes Microcomputer Products, Inc., P.O. Box 105203, Atlanta, GA 30348.

Circle 97 on Reader Service Card

Take a look at the first typefoundry-quality fonts for IBM users that can be scaled to any size.



Bitstream explains, "Fontware is composed of expert software and intelligent font outlines encoded with typographic rules."



"These rules are used to generate high quality bitmaps in which consistency of stroke weights and other typographically significant features are maintained."

"The essentials of good type design → shape, spacing, weight, rhythm — are preserved at all sizes and resolutions."

SoftCraft has incorporated Bitstream's Fontware™ program into their extensive collection of personal typesetting software. This program utilizes Bitstream® Fontware outlines to produce fonts, giving you an unlimited number of sizes, the very best font quality and a large number of professional typefaces.

Fonts can be produced in any size from 3 to 120 points. The most common sizes can be produced automatically, with no interaction required. Otherwise, simply specify the size, type of printer, font file format, and symbol set (such as *Roman-8*, *IBM-PC*, or *foreign languages*) and the SoftCraft Fontware Installation Program does the rest.

There are currently 80 typefaces available and many more soon to come from Bitstream, including new typefaces specially designed for laser printers.

Font files can be produced in HP soft font format (*portrait* and *landscape*) for use with the many printers and desktop publishing packages that use this standard format. Or produce fonts in SoftCraft format for use with SoftCraft printing programs on laser and dot matrix printers.

Works with any IBM or MS-DOS computer with MS-DOS or PC-DOS 3.1 or later. Requires 512K memory and a hard disk. The SoftCraft Fontware Installation Program is $95 and the Bitstream Outline Packages are $200 for a set of four typefaces.

SoftCraft products let you take full advantage of these fonts.

SoftCraft's Laser Fonts™ program downloads fonts into laser printers and automatically configures Microsoft® Word, Word Perfect and Wordstar 2000® to use the downloaded fonts. Fancy Font® can be used with any program that can produce an ASCII file to print the large fonts produced with the Fontware Installation Program. Fancy Word® works with Microsoft Word to print graphics and fonts up to 120 points. SoftCraft also has an extensive library of low-cost bitmap fonts ($15/disk).

The headlines and copy in this ad were set in Bitstream's version of Times Roman, and printed with Fancy Font on an HP LaserJet+.

SoftCraft, Inc.

16 N. Carroll St., Suite 500, Madison, WI 53703
(608) 257-3300 **1-800-351-0500**

Circle 203 on Reader Service Card (Dealers: 204)

TYPEFACES

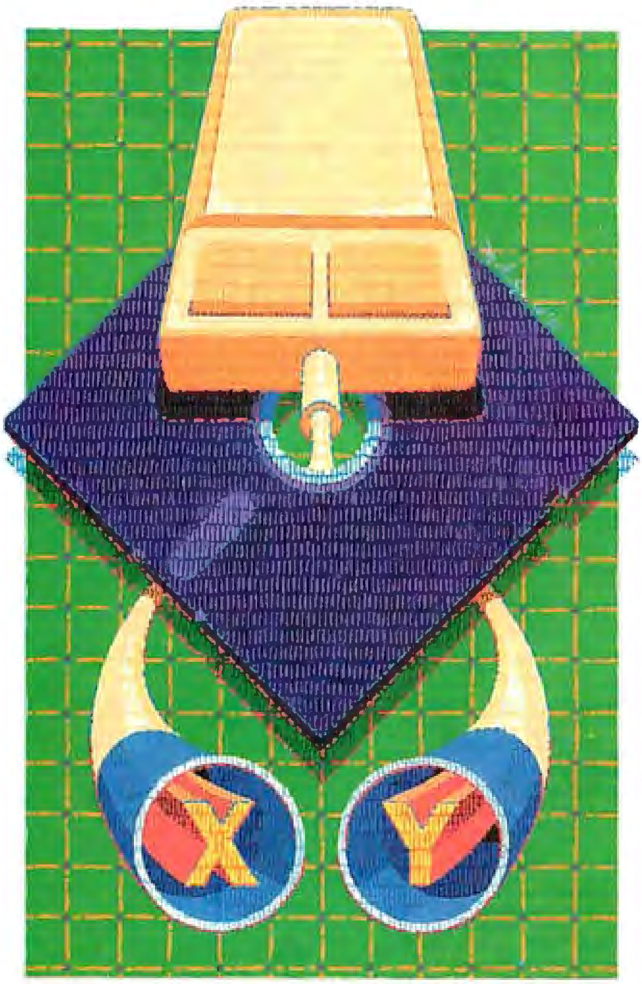
Each Outline Package includes four typefaces, usually roman, italic, bold and bold italic.

- Dutch 801. Bitstream's version of the Times Roman® typeface.
- Swiss 721. Bitstream's version of the Helvetica® typeface.
- Courier
- Headlines: Cloister Black, Bitstream's Cooper Black, Broadway,® the University Roman® typeface.
- Swiss 721 Condensed
- Swiss 721 Light
- Letter Gothic
- Prestige
- Zapf Calligraphic. Bitstream's version of the Palatino® typeface.
- Zapf Humanist. Bitstream's version of the Optima® typeface.
- Century Schoolbook®
- ITC Souvenir®
- ITC Garamond®
- ITC Avant Garde Gothic®
- ITC Galliard®
- ITC Korinna®
- Bitstream Charter™
- Futura® Book
- Futura Medium
- Futura Light

How to put a device-independent interface between your application and input devices

Application Input Drivers

Jeremy Sagan



As window-based user interfaces become more prevalent, learning to incorporate input devices such as mice, joysticks, or graphics tablets into applications is becoming important. Often, the application's writer does not want to limit a program's users to a particular input device from a particular company. Here I will present a technique for isolating the device-dependent details from the device-independent details of writing an input driver, so a programmer can design an application to use various input devices.

Writing an effective application input-device driver requires a minimal set of software subroutines. You must consider all varieties of currently available input devices before finalizing these subroutines.

Take a mouse, for example. It sends data to the computer in the form of relative coordinates, which are the difference between the current coordinates and the previous coordinates. A graphics tablet, on the other hand, sends data in absolute coordinates. An application input driver that supports both types of devices must receive data in either relative or absolute coordinates. If it receives relative coordinates, it is a simple matter to have the application convert absolute coordinates into relative coordinates for it.

I will illustrate this technique with two approaches to writing a mouse driver, but you can extrapolate from that for other input devices, such as a graphics tablet or a joystick. Figure 1 shows the overall strategy. The set of routines that I will use are initialization, return-button status (for up to eight buttons), motion detection, and cleanup. A device-dependent routine translates the hardware characteristics of a particular device to satisfy these routines.

The Microsoft Mouse interface and system calls (see reference 1) have become the de facto standard for interfacing an application to a mouse. They provide a set of functions ranging from mouse position and button status to cursor control. Most mouse manufacturers provide drivers that let their mice work with software originally written for the Microsoft Mouse. The advantage of this standard is that it lets you write applications that will work with most mice; the down side is that this standard is for two-button mice, and some applications might need three or more buttons. If you want to use special features of a mouse beyond the Microsoft standard, you must write directly to the hardware.

I have written two drivers. One takes input from a Microsoft-compatible mouse; the other takes input from a three-button Mouse Systems serial mouse and translates hardware dependencies to the device-independent set of routines that I defined for the application input routine.

I have also written a demonstration program, called MDRIVER.ASM, that shows how an application would use this device-independent interface. You set a flag at the top of the program to determine whether you want to use a Microsoft-compatible mouse (which causes MICROSOFT.ASM to be included), or the Mouse Systems serial mouse (which causes MSYSMOUSE.ASM to be included). Because it is beyond the scope of this article to provide all the code needed to implement the loading of device drivers, I will present the code that assumes that the driver and the applica-

continued

Jeremy Sagan is the director of advanced technology for Business & Professional Software. He can be contacted at 143 Binney St., Cambridge, MA 02142.

Illustration: Mike Hodges © 1987

tions program both reside in the same 64K-byte segment of address space. The input driver would typically be loaded in a different segment, but for simplicity's sake, I have compiled them together and made the entry routine a near procedure.

[Editor's note: *The custom mouse driver shown in listing 1, MSYSMOUSE-.ASM, MICROSOFT.ASM (not shown), and the demonstration code MDRIVER-.ASM (not shown) are available on disk, in print, and on BIX. See the insert card following page 208 for details. Listings are also available on BYTEnet. See page 4. This code is in 8086/88 assembly language, and you will need the Microsoft Macro Assembler to assemble it.*]

Talking to the Hardware

Making calls to the Microsoft driver consists of simply loading registers ax, bx, cx, and dx with values that determine the function call you want, and executing an INT 51, which is the entry point of the Microsoft Mouse driver. Register ax is the function number, and the other registers, if used, are parameters for that function. All the Microsoft Mouse system calls are thoroughly documented in the *Microsoft Mouse User's Guide.* The file MOUSE.COM takes care of the details of communicating with the hardware.

In contrast, the custom mouse driver I wrote interfaces to the mouse at a hardware level by reading data directly from the serial port in the form of a 5-byte packet (see reference 2). The first byte is a combination of five synchronization bits and three bits for button 1, button 2, and button 3, respectively. A clear bit means that the corresponding button is pressed; the device-dependent driver translates this to a set bit for a button press, because this is what the application input driver expects. The next 4 bytes are two updates of the mouse-movement counters, delta x, delta y, delta x, delta y.

These packets of information are transmitted from the mouse to the host computer whenever the mouse detects movement. This is sometimes referred to as "stream mode." If you were to use the polling method for gathering data, your program would either be severely handicapped, capable only of reading mouse data, or it would lose data while it was busy performing other tasks. You must have an interrupt handler execute each time new data becomes available if you want to ensure that your data is not lost.

Figure 2 shows a flowchart of the interrupt service routine (ISR). It reads and processes 1 byte from the serial port, then returns from the interrupt. A packet-counter variable keeps track of the cur-

continued

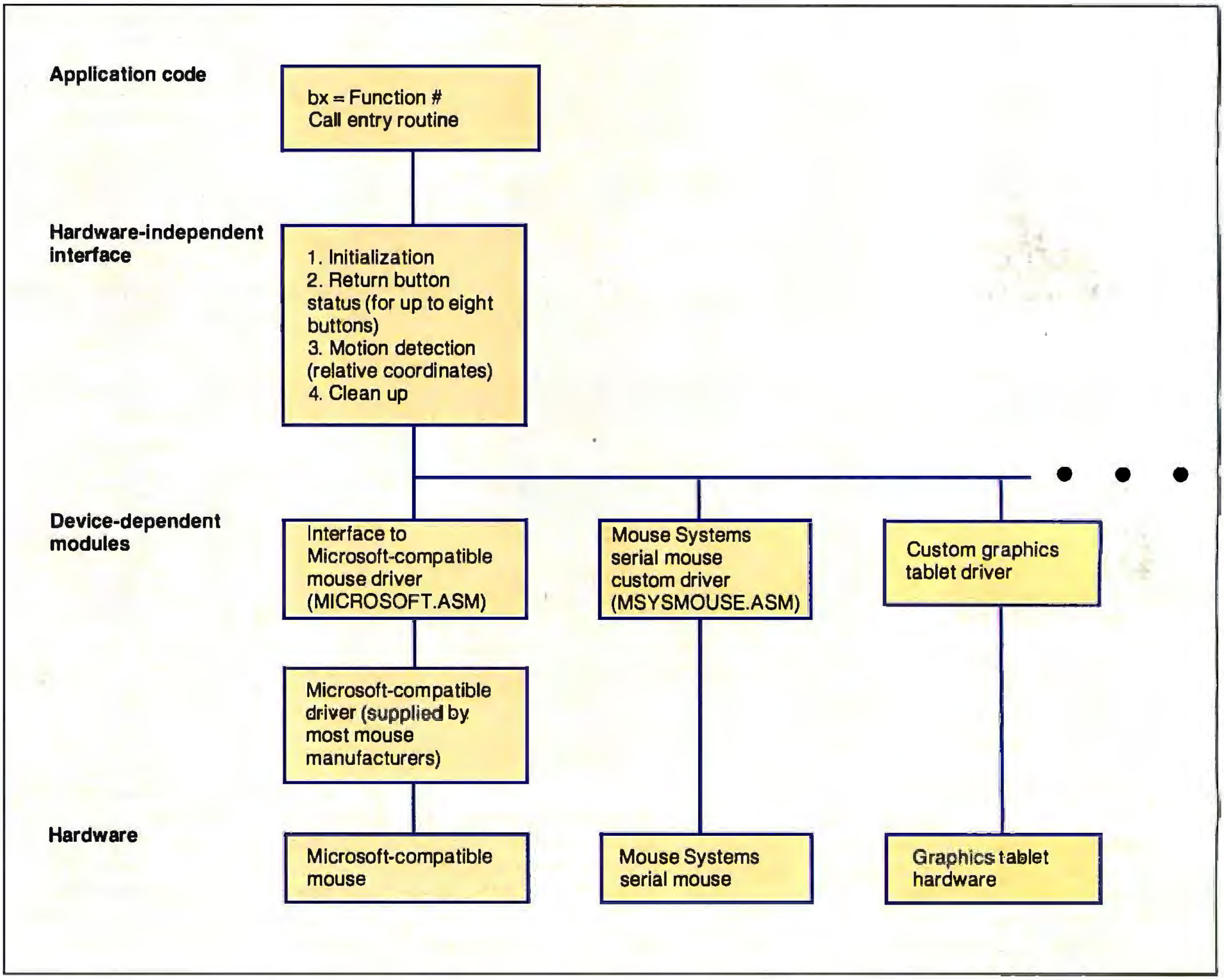


Figure 1: *By placing a device-independent interface between the application code and the input hardware, a programmer can write an application that works with a variety of input devices. The device-dependent code is isolated into modules and translates the hardware requirements into the standard interface. For example, the custom graphics tablet driver can translate absolute coordinates to the relative coordinates that the application is expecting.*

You must have an interrupt handler execute each time new data becomes available if you want to ensure that your data is not lost.

rent position in the packet. A packet is assumed to be "in progress" when the packet counter is not equal to zero. If interrupts are disabled too long, a serial overrun error can occur, which means that the interrupt handler did not have a chance to read the previous byte of data. In the event of a serial error, the ISR sets the packet counter to 0 and returns. If there is no serial error, ISR tests to see if the packet counter is 0 and the byte is the sync byte. This ensures that the routine is synchronized with the mouse. If the counter is not 0, then you accumulate delta values—by adding to delta *x* for an odd packet counter, and by adding to delta *y* for an even packet counter.

The Device-Dependent Routines

Implementing the routines for the Microsoft standard interface is simple. The initialization routine inspects interrupt vector 51 to see if the mouse interface and system calls are resident in memory. If they are, it initializes the mouse using function 0 and sets the variable MOUSEF to the value returned in ax. This value will be negative if the mouse hardware and software are installed.

Function 5 returns the button status, in register ax; bits 0 and 1 represent the left and right buttons, respectively. These bits are set if the corresponding button has been pressed, and cleared if it has been released. On entry to this function, register cx corresponds to the button you are interested in. If cx equals 0 on entry, then on exit a count of the times the left button has been pressed will be returned in bx. Similarly, if bx equals 1, a count of the times the right button was pressed will be returned.

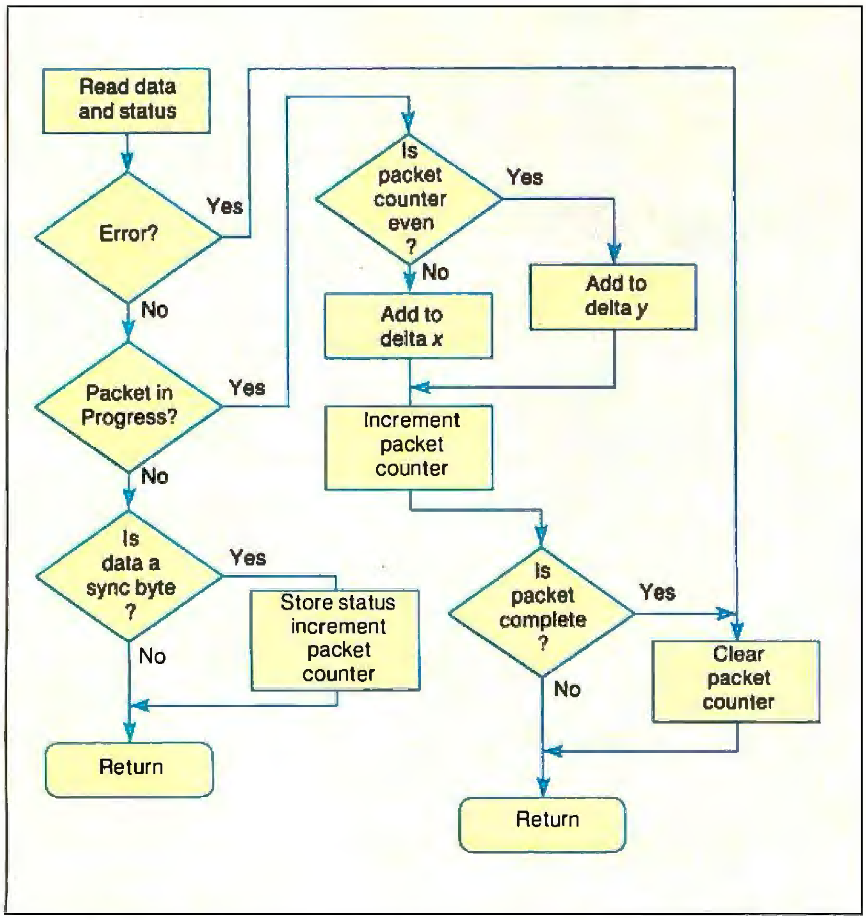


Figure 2: *The flowchart of the interrupt service routine for the Mouse Systems serial-mouse custom driver.*

You use function 11 to determine how far the mouse has moved since the last call to this function. The change in the *x* coordinate is returned in cx; the change in *y* is returned in dx. The motion counter is device-dependent; since Microsoft has two different mice (one at 100 dots per inch and one at 200 dpi), you may have to implement two Microsoft Mouse drivers.

Listing 1 shows the code for the custom Mouse Systems driver. The initialization routine, ISERIAL, is called with the communications-port number in ax (1=com1, 2=com2), and it initializes the serial port to 1200 bits per second. This routine also enables the serial port's data-available interrupt. Some serial cards can be configured for com3 and com4, but since there are only two serial interrupt lines, only com1 or com2 will work with the routines presented here. The button-status routine, which simply reads BSTAT, translates the Mouse Systems button-status bits, to give an on bit for every button pressed.

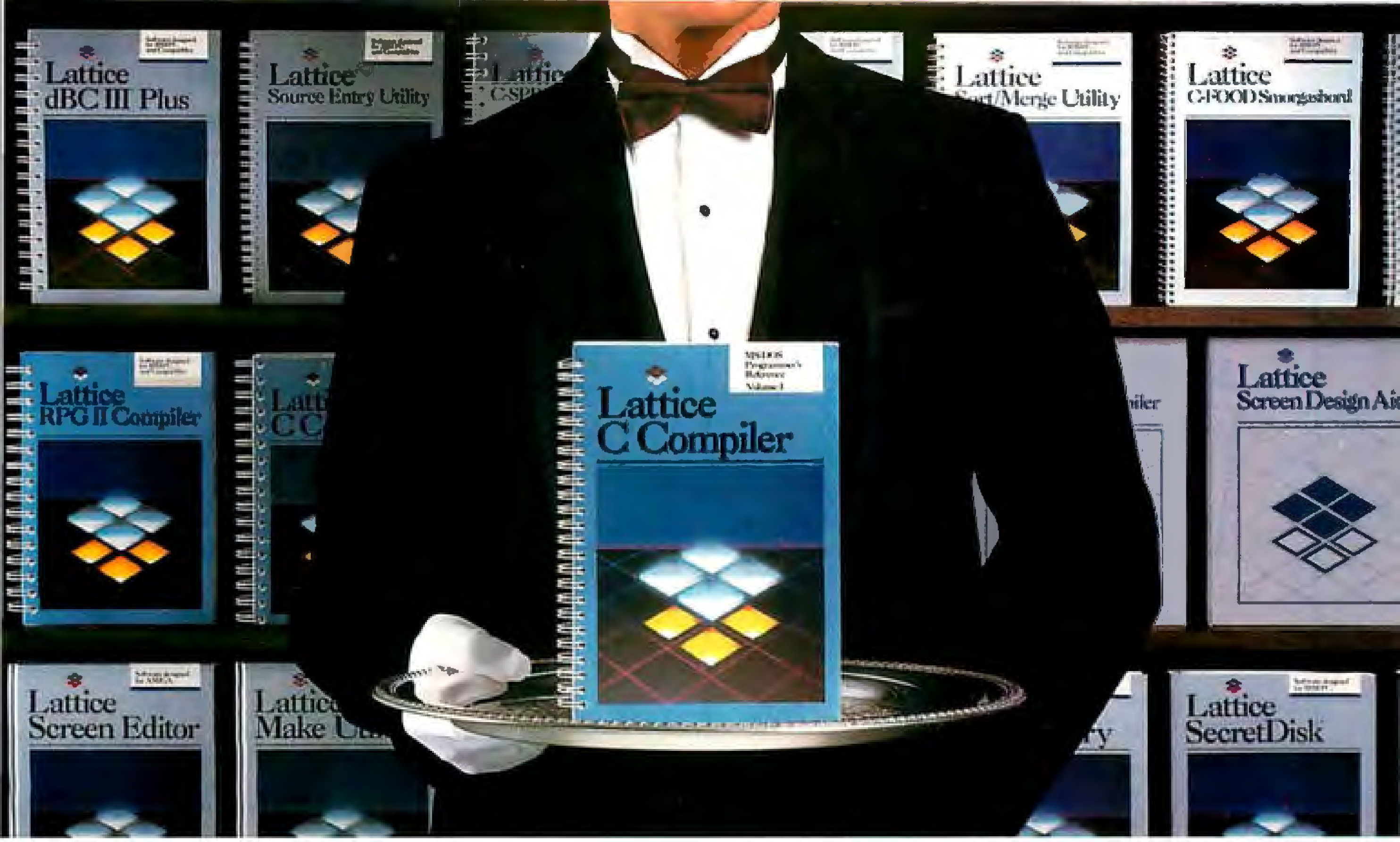
The motion-counter routine is called with the current absolute *x* and *y* values in the ax and cx registers, respectively. If this device dealt exclusively with absolute coordinates, it would need these values to return a relative value to the application. However, since the Mouse Systems mouse uses only relative data, you can ignore these values. The motion-counter routine reads the current delta values and then clears them to 0. The *y* coordinate is negated; I arbitrarily decided I wanted *y* coordinates to move downward as they increased. This function call returns with ax equal to delta *x* and bx equal to delta *y*.

The cleanup routine disables the serial interrupts that the initialization routine enabled. If you don't disable the serial interrupts before you exit to DOS, then the interrupt handler will be overwritten as soon as you load another program. The next time a serial data-available interrupt executes, the CPU jumps into the middle of the newly loaded program, inevitably crashing the computer.

Unfortunately, if your applications program uses any DOS functions that produce a critical error, and if the user selects `abort` instead of `retry` or `ignore` (or presses Control-C and DOS processes it), the result will be the same: a crashed computer. You can circumvent this problem in several ways.

One extremely impractical way is to avoid using DOS calls that process Control-C and to not perform any I/O. A more realistic solution is to load the driver with DOS function-call number 31 hexadecimal, "Terminate Process and Remain Resident," and keep it in memory. A third solution, which I commonly

continued



Our software comes with something no one else can offer.

Lattice Service.

When you join the Lattice family of customers, you'll discover that your software purchase is backed by more than just an excellent warranty. It's backed by unparalleled technical support. By a total commitment to your success and satisfaction. And by Lattice's dedication to excellence in products and services.

Unlike other software manufacturers who charge you for services after you've purchased their product, Lattice offers a unique package of support programs at a price we can all live with—FREE.

Lattice Bulletin Board Service
LBBS is our 24-hour a day bulletin board system that allows you to obtain notification of new releases, general information on Lattice products, and programs for the serious user. And if you've ever experienced the frustration of having to wait a year or more for a new release (that has corrected a bug), you'll really appreciate LBBS. Because with this service, you can actually download the latest program fixes to instantly eliminate any bugs discovered after release.

Technical Support Hotline
Responsible, dependable and capable Support Representatives are only a phone call away. You will talk to a highly skilled expert who is trained to answer any questions you have relating to specific Lattice products. Remember, your complete satisfaction is our goal.

McGraw-Hill BIX™ Network
The Byte Information Exchange (BIX) Network is a dial-in conference system that connects you with a Special Interest Group of Lattice users. The nominal one-time registration fee allows you to BIX-mail your questions—via your modem—directly to Lattice. Or you can post your questions in the conference mode for Lattice or other users to answer. Once again, you have 24-hour access.

You Also Receive:
▪ Timely updates and exciting enhancements ▪ 30-day, money-back guarantee ▪ Lattice Works Newsletter ▪ Technical Bulletins ▪ Access to Lattice User Groups

Lattice has developed more than 50 different Microcomputer software tools that are used by programmers worldwide. We were there for every MS-DOS release. We're there now for OS/2. And we'll be there for the next generation of technical changes. But most of all, Lattice is there for you.

Available through dealers and distributors worldwide.

Lattice
Subsidiary of SAS Institute Inc.

Lattice, Incorporated
2500 S. Highland Avenue
Lombard, IL 60148
Phone: 800/533-3577
In Illinois: 312/916-1600

Circle 121 on Reader Service Card

Listing 1: *The 8088/86 assembly code for the Mouse Systems custom driver, MSYSMOUSE.ASM. The routine ISERIAL installs the interrupt service routine in one of the serial-port interrupt vectors, 0Ch or 0Dh. The application places the function number in bx and issues a call to the entry routine.*

```
;
        ASSUME CS:CSEG, DS:CSEG, ES:NOTHING, SS:NOTHING
;
SERIAL   EQU         14H
MSDOS    EQU         21H
;
;
; This is the main entry point
; all driver routines take the function-call number in BX
;
;           function 0 = initialize mouse
;           function 1 = return button status
;           function 2 = return relative motion
;           function 3 = de-initialize mouse
;           function 4 = return current serial port
;
; Normally this would be a far procedure, but to avoid
; getting into all the intricacies of loading and
; calling drivers, I've converted ENTRY to a near
; procedure and combined it with the sample program.
;
ENTRY   PROC    NEAR
        CLD                        ; Go in the forward
                                   ; direction
        PUSH    DS                 ; Save caller's segment
        PUSH    CS                 ; Make this segment
                                   ; addressable
        POP     DS
        SHL     BX,1               ; Point to routine
        CALL    ROUTINES[BX]       ; and call it through
                                   ; table
        POP     DS                 ; Restore user's segment
        RET                        ; Return far to caller
ENTRY   ENDP
;
        DB      'Mouse Systems',00         ; name
;
ROUTINES LABEL WORD
        DW      ISERIAL        ; function 0 = initialize
                               ; mouse
        DW      BUTTONSTAT     ; function 1 = return
                               ; button status
        DW      MOTIONCOUNT    ; function 2 = return
                               ; relative motion
        DW      DSERIAL        ; function 3 = de-
                               ; initialize mouse
        DW      GSERIAL        ; function 4 = return
                               ; current serial port
        DW      RETADR         ; function 5 = reserved
        DW      RETADR         ; function 6 = reserved
        DW      RETADR         ; function 7 = reserved
;
COMNUM  DW      00             ; com#
;
NEWX    DW      00             ; new x coordinate
NEWY    DW      00             ; new y coordinate
XACCUM  DW      0              ; old x coordinate
YACCUM  DW      0              ; old y coordinate
BSTAT   DB      07H            ; button-status byte
CPORT   DW      03F8H          ; communications-port
                               ; address
PCOUNT  DB      0              ; packet counter
IMSK    DB      0EFH           ; interrupt mask
;
```

```
; This is the heart of the code.
; The serial interrupt handler. This code catches
; serial bytes and maintains a running total of delta x
; and delta y values.
;
ISR:
        STI                     ; Ints back on
        PUSH    AX              ; Save all registers used
        PUSH    BX
        PUSH    DX
        PUSH    DS
        PUSH    CS
        POP     DS              ; Make code segment
                                ; addressable
        MOV     DX,CPORT        ; Get port address
        ADD     DX,5            ; Status
        IN      AL,DX           ; Read status
        MOV     AH,AL           ; Save in AH
        SUB     DX,5            ; Back to data port address
        IN      AL,DX           ; Get byte from port
        AND     AH,01EH         ; Mask error bits of status
        JNZ     ISR3            ; Jmp if error
;
; Jump if an error has occurred on the serial line, most
; likely an overrun error caused by interrupts cleared
; for long periods of time. This will be handled simply
; by clearing the packet counter.
;
ISR2:
        CMP     PCOUNT,0        ; Is this the first byte of
                                ; packet?
        JNE     ISR25           ; No, so accumulate.
        MOV     AH,AL           ; It is the first byte, so
                                ; check certain
        AND     AH,0F8H         ; bits to see if we're in
                                ; sync with the
        CMP     AH,080H         ; data stream. If we're
        JNZ     ISR4            ; not, then we'll just
                                ; return.
        MOV     BSTAT,AL        ; We are in sync, so stuff
                                ; button-status byte.
ISR25:
        MOV     BL,PCOUNT       ; Get packet counter
        INC     PCOUNT          ; Increment for next serial
                                ; interrupt
        OR      BL,BL           ; If it's zero we're done
        JZ      ISR4
;
        CBW                     ; Convert delta byte to
                                ; delta word
        TEST    BL,1            ; Check if odd or even:
                                ; odd = x values
        JZ      ADDY            ; even = y values
        ADD     XACCUM,AX       ; Add to running x
                                ; accumulator
        JMP     SHORT ISR29
ADDY:
        ADD     YACCUM,AX       ; or add to running y
                                ; accumulator
ISR29:
        CMP     BL,4            ; End of packet
        JB      ISR4            ; No
ISR3:
        MOV     PCOUNT,0        ; Yes, so reset packet
                                ; counter
ISR4:
        CLI
        MOV     AL,020H         ; must issue end of interrupt
        OUT     020H,AL
        POP     DS
        POP     DX
```

continued

```
        POP     BX
        POP     AX              ; Restore registers
        IRET                    ; and return from interrupt
;
; This table maps button values into ones appropriate
; for returning to application
;
   MAP DB       07,03,05,01,06,02,04,00
;
   ıis routine returns button status = ax
   1 bit indicates button presses
;
BUTTONSTAT:
        PUSH       BX
        MOV        AL,BSTAT
        AND        AX,7
        MOV        BX,OFFSET        ; Convert to
                   BUTMAP           ; Microsoft format
        XLAT
        POP        BX
RETADR:
        RET
;
   ›tion-count routine
;          on entry:
;  ax=cursor x, cx=cursor y (ignored by this driver)
;          on exit:
;  ax=delta x, bx=delta y
;
MOTIONCOUNT:
        CALL    QREADPACKET ; Read a packet
        MOV     BX,NEWY     ; Return y coordinates
        NEG     BX          ; Positive coordinate
                            ; move down the screen
        MOV     AX,NEWX     ; Return x
        RET
;
   Lean up the serial-port interrupts and masks
;
DSERIAL:
        CLI
        IN      AL,021H     ; Read interrupt mask
        MOV     AH,IMSK     ; Clear appropriate int
        NOT     AH          ; by setting bits
        OR      AL,AH
        OUT     021H,AL     ; Write it out
        MOV     DX,CPORT    ; Get port address
        ADD     DX,3        ; line-control register
        IN      AL,DX       ; Fetch it
        AND     AL,07FH     ; Set low to access
                            ; interrupt-enable
        OUT     DX,AL       ; register
        SUB     DX,2        ; Point at interrupt-enable
                            ; register
        SUB     AL,AL       ; Clear
        OUT     DX,AL       ; it
        ADD     DX,3        ; and clear
        OUT     DX,AL       ; modem-control register
        STI                 ; Finished
        RET
;
GSERIAL:
        MOV     AX,COMNUM   ; returns com#
        INC     AX
        RET
;
   ıis code initializes the Mouse Systems serial mouse
   ; takes the com number in AX (1 = com1, 2 = com2)
;
   RIAL PROC      NEAR
        PUSH      CX
```

continued

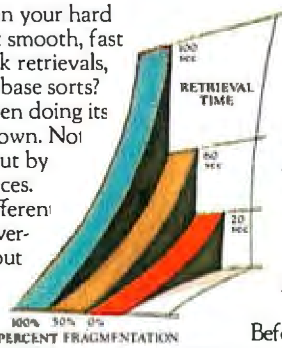
Get Disk Optimizer.™ Your hard disk will run faster when it's not chasing around after files.

NEW! MORE POWERFUL

It's the safest, most effective way to put the zip back in hard disk performance.

Remember the old days when your hard drive was new? Remember that smooth, fast slick performance? Those quick retrievals, rapid saves, lightning-like database sorts?

Well ever since, DOS has been doing its best to slow your hard drive down. Not by slowing down the motor, but by breaking your files up into pieces. Storing different chunks in different places. Data files, programs, overlays and batches that started out in one seamless piece are now scattered all over.



The more fragmented your files get, the longer they take to retrieve.

Loading is slower.
Sorting is slower.
Retrieving, backing-up.
Everything takes longer because your disk has to work harder.

Problem is, it's something that happens so gradually you may not notice the difference. At least, not until you see the dramatic improvement after using Disk Optimizer.

File fragmentation—It's a problem you can see.

Watch your hard drive the next time it reads or writes a file. Each "blip" of the LED means the drive-head is moving to another place on the disk—either to pick up or lay down another chunk of data.

And the truth is, head movement takes time. Far more time than actual reading and writing. What's worse, all this head movement causes extra wear and tear that can shorten the life of your drive.

Disk Optimizer—Tunes up your disk by cleaning up your files.

Disk Optimizer works by finding all the scattered pieces of your files and putting them back together where they belong. Next time your drive reads it, there's just one place to look.

And the results are often dramatic. Reading and writing times may be cut by as much as two thirds. Database sorts that used to take hundreds of head moves now proceed quickly and efficiently. And since head movement is now at an absolute minimum, your disk drive will lead a longer, more productive life.

Analyze, scrutinize, optimize.

Before you optimize, you'll probably want to analyze. So Disk Optimizer shows you, in percentages, how much fragmentation has taken place—on the entire disk, in individual directories, or for groups of files you specify using global or wildcard names.

Plus, there's built-in data security that lets you assign passwords to as many files or file groups as you want.

And the File Peeker gives you an inside look at the structure of files. It's a great way for non-programmers to learn more about computers, and a powerful tool for professionals who want to analyze the contents of their disks.

Get your hard disk back in shape—with new improved Disk Optimizer.

Hard to believe, but the new Disk Optimizer is even better than before. Not only will it optimize your disks in far less time than it used to, but it actually speeds up retrievals even more by letting you give priority treatment to your most used files, like programs and batches.

When you think about it, it's simple. The longer you own your hard drive, the more you come to depend on it. But the longer you wait to get Disk Optimizer, the less performance you'll get.

Use it just once and discover what thousands of satisfied PC users already know—$59.95 is a small price to pay to restore the speed and performance you count on.

Buy Disk Optimizer at better computer dealers everywhere, or order today by calling SoftLogic Solutions at 800-272-9900 (603-627-9900 in NH).

Disk Optimizer $59^{95}

Here's what people are saying about Disk Optimizer.

"I feel this is essential software for anyone with a hard drive."
-D. Sorenson, TEXACO, California

"I find Disk Optimizer excellent. Disk access–even at 4.77MHz–is much faster. And loading 100+ Fontasy fonts now takes seconds."
-Elizabeth Joan Doyle, Indiana

"Disk Optimizer...(has) found a permanent place on my software shelf."
-Stephen R. Davis, PC MAGAZINE

SoftLogic Solutions, Inc.
One Perimeter Road
Manchester, NH 03103
800-272-9900
(603-627-9900 in NH)

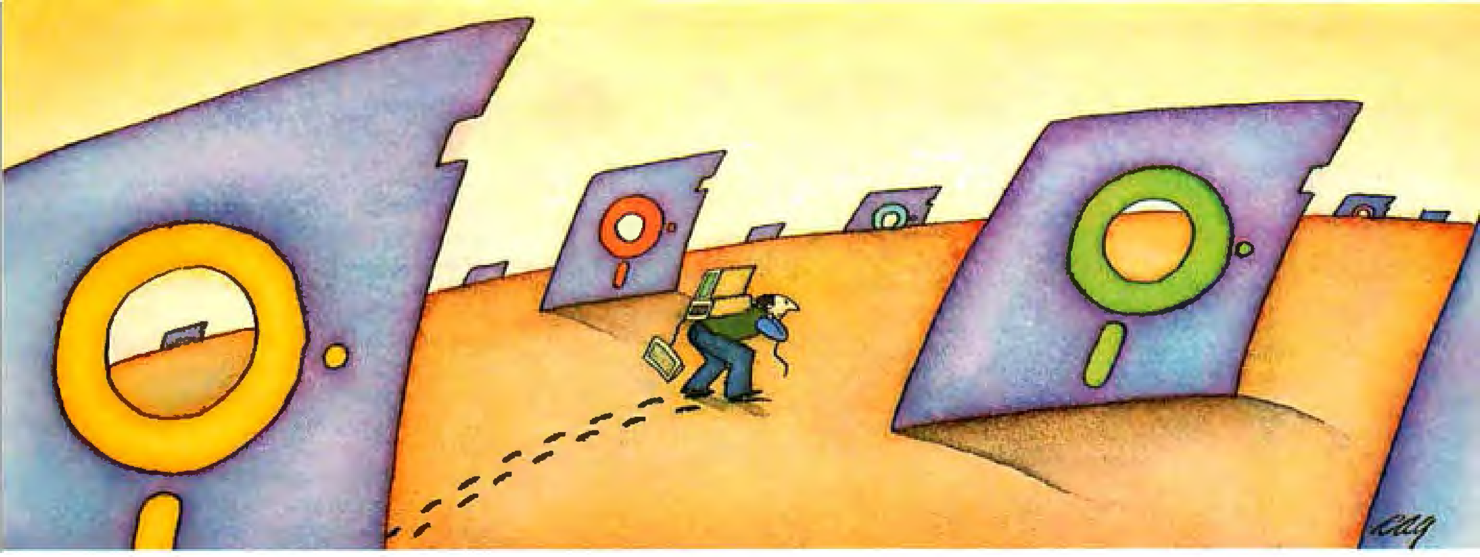
SOFTLOGIC SOLUTIONS

#64

Call today:800-272-9900

```
        DEC     AX
        MOV     COMNUM,AX       ; Save com#
        MOV     DX,AX
        MOV     AX,087H         ; 1200 bps
        INT     SERIAL          ; Let BIOS initialize
                                ; data-transfer rate
                                ; and stuff
;
        PUSH    BX
        PUSH    DX
        PUSH    CS
        MOV     AX,040H         ; point at BIOS data
                                ; segment
        MOV     DS,AX
        MOV     BX,DX
        SHL     BX,1
        MOV     DX,ZERO[BX]     ; Get port address at
                                ; 40:0 or 40:2
        POP     DS
        MOV     CPORT,DX        ; Save it
        CLI
        MOV     DX,OFFSET ISR   ; Stick the serial
                                ; interrupt handler
        MOV     AH,025H         ; in either
        MOV     AL,00CH         ; Int 0CH
        SUB     AL,BYTE PTR COMNUM  ; or 0BH
        INT     MSDOS
;
        IN      AL,021H         ; Mask the interrupt
                                ; controller
        MOV     AH,0EFH
        CMP     COMNUM,0
        JZ      ISERIAL2
        MOV     AH,0F7H
ISERIAL2:
        MOV     IMSK,AH         ; Save for later
        AND     AL,AH
        OUT     021H,AL
;
        MOV     DX,CPORT        ; Get port address
        ADD     DX,3            ; line-control
                                ; register
        IN      AL,DX
        AND     AL,07FH         ; Make interrupt-
                                ; enable register
        OUT     DX,AL           ; addressable
        SUB     DX,2            ; Point at it
        MOV     AL,1            ; Set the data-
                                ; available int
        OUT     DX,AL
        ADD     DX,3            ; modem-control
                                ; register
        MOV     AL,09           ; Set it
        OUT     DX,AL
        STI
        POP     DX
        POP     BX
        POP     CX
        MOV     AL,-1           ; mouse available.
        RET
ISERIAL ENDP
;
;
QREADPACKET:
QR1:
        CLI
```

continued



PROBLEM: There's just no easy way to move from one software program to another.

THE SOFTLOGIC SOLUTION: Software Carousel™

Now you can keep up to 10 programs loaded and ready to run.

Hard to believe, but some people are happy with just one kind of PC software. Well, this is not a product for them.

But if you're someone who depends on many packages, all the time—someone who'd use several programs at once if you could, well now you can. With Software Carousel.

Why call it "Software Carousel"?

In some ways, Software Carousel works like the slide projector you're used to. You load a handful of pictures, view one at a time, then quickly switch to another. A simple idea, with powerful possibilities for computing.

Here's how it works. When you start Software Carousel, just tell it how much memory you have, load your software and go to work.

Need to crunch numbers? Switch to your spreadsheet. Need your word processor? Don't bother saving your spreadsheet file. Just whip over to your document and do your work. Snap back to your spreadsheet, and it's just like you left it.

With up to ten different programs at your fingertips, you'll have instant access to your database, communications, spelling checker, spreadsheet, word processor, RAM resident utilities, languages, anything you like.

Reach deep into expanded memory.

This could be the best reason ever for owning an expanded memory card, like the Intel Above Board, AST RAMpage, or any card compatible with the L/I/M Extended Memory Standard.

Software Carousel puts programs into this "high-end" memory for temporary storage when they're not in use. And switches them back out when you want them. It's fast, efficient, and easy.

With Software Carousel running in RAM, you can load a program and retrieve a file up to 15 times faster. Test conducted on an IBM XT.

If you want, Software Carousel will even use your hard drive for swapping. Just allocate a portion for storage, and go to work.

Sidekick, Superkey and Ready. All at the same time.

You know what happens if you try loading two or more RAM resident utilities at once. You get crashed keyboards, frozen screens, all kinds of interference between programs fighting for control.

With Software Carousel, you can have as many accessories and utilities on-tap as you want. Just load different ones in different Carousel partitions. Since they can't see each other, they can't fight.

The easy way to maximize PC power.

With all this power, you might think Software Carousel is complicated and difficult to use. Not so. Particularly when you compare it to other programs like Windows and TopView.

Set it up once and Software Carousel remembers forever. Better still, Carousel will look for the programs you use most often and optimize them for quickest access.

The way we see it, there are certain things you have a right to expect from your computer. Access to your software is one of them. At just $59.95* a copy, Software Carousel is the best way to get it.

See Software Carousel at better computer dealers everywhere. Or order today by calling 800-272-9900 (603-627-9900 in NH), or mail the coupon below. If you find this product does not meet our advertised claims within 30 days of purchase we will arrange for a prompt refund.

Software Carousel $59.95*

YES! Please send me ______ copies of Software Carousel at just $59.95* each (not copy protected).

Name ______

Company ______

Address ______

City ______ State/Zip ______

Che ISA ☐ MC ☐ AMEX ☐

Card # ______ Exp. Date ______

Signature ______

SoftLogic Solutions, Inc.
One Perimeter Road
Manchester, NH 03103
800-272-9900
(603-627-9900 in NH)

SOFTLOGIC SOLUTIONS

#64

Call today: 800-272-9900

*plus $5.00 shipping and handling.

Circle 207 on Reader Service Card

```
;
        NOP
        MOV     AX,XACCUM       ; Get x accumulator
        MOV     NEWX,AX         ; Move to new delta x
        MOV     AX,YACCUM       ; Get y
        MOV     NEWY,AX         ; to new y
        SUB     BX,BX
        MOV     XACCUM,BX       ; Clear
        MOV     YACCUM,BX       ; accumulators
;
        STI
QREXIT:
        RET
;
```

use, is to insert a section of code in the termination procedure at interrupt 22h. DOS passes control to this code (via a far jump) before it returns to a parent procedure. The only way I've been able to obtain control is by placing the address of my termination routine into 0Ah and 0Ch of the program prefix, in addition to INT 22h. After the necessary cleaning up, the termination routine should exit by issuing a far jump to the original contents of addresses 0Ah and 0Ch of the program segment prefix.

In addition to the basic input-driver design, several other functions might be desirable. Different devices have different resolutions; it's up to your driver to scale the *x,y* coordinates into something consistent—for example, 100 dpi. One useful feature is nonlinear motion detection; with this feature, if you move the mouse quickly, the cursor will move a lot farther than if you move the mouse the same distance slowly.

You also might want to add a routine to determine the number of times a particular button has been pressed since the last call to this function. Such a routine might be necessary if your program is too busy to process button-status information immediately. A possible addition to the interrupt handler compares the current button-status information to BSTAT, and increments individual button-press counters appropriately before storing the current button-status information in BSTAT.

Inevitably, you have to write software that interfaces directly to hardware (i.e., that is hardware-dependent). To retain your main application's hardware independence, you should separate the hardware-dependent code into modules or drivers that can be easily replaced. Although this involves considerable effort, you save a substantial amount of time in the long run. Furthermore, you are ensuring device independence, as well as preparing your applications program for continuing expandability. ■

REFERENCES

1. *Microsoft Mouse User's Guide.* Redmond, WA: Microsoft Corp., 1986.. *PC Mouse Reference Manual 5.00.* Santa Clara, CA: Mouse Systems Corp., June 1986.

New!

GŌfer™

"Another miracle from Microlytics!" —Jerry Pournelle

Pops Up,

Searches
and *finds* anything on disk

Inserts

—Instantly!

"The most exciting and essential add-on since Sidekick."—Tim Bajarin, *Creative Strategies*

You hear yourself thinking: "I know it's here somewhere . . . but where?"

Now you can let GOfer go for it.

Stay in the document and file you're working on.

Pop up GOfer with one keystroke. Send him for an old memo, a phrase, a fact, a date, a number.

He'll whiz through floppy and hard disks and zip back to you on the double.

Then GOfer helps you use what he found. Insert it. Or print it. Or store it. Instantly.

GOfer needs no housekeeping, no indexing, no file conversion. With GOfer, *nothing* gets lost.

Anatomy of a GOfer

• works with IBM PC's & compatibles and PS/2's. • works with all leading word processors & utilities. • searches are extremely fast and free-form (NO INDEXING), based on simple AND/OR/NOT (Boolean) logic, and "how close?" parameters. • *requires 79K of RAM.* • *not copy-protected.*

~~$79.95~~ **$59.95** special introductory price

Get your GOfer

Now at dealers nationwide or order direct by calling **1-800-828-6293** (in NY, 716/377-0130)
C.O.D.'s and credit cards welcome
30-day money-back guarantee

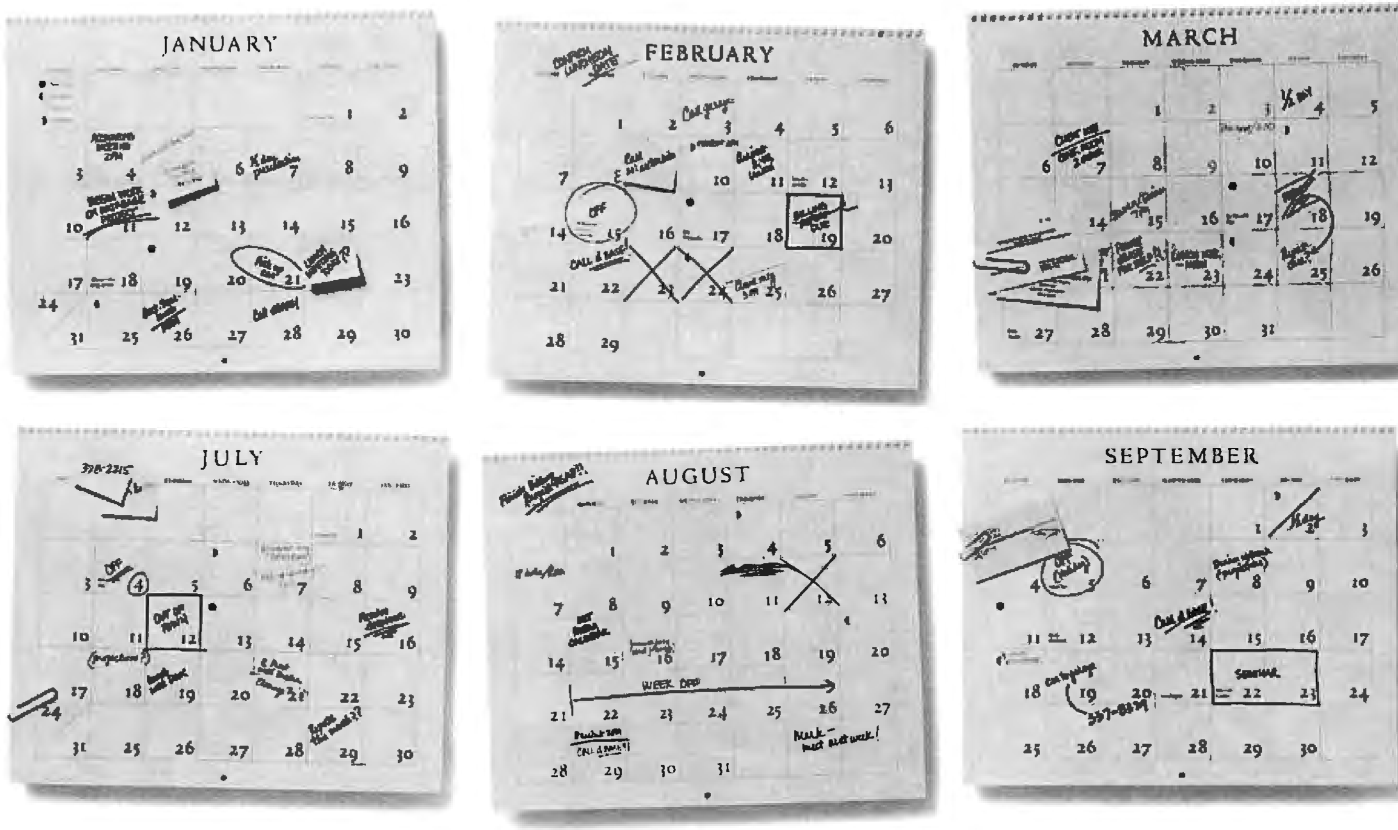
GOfer makes molehills out of mountains.

Microlytics, Inc. 300 Main St., Suite 538, East Rochester, NY 14445

GOfer™ is the Text Finder™ utility from Microlytics™, the makers of Word Finder™.

Circle 135 on Reader Service Card

"I FLIPPED TH
OF PAGES TRYING TO



Choosing the wrong database management software can be like doing time.

Ask Harry Viens, the executive vice-president of an advertising agency.

Using a copy of dBASE,® he spent a year trying to learn its programming language to develop applications on the agency's PC to track income against forecasts.

But with dBASE, all he ended up with was wishing he'd used R:BASE® System V, the database software you don't have to be a programmer to use.

"AFTER USING dBASE I HAD TO BE DEPROGRAMMED."

Viens knew his mini-computer was fine for bookkeeping, but wasn't capable of turning around management reports—critical for making quick decisions.

"We've got to make decisions in real time, not wait a couple of months to find out what's happening today," Viens said.

"I tried to build this dBASE application nights, weekends and during holes in my schedule. But almost a year later, the miserable thing still wasn't done," Viens told us.

"The language is unnatural. I guess some people love writing code. For me, that's like trying to mow the lawn with nail clippers."

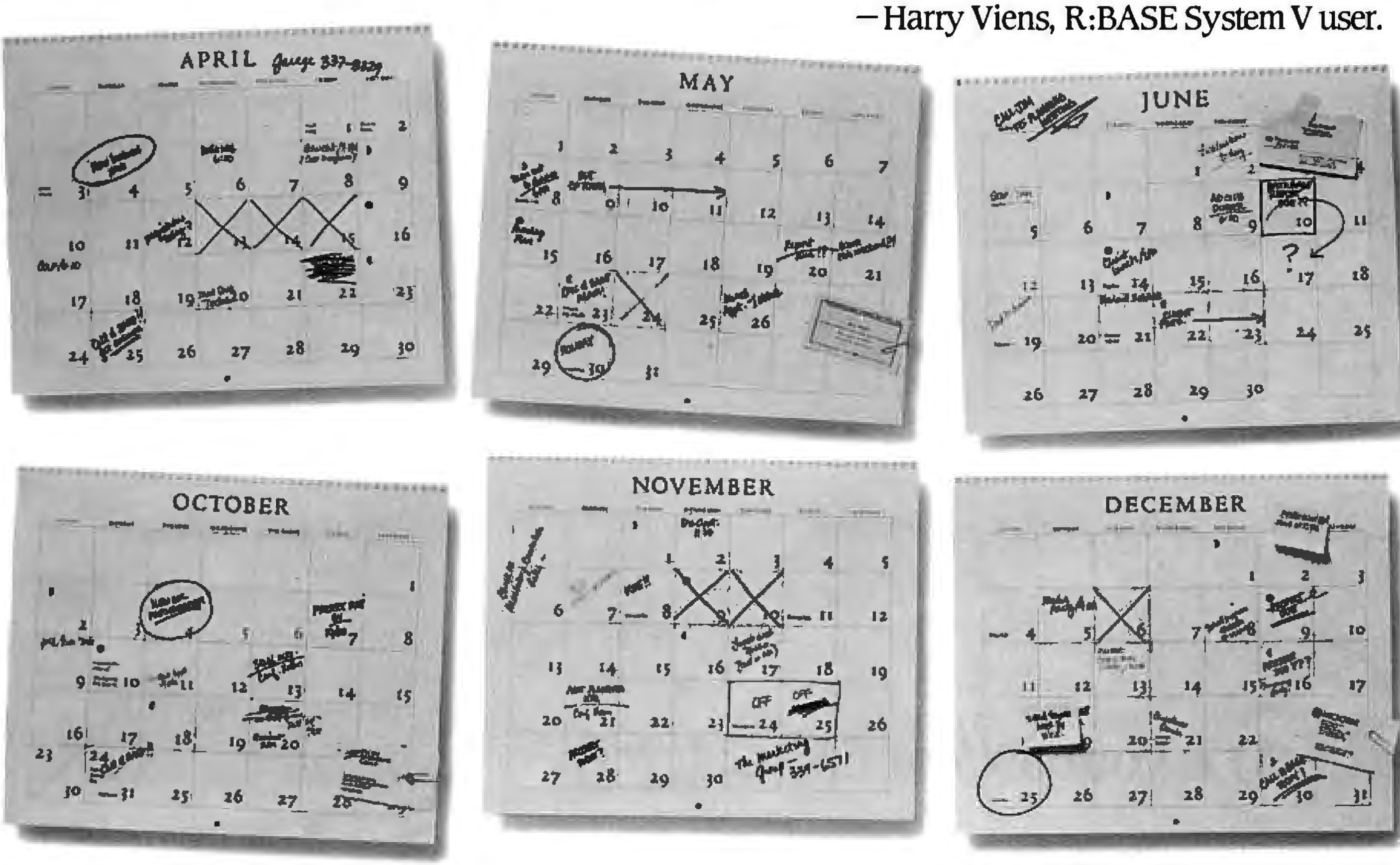
"I SAW, AND I BELIEVED."

Being an advertising person, Viens actually reads ads. He saw our claim that you don't have to be a programmer to use R:BASE, and bought it.

Because of our EXPRESS System, he was

ROUGH A LOT FIGURE OUT dBASE."

—Harry Viens, R:BASE System V user.



able to build the kind of application you'd expect in a custom program from a professional programmer. All without having to write one line of code.

"It took me just three days to do it working part-time," Harry said. So now he can track his forecasts and break them out by the account person who made them.

"R:BASE System V is an amazing product. Despite its power you never seem to get trapped in a corner you can't get out of."

CALL 1-800-624-0810, DEPT. BY1087 FOR A TRIAL PACK.

Lots of people like Harry Viens have been saving time with R:BASE System V.

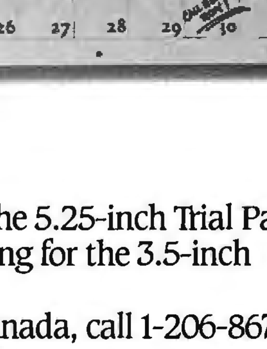
Discover the difference R:BASE can make. Call for our fully-functional Trial Pack. It's just $19.95 plus shipping for the 5.25-inch Trial Pack. Or just $24.95 plus shipping for the 3.5-inch Trial Pack.

From Alaska and Canada, call 1-206-867-1800, Department BY1087.

We can also put you in touch with an authorized Microrim dealer who will take you through the software firsthand.

It will be time well-spent.

R:BASE SYSTEM V
MICRORIM



InfoWorld's Overall MS-DOS Software Product of the Year.

The 5.25-inch formats run on IBM PC, XT, AT and 100% compatibles, and on all major LANS, with no additional cost for extra users attached to the server. The 3.5-inch formats run on IBM PS/2 and other 100% MS-DOS compatibles. Trademarks/Owners: Microrim, R:BASE/Microrim, Inc.; IBM, PS/2/International Business Machines, Inc.; dBASE/Ashton-Tate Corporation; InfoWorld/CW Communications, Inc. © 1987 Microrim.

Circle 138 on Reader Service Card

CLEO is your SNA or BSC Gateway

Remote Sites Communication

Whatever your industry, your remote computers need to share information with your mainframe. Or, they need to exchange data with other remotes. In either case, you need a total solution at the remote sites. You need software, hardware interfaces and modems that all work together smoothly. You need CLEO!

CLEO software products allow micro-computers to communicate with mini-computers and mainframes, and to emulate their workstations. Since 1981, CLEO has provided remote communications between micros and mainframes for the automotive, insurance, medical and banking industries. Today over 44,000 CLEO users worldwide are running on all major brands of micro-processors. The greatest number of these users run CLEO software on IBM Personal Computers and NETBIOS LANs.

Complete Software/Hardware Package

Every CLEO package contains all the soft-ware and hardware accessories needed at the remote site. Your selected CLEO SNA or BSC software is packaged with 1) an internal modem card for dial-up applications, or 2) an interface card and cable for use with your existing modem. There's no waiting for non-CLEO add-ons. And, you get prompt, single-source service.

Package prices range from $795.00 for most stand-alone packages, up to $2,995.00 for the 32-user SNA gateway.

Call us today to discuss your application.

CLEO Software
1639 North Alpine Rd.
Rockford, IL 61107
Telex 703639

Headquarters:
USA: **1-800/233-2536**
Illinois: 1-800/422-2536
International: 815/397-8110

Sales and Distribution:
Benelux, Scandinavia: 31 (71) 899202
Canada, East: 800/361-3185
Canada, West: 800/361-1210
Canada, Montreal: 514/737-3631
Colombia, S.A.: 12875492
Denmark: 451 628300
Italy: (0331) 634 562
Mexico City: 203-0444

CLEO®

Increase your productivity by adding simple assembly language programs to DOS batch files

Better Batch Files Through Assembly Language

William J. Claff

One simple and easy-to-use tool in PC-DOS for increasing productivity is the *batch file*, a text file that executes its contents as if you had typed in the individual lines from the keyboard. This article will show you how to write more versatile batch files by adding small assembly language programs that become new batch-file "commands." These programs let you influence what the batch file does while the file is executing. Using the programs, you can write batch files that recover from errors and let the user interact with many programs through a menu system.

You don't need to be familiar with assembly language to understand how to use these programs in batch files. However, if you are already familiar with assembly language, you will be able to modify the programs and write similar ones suited to your specific needs. [Editor's note: *EXKEY.BAT, KEYIN.ASM, TOPATH.ASM, ONEKEY.ASM, and STRING.ASM are available on disk, in print, and on BIX; see the insert card following page 208 for details. Listings are also available on BYTEnet; see page 4. Assembled versions of the files are available for IBM PC-compatible computers on BIX and BYTEnet only.*]

The Missing Link

In their simplest form, batch files save typing: COMMAND.COM reads characters from the batch file as if they were keyboard inputs. (COMMAND.COM is the command processor with which you interact when you type in a command at the A> prompt.) With the use of replaceable parameters (variables within a batch file whose values you type in after the batch-file name) and the batch-file-oriented PC-DOS resident commands ECHO, FOR, GOTO, IF, PAUSE, REM, and SHIFT, batch files have the power of a simple programming language.

A batch file cannot read the keyboard and assign the value read into a variable. The PAUSE command will wait for a keystroke, but the batch file has no way of knowing which key you pressed. You can remedy this deficiency by using Debug to create the following program, which I call "the missing link." Just type in everything that is italicized, pressing Enter at the end of each line. The value xxxx is not important and will vary depending on your computer's configuration. This recipe creates KEY.COM, an 8-byte assembly language program.

```
A>DEBUG
-A
xxxx:0100 MOV AH,0
xxxx:0102 INT 16h
xxxx:0104 MOV AH,4C
xxxx:0106 INT 21h
xxxx:0108
-RCX
CX 0000
8
-NKEY.COM
-W
Writing 0008 bytes
-Q
A>
```

KEY uses the BIOS's keyboard function to wait for a keystroke. When you press a key, this program reads the next keystroke (by executing INT 16h) and ends, returning its ASCII value by executing the DOS function to terminate a process (DOS function

continued

William J. Claff (7 Roberts Rd., Wellesley, MA 02181) is a senior software engineer with SoftSet Associates Inc. He holds an M.S. in applied mathematics from Harvard University, publishes an IBM technical newsletter, and is the leader of the IBM PC Technical Subgroup of the Boston Computer Society.

Assembling KEYIN

All the programs in this article are .COM files. To assemble them and produce a usable program, I use a batch file called MAKECOM.BAT. [Editor's note: *MAKECOM.BAT is available on disk, in print, and on BIX; see the insert card following page 208. It is also available on BYTEnet; see page 4.*] At a minimum, a MAKECOM.BAT file should contain:

```
masm %1;
link %1;
exe2bin %1 %1.com
```

To assemble KEYIN.ASM, enter

```
make com keyin
```

Ignore the linker's stack segment message; .COM files can't have one.

Listing 1: *Using* KEY *in batch files. This file, EXKEY.BAT, uses* KEY *to get a keystroke and* ERRORLEVEL *to hold the returned value. The* KEYIN *program, a more sophisticated version of* KEY, *is defined in listing 2.*

```
ECHO OFF
:prompt
CLS
ECHO Press 1    for program A
ECHO Press 2    for program B
ECHO Press Esc  to exit to DOS
:getkey
key
IF ERRORLEVEL 49 IF NOT ERRORLEVEL 50 GOTO a
IF ERRORLEVEL 50 IF NOT ERRORLEVEL 51 GOTO b
IF ERRORLEVEL 27 IF NOT ERRORLEVEL 28 GOTO exit
GOTO getkey
:a
ECHO Execute program A
PAUSE
GOTO prompt
:b
ECHO Execute program B
PAUSE
GOTO prompt
:exit
```

Listing 2: *KEYIN.ASM. This program reads the next keystroke and returns its ASCII value in the* ERRORLEVEL *variable.*

```
CODE    SEGMENT
        ASSUME  CS:CODE         ;<- ASSUMES FOR .COM FILE
        ASSUME  DS:CODE
        ASSUME  ES:CODE
        ASSUME  SS:CODE
        ORG     00100H          ;<- REQUIRED FOR .COM FILE
IP      LABEL   NEAR            ;   (USED ON END STATEMENT)
        JMP     START
CURSOR  DW      ?
START   LABEL   NEAR
        MOV     AH,3            ;<- GET CURSOR MODE
        INT     010H
        CMP     CX,00067H       ;<- CHECK FOR BUG
        JNE     NOBUG
        MOV     CX,00607H
NOBUG   LABEL   NEAR
        MOV     CURSOR,CX
        MOV     AH,1            ;<- TURN CURSOR OFF
        MOV     CX,02000H
        INT     010H
```

continued

4C hexadecimal). [Editor's note: *To call a DOS function, put the function number in the AH register—usually with a* MOV AH,<value> *instruction—then call interrupt 21h with* INT 21h.] The batch program that called KEY can test this error level by using the IF command and the special keyword ERRORLEVEL.

The batch file EXKEY.BAT (see listing 1) shows how to use KEY to implement a menu. (Note that once you compile and link the file *<name>*.ASM, you refer to it as a command named <*name*>.) The heart of this batch file is the KEY program and the IF commands. The IF ERRORLEVEL *nn* command is true when the error level is greater than or equal to *nn*, where *nn* is a number between 0 and 255. Stringing together IF commands has the effect of a logical AND operation. In the first IF statement, if the error level is greater than or equal to 49 and not greater than or equal to 50, then the error level must be equal to 49, the ASCII value for 1. The ASCII values for 1, 2, and Escape are 49, 50, and 27, respectively.

KEY.COM's chief virtue is that you can easily construct it without an assembler. The limitations of KEY are that it does not support function keys and is cumbersome to use.

KEYIN

KEYIN.ASM (see listing 2) is a grown-up version of KEY (see also the text box "Assembling KEYIN" above). KEYIN flushes the keystroke buffer to eliminate stray keystrokes, turns off the cursor, supports function keys, and converts lowercase letters to uppercase. KEYIN works like KEY, except that you can test for function keys, and you don't have to do extra tests for lowercase responses. KEYIN still requires that you supply the ASCII value of the character in the IF statement, which is tedious. The value returned for a function key is the scan code with the high bit turned on (see table 1).

KEYIN, like all programs in this article, starts executing at the label IP. Its first step is to save the current cursor (this is slightly complicated by a bug in the earliest IBM ROM BIOS).

Next, it flushes the keyboard of any pending keystrokes and waits for a keystroke. I have previously decided that I will mark non-ASCII characters by setting the high bit of the keystroke's scan code. KEYIN detects non-ASCII values by checking for a zero value in the AL register. If it finds one, it sets the high bit by ORing it with 80 hexadecimal. It then converts ASCII alphabetic values to uppercase, restores the cursor, and exits to DOS with the scan code in the error-level variable.

continued

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS MAIL PERMIT NO. 201 PISCATAWAY, NJ

POSTAGE WILL BE PAID BY ADDRESSEE

BYTE
Subscription Dept.
P.O. Box 6807
Piscataway, NJ 08855-9940

```
FLUSH      LABEL     NEAR          ;<- FLUSH THE KEYBOARD BUFFER
           MOV       AH,1
           INT       016H
           JZ        FLUSHED
           MOV       AH,0
           INT       016H
           JMP       FLUSH
FLUSHED    LABEL     NEAR
           MOV       AH,0          ;<- WAIT FOR A KEYSTROKE
           INT       016H
           OR        AL,AL
           JNZ       REGULAR
           MOV       AL,AH         ;<- FUNCTION AND OTHER SPECIAL KEYS
           OR        AL,10000000B  ; TURN ON HIGH BIT
           JMP       SHORT DONE
REGULAR    LABEL     NEAR          ;<- REGULAR KEY
           CMP       AL,'a'        ; CONVERT TO UPPERCASE
           JB        DONE
           CMP       AL,'z'
           JA        DONE
           ADD       AL,'A'-'a'
DONE       LABEL     NEAR
           PUSH      AX
           MOV       AH,1
           MOV       CX,CURSOR
           INT       010H
           POP       AX
EXIT       LABEL     NEAR          ;<- EXIT
           MOV       AH,04CH
           INT       021H
CODE       ENDS
           END       IP            ;<- REQUIRED FOR .COM FILE
```

Table 1: *Scan codes as returned by the ROM BIOS.*

Scan code	Keystroke	Scan code	Keystroke
3	NUL	82	Insert
15	Left arrow	83	Delete
16-25	Alt-Q,W,E,R,T,Y,U,I,O,P	84-93	F11-F20 (Shift-F1-F10)
30-38	Alt-A,S,D,F,G,H,J,K,L	94-103	F21-F30 (Ctrl-F1-F10)
44-50	Alt-Z,X,C,V,B,N,M	104-113	F31-F40 (Alt-F1-F10)
59-68	F1-F10	114	Ctrl-PrtSc
71	Home	115	Ctrl-Left arrow
72	Up arrow	116	Ctrl-Right arrow
73	PageUp	117	Ctrl-End
75	Left arrow	118	Ctrl-PgDn
77	Right arrow	119	Ctrl-Home
79	End	120-131	Alt-1,2,3,4,5,6,7,8,9,0,-,=
80	Down arrow		
81	PageDown	132	Ctrl-PgUp

Listing 3: *Using the error level as a success/failure flag. This program, TOPATH.ASM, is called with a single argument, a path name (e.g.,* `TOPATH A:\DATA\CALC`*), and it attempts to change the current directory to the one given by the argument. It returns a zero value if the directory change is successful, and a nonzero value if it is not.*

```
CODE       SEGMENT
           ASSUME    CS:CODE       ;<- ASSUMES FOR .COM FILE
           ASSUME    DS:CODE
           ASSUME    ES:CODE
           ASSUME    SS:CODE
```

continued

Changing Directories

TOPATH.ASM (see listing 3) is an example of using the error-level variable as a success/failure flag. The `TOPATH` command takes a path as its only parameter. The program does a DOS change-directory function (DOS function 3Bh) with this path, indicating success with a zero error-level value or failure with a nonzero error-level value. When you use it with the `STRING` program in listing 5, you can use `TOPATH` to let the user enter a new path name and, depending on the result, either begin using the path name or recover from an error.

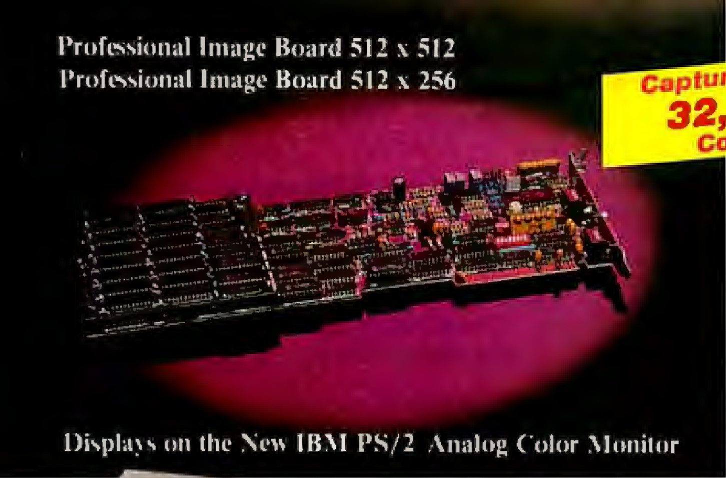
To understand how `TOPATH` works, you must be familiar with some of the internal workings of COMMAND.COM. COMMAND.COM places all the characters to the right of a command in an area of the program segment prefix (PSP) called the *command tail.* The command tail is 1 byte long, followed by the characters themselves and a carriage return (which is not included in the length value). `TOPATH` uses the command-tail length as an index into the command tail to change the carriage return to a NUL (ASCII value 0). This is necessary because the DOS change-directory function expects an ASCIIZ string. (An ASCIIZ string is a string that ends with a NUL.) Next, the program scans forward from the beginning of the command tail to the first nonblank character. Finally, `TOPATH` executes the DOS change-directory function (function 3Bh) and returns a flag indicating success (zero value) or failure (nonzero value) in the error-level variable.

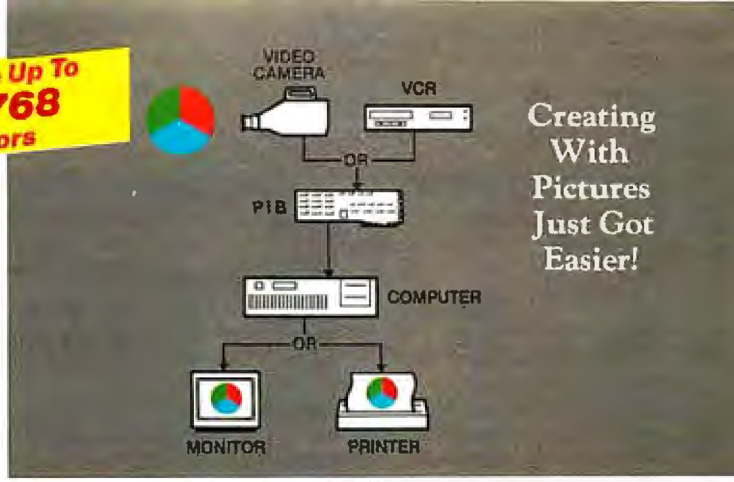
The Environment

The *environment* is an area of memory maintained by COMMAND.COM that contains *environment variables* that hold string values. Strings are placed into and deleted from the environment using the `SET` command. The `PATH` and `PROMPT` commands are shorthand versions of `SET` (i.e., you could also use `SET PATH=...`). A typical environment contains the variables `COMSPEC`, `PATH`, and `PROMPT`. These variables have special meaning to COMMAND.COM itself. Other programs can use environment variables to get information from the user. For example, the Microsoft Macro Assembler uses the environment variable `INCLUDE` to locate include files that are not on the current directory.

A powerful additional feature of the batch processor was undocumented until PC-DOS 3.3. When the batch-file processor sees the name of an environment variable between percent signs, it replaces the name and the surrounding percent signs with the value of that environ-

continued





ADVANCE TO THE NEXT LEVEL

Professional Image Board 512 x 256

Just plug the PIB board into your IBM PC/XT/AT or compatible which allows an ordinary home video camera (color or B/W) or home VCR to be plugged into your system. Now, live, fast action images can be instantly captured and frozen (1/30 second) in a full 32,768 colors. The frozen picture is as close to television quality as can be, digitized by the software package we bundle with PIB board. This software package HALOVISION Written by MEDIA CYBERNETICS INC. allows you to edit, cut, paste, copy, rotate, brush, zoom, scale, add text (21 text fonts) free hand draw, print, etc. . . The PIB is also compatible with the new IBM PS/2 Model 30 and the image can be displayed on IBM's new analog color monitor. Or you can convert the image to EGA display mode. The image can be stored on floppy or hard disk and transmitted to any remote location in the world via modem.

ATI-6/12 System 286

The ATI-6/12 SYSTEM 286 runs at an amazing speed of 12MHz. That's 20% faster than new IBM PS/2 model 50 and 60. To protect your existing software investment, we build-in a normal speed 6MHz, a simple combination of keystrokes will change the speed of the system at any time when necessary. 6/8MHz, 6/10MHz zero wait state and 1 wait state systems also available. Runs all existing software written for IBM PC/XT/AT, including new OS/2 development software.

Professional Image Board Plus 512 x 512

Same features as the PIB 512 x 256 with resolution enhanced to 512 x 512 pixels.

ATI-8/16MHz System Board

This system board runs at 16MHz, 1 wait state Norton 3.0 rating is 18.7 (Compaq deskpro 386 and the new IBM PS/2 Model 80 rating is 18.00 to 18.7). With a fraction of the cost of 386 systems, you can upgrade your 286 system to 386 system performance by replacing your existing 286 system board. Almost all existing add-on cards still work with this system board. This system boards standard feature is a 1MB high speed memory also switchable to 8MHz by keyboard when necessary.

ATI
ATronics International Inc.
We Deliver Advanced Technology
1830 McCandless Dr. Milpitas CA. 95035 USA
(408) 942-3344 **TLX:** 510-600-6093 **FAX:** (408) 942-1674

Southern California Contact: **CNT Marketing**
16580 Harbor Blvd., Ste. J, Fountain Valley, CA 92780 **(714) 839-3724**

Circle 24 on Reader Service Card

```
          ORG       00080H         ;<- LENGTH OF COMMAND TAIL
TAILLEN   DB        ?
          ORG       00081H         ;<- COMMAND TAIL
TAIL      DB        127 DUP(?)
          ORG       00100H         ;<- REQUIRED FOR .COM FILE
IP        LABEL     NEAR           ;   (USED ON END STATEMENT)
          XOR       BH,BH          ;<- PUT NUL AT END OF COMMAND TAIL
          MOV       BL,TAILLEN
          MOV       TAIL[BX],000H
          MOV       AL,' '         ;<- SCAN FORWARD ACROSS BLANKS
          MOV       CX,BX
          MOV       DI,OFFSET TAIL
          CLD
          REPE      SCAS TAIL[DI]
          DEC       DI
          MOV       AH,03BH        ;<- CHANGE DIRECTORY
          MOV       DX,DI
          INT       021H
          JC        EXIT
          XOR       AL,AL
EXIT      LABEL     NEAR           ;<- EXIT
          MOV       AH,04CH        ;   0=OK OTHERWISE=ERROR NUMBER
          INT       021H
CODE      ENDS
          END       IP             ;<- REQUIRED FOR .COM FILE
```

Listing 4: *Reading a keystroke into a previously defined environment variable using the program ONEKEY.ASM. When a batch file executes* ONEKEY, *the program waits for you to press a key, then transfers the result to the environment variable* K.

```
;===========DUMMY SEGMENTS
ENV       SEGMENT   AT OFFFFH      ;<- THE ENVIRONMENT
STRINGS   DB        ?
ENV       ENDS
MEM       SEGMENT   AT OFFFFH      ;<- A MEMORY BLOCK
MEMTYPE   DB        ?
MEMID     DW        ?
MEMSIZE   DW        ?
MEM       ENDS
PSP       SEGMENT   AT OFFFFH      ;<- COMMAND.COM
          ORG       0002CH         ;<- ENVIRONMENT SEGMENT
ENVSEG    DW        ?
PSP       ENDS
;===========CODE SEGMENT
CODE      SEGMENT
          ASSUME    CS:CODE        ;<- ASSUMES FOR .COM FILE
          ASSUME    DS:CODE
          ASSUME    ES:CODE
          ASSUME    SS:CODE
          ORG       00016H         ;<- COMMAND.COM SEGMENT
CMDSEG    DW        ?
          ORG       00100H         ;<- REQUIRED FOR .COM FILE
IP        LABEL     NEAR           ;   (USED ON END STATEMENT)
          JMP       START
CURSOR    DW        ?
START     LABEL     NEAR
;===========TURN CURSOR OFF
          MOV       AH,3           ;<- GET CURSOR MODE
          INT       010H
          CMP       CX,00067H      ;<- CHECK FOR BUG
          JNE       NOBUG
          MOV       CX,00607H
NOBUG     LABEL     NEAR
          MOV       CURSOR,CX
          MOV       AH,1           ;<- TURN CURSOR OFF
          MOV       CX,02000H
          INT       010H
```

continued

ment variable. [Editor's note: *The author has told us that he has had reports that this feature does not work under certain circumstances with PC-DOS 3.0; he added that he has never had any problems with it using PC-DOS 3.1 or later.*] Consider the line

```
PATH %PATH%;C:\MASM
```

If the current path is C:\DOS, this will be expanded to

```
PATH C:\DOS;C:\MASM
```

In this case, you can append the directory C:\MASM to the path without knowing the current path. This expansion takes place only within batch files.

This feature is useful, but it has several problems. One is that there are no DOS functions to get strings out of or into the environment. This is because the environment is not a feature of DOS, but of COMMAND.COM. I view COMMAND.COM as the user interface to DOS and not as a part of the operating system itself. (COMMAND.COM can be replaced with a properly written substitute by adding a line beginning SHELL= in the file CONFIG.SYS.) For most people, the distinction between DOS and COMMAND.COM is lost in practice. Realizing that COMMAND.COM is not DOS helps in understanding why no DOS functions relate to the environment.

Two serious difficulties arise from the add-on nature of the COMMAND.COM user interface. One is that the environment is considered read-only. How a program gets access to anything other than its read-only copy of the true environment is undocumented. Even worse, the environment cannot always be expanded to hold additional or longer strings.

There are two well-known solutions to the problem of limited environment size. The first is to start with a larger environment. Many computer magazines have published articles and tips explaining documented and undocumented ways to increase the initial environment size (see the bibliography for two environment-related articles). Unfortunately, this is not a complete solution. When COMMAND.COM makes a copy of the environment, it discards any excess space (an action that is sensible in relation to a read-only copy). This compression also takes place when COMMAND.COM invokes itself in a nested batch file using the COMMAND /C construct. Therefore, the effect of the larger initial environment is lost in a nested batch file.

The second solution is to reserve space in the root environment for any variables

continued

FULL BLOWN PERFORMANCE.

The Micro Express line of IBM-compatible personal computers isn't just a bare essentials copy of the industry leader, but a powerful system that actually outperforms the "big name" brand.

ME 386
- ☐ 16 MHz 80386 microprocessor
- ☐ 512K 32-bit memory on main board
- ☐ Two 32-bit expansion slots
- ☐ CMOS clock/calendar circuit
- ☐ Socket for 80387 math co-processor
- ☐ Floppy/hard disk controller
- ☐ 1.2 MB floppy disk

$1899

ME 286
- ☐ 6/10 MHz 80286 microprocessor (12 MHz option)
- ☐ 640K memory on main board (upgradable to 1 MB)
- ☐ 8 I/O expansion slots
- ☐ CMOS clock/calendar circuit
- ☐ 80287 math co-processor (optional)
- ☐ Floppy/hard disk controller
- ☐ 1.2 MB floppy disk drive
- ☐ AT style keyboard

$899

ME V20 (AT jr)
- ☐ NEC V20 8 MHz CPU
- ☐ 640K with "0" wait state on mother board
- ☐ Eight expansion slots
- ☐ Socket for 8087-2 math co-processor
- ☐ LED indicator for turbo mode
- ☐ Up to three times faster than the IBM XT®
- ☐ AT style keyboard (RT style optional)
- ☐ 360K floppy disk drive
- ☐ Serial (RS232) port
- ☐ Parallel (printer) port and game port
- ☐ Real time clock/calendar with battery back-up
- ☐ Enhanced graphics card
- ☐ Enhanced graphics monitor

$999

The Road Runner
- ☐ 6/10 MHz 80286 processor
- ☐ 1 MB memory with "0" wait state
- ☐ New Super Twist LCD with Backlit and Reverse; 640 × 200 dots; 80 × 25 characters
- ☐ Monochrome/Color graphics card for external monitor
- ☐ 6 expansion slots
- ☐ Floppy/hard disk controller
- ☐ Serial/printer/clock-calendar
- ☐ 1.2 MB floppy disk drive
- ☐ 20 MB hard disk (40 MB option)
- ☐ 12 function AT style keyboard
- ☐ 20 pounds

$1699

CALL TOLL-FREE:

(800) MICRO-21

(714) 662-1973, (818) 785-0274

ME MICRO EXPRESS

2114 South Grand Ave., Santa Ana, CA 92705
14640 Victory Blvd., Ste. 130, Van Nuys, CA 91411
Fax: (714) 662-1258, Telex:9102403029

IBM/AT/XT are trademarks of International Business Machines Corp.

Circle 133 on Reader Service Card (Dealers: 134)

```
;===========LOCATE ENVIRONMENT
            MOV        DX,CMDSEG          ;<- LET DS=COMMAND.COM SEGMENT
            MOV        DS,DX
            ASSUME     DS:PSP
            MOV        AX,ENVSEG          ;<- CHECK ENVIRONMENT SEGMENT
            OR         AX,AX
            JNZ        HAVEIT
            MOV        AX,DS              ;<- ROOT COMMAND.COM
            DEC        AX                 ;   WALK MEMORY BLOCKS FORWARD
            MOV        DS,AX              ;   TO THE ENVIRONMENT
            ASSUME     DS:MEM
NEXTSEG     LABEL      NEAR
            CMP        MEMTYPE,'M'
            JNE        ERROR1             ;<- !ERROR! SHOULD NOT OCCUR
            MOV        AX,DS
            INC        AX
            ADD        AX,MEMSIZE
            MOV        DS,AX
            ASSUME     DS:MEM
            CMP        MEMID,DX
            JNE        NEXTSEG
            MOV        AX,DS
            INC        AX
HAVEIT      LABEL      NEAR
;===========LOCATE K=IN THE ENVIRONMENT
            MOV        ES,AX
            ASSUME     ES:ENV
            XOR        AL,AL
            MOV        CX,08000H
            XOR        DI,DI
SEARCH      LABEL      NEAR
            REPNE      SCAS     STRINGS
            CMP        STRINGS[DI],AL
            JE         ERROR2             ;<- !ERROR! K=MISSING
            CMP        STRINGS[DI],'K'
            JNE        SEARCH
            CMP        STRINGS[DI][1],'='
            JNE        SEARCH
            ADD        DI,2               ;<- POSITION AFTER = SIGN
;===========GET KEYSTROKE. RESTRICT TO ASCII ! THROUGH ~
FLUSH       LABEL      NEAR               ;<- FLUSH THE KEYBOARD BUFFER
            MOV        AH,1
            INT        016H
            JZ         FLUSHED
            MOV        AH,0
            INT        016H
            JMP        FLUSH
FLUSHED     LABEL      NEAR

GETKEY      LABEL      NEAR
            MOV        AH,0               ;<- WAIT FOR A KEYSTROKE
            INT        016H
            OR         AL,AL
            JZ         GETKEY
            CMP        AL,'!'
            JB         GETKEY
            CMP        AL,'~'
            JA         GETKEY
REGULAR     LABEL      NEAR          ;<- REGULAR KEY
            CMP        AL,'a'        ;   CONVERT TO UPPERCASE
            JB         STORE
            CMP        AL,'z'
            JA         STORE
            ADD        AL,'A'-'a'
STORE       LABEL      NEAR
            MOV        STRINGS[DI],AL     ;<- STORE IN THE ENVIRONMENT
            MOV        AH,1               ;<- RESTORE CURSOR
            MOV        CX,CURSOR
            INT        010H
```

continued

you might need later. You do this by using the SET command to set each variable to a value that is as long as the longest possible value. This solution has several problems, too. One is anticipating all the variables that you will need. Another is making certain to reset these strings to the longest possible value after you use them. This second problem is insurmountable—when the user hits Control-Break to exit a batch file, the space set aside is often lost.

Reading a Keystroke

ONEKEY.ASM (see listing 4) illustrates a different approach to the problem of user input into batch files. Before you can use ONEKEY, you must define the environment variable K by executing the line:

```
SET K=X
```

which sets K to the value of the one-letter text string X; you can set K to any initial value you want.

ONEKEY waits for a keystroke, then places its value in K. It follows a complicated search to find the environment variable it needs. First, it saves and hides the cursor. Then it uses the undocumented fact that the segment of the current COMMAND.COM is stored at offset 16 hexadecimal in the PSP. It then locates the environment of this COMMAND.COM using the segment value at offset 2C hexadecimal. Unfortunately, when the value is 0 (as it is for the root COMMAND-.COM), it must follow the undocumented structure of the memory chains to locate the environment.

Each piece of DOS memory is located on a 16-byte boundary and contains a 16-byte memory-control block followed by data. Only the first 5 bytes of the control block have meaning. The first byte is either the letter M or Z, where M indicates more, and Z indicates the last block. The next word contains either the segment of the PSP that owns this memory block, or 0 if it is unallocated. The final word contains the size, in 16-byte increments, of the memory block.

Once it locates the environment, ONEKEY scans through it, looking for the variable K. It does this by looking for a NUL character followed by a K followed by an equal sign. It then flushes the keyboard and reads the next keystroke, accepting only a printable keystroke in the ASCII range from ! to ~. For example, the batch statement

```
IF ERRORLEVEL 49 IF NOT ERRORLEVEL
  50 GOTO a
```

becomes

continued

Q: What's the difference?

	PS/2 MODEL 50	IBM PC AT with TurboSwitch*
AUTOCAD regen (nozzle)	8.9 sec.	8.9 sec.
344K Spreadsheet Recalc	6.9 sec.	6.9 sec.
PC Magazine Benchmark NOP	3.3 sec.	3.3 sec.
PC Magazine Benchmark String Sort & Move	2.3 sec.	2.3 sec.
PC Magazine Benchmark Prime Number Sieve	3.2 sec.	3.2 sec.
Upgrade Cost	$3,595.00	$124.95

A: About 3,500 bucks.

The benchmarks and price tags tell the story. Having prestigious Fortune 500 customers confirms that the story is true. And everyone except IBM admits that IBM PC AT models 68, 99, 239, and 339 will support OS/2 well into the 1990s. *As long as they run fast enough (that's where TurboSwitch comes in).*

A 30-day money-back guarantee ensures that you will be a satisfied customer. And a 2-year warranty gives you the confidence Megahertz builds into every product.

Sounds like you don't need a PS/2 after all.

TurboSwitch®

MHz

MEGAHERTZ CORPORATION

2681 Parley's Way, Salt Lake City, UT 84109 (801) 485-8857

1-800-33-TURBO

a subsidiary of Vector Development, Inc., a public company

*10 MHz speed.

```
;===========
            MOV         AL,0
EXIT        LABEL       NEAR              ;<- EXIT
            MOV         AH,04CH
            INT         021H
ERROR1      LABEL       NEAR              ;<- COULD NOT LOCATE ENVIRONMENT
            MOV         AL,1
            JMP         EXIT
ERROR2      LABEL       NEAR              ;<- K=MISSING FROM ENVIRONMENT
            MOV         AL,2
            JMP         EXIT
CODE        ENDS
            END         IP                ;<- REQUIRED FOR .COM FILE
```

Listing 5: *Entering a text string into a previously defined environment variable using the program STRING.ASM. When a batch file executes* STRING, *the program accepts a string of characters ended by a carriage return and assigns the result to the environment variable* STRING *(up to the maximum length reserved for it).*

```
VECTOR      STRUC
REGIP       DW          ?
REGCS       DW          ?
VECTOR      ENDS
CODE        SEGMENT
            ASSUME      CS:CODE           ;<- ASSUMES FOR .COM FILE
            ASSUME      DS:CODE
            ASSUME      ES:CODE
            ASSUME      SS:CODE
            ORG         00100H            ;<- REQUIRED FOR .COM FILE
IP          LABEL       NEAR              ;   (USED ON END STATEMENT)
            JMP         SHORT START
HOLDS       DB          16                ;<- FOR DOS BUFFERED INPUT FUNCTION
HAS         DB          0
INPUT       DB          15 DUP(' '),0
SET         DB          11                ;<- FOR INT 2E
            DB          'SET STRING='
SETTING     DB          16 DUP(?)
;===========INT 2 E PROCEDURE
SS_SP       VECTOR      <>
INT2E       PROC        NEAR
            PUSH        AX                ;<- SAVE ALL REGISTERS
            PUSH        BX
            PUSH        CX
            PUSH        DX
            PUSH        BP
            PUSH        SI
            PUSH        DI
            PUSH        DS
            PUSH        ES
            PUSHF
            MOV         SS_SP.REGIP,SP    ;<- SAVE SS:SP
            MOV         SS_SP.REGCS,SS
            INT         02EH              ;<- DO INT 2E
            ASSUME      DS:NOTHING
            ASSUME      ES:NOTHING
            ASSUME      SS:NOTHING
            MOV         SS,SS_SP.REGCS    ;<- RESTORE SS:SP
            ASSUME      SS:CODE
            MOV         SP,SS_SP.REGIP
            POPF                          ;<- RESTORE ALL REGISTERS
            POP         ES
            ASSUME      ES:CODE
            POP         DS
            ASSUME      DS:CODE
            POP         DI
            POP         SI
```

continued

```
IF %k% == 1 GOTO a
```

Also, a program can save the current keystroke value for future use, as in

```
SET previous=%k%
```

Direct Manipulation of Environment Variables

Another way to set environment variables from a program is to use the COMMAND.COM "back door." You do this by exploiting the undocumented interrupt 2Eh. INT 2E executes the command pointed to by DS:SI (the data-segment register offset by the source-index register). This command must be in the form of a command tail, a 1-byte command length followed by the command and a carriage return. Keep in mind that INT 2Eh affects the root environment. This means that if you have executed another copy of COMMAND.COM (which has its own copy of the environment), the execution of an INT 2Eh instruction will *not* affect the current (copied) environment.

The program STRING.ASM (see listing 5) is an extension of ONEKEY. STRING uses INT 2Eh to set an environment variable, also called STRING, to the value typed in by the user. As with ONEKEY, you must define the environment variable before you can use the STRING program; in this case, you should set it to the longest string you think you will ever want to enter from the keyboard.

STRING gets a text string from the keyboard, builds a string that starts SET STRING=<*input string*>, then executes that string using the undocumented INT 2Eh function. This last step causes the environment variable *STRING* to take on the value you typed in.

The STRING.ASM program starts with a definition of the VECTOR structure. This will be used in the INT 2Eh procedure to save and restore the stack. The DOS-buffered input function (the INT 21h call with the AH register set to 0C hexadecimal) uses HOLDS, HAS, and INPUT to read the keyboard input into memory starting at location INPUT. INPUT and SET will be used in the call to the INT 2Eh procedure. INT 2Eh pushes all registers and saves the SS and SP registers in SS_SP, performs the INT 2Eh that executes the SET STRING= statement, then restores the stack and all registers.

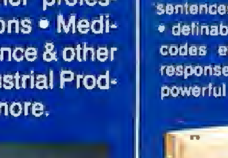
During execution of the main routine (at START), first free all unneeded memory by doing a DOS function 4A hexadecimal. (The calling program begins with control of the computer's entire address space. If no memory is freed, the INT 2Eh done later in the STRING program will fail because COMMAND-

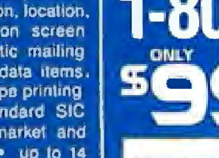
continued

SALES GENERATOR PLUS

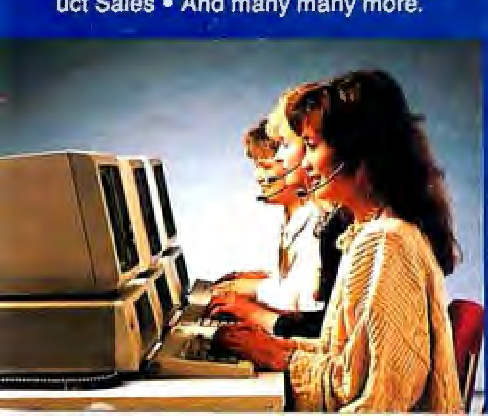
A revolution in automating sales & marketing

SALES GENERATOR PLUS is so easy to use the beginner can amazingly have it running in minutes. The advanced user may be exploring new applications for this complete and powerful program several years from now. It features all the software tools necessary to conduct an effective sales campaign. Right at your fingertips without complicated commands, Sales Generator Plus can automate : Employment & Real Estate Agents • Service Organizations • Legal & Other professionals • Educational Institutions • Medical & Dental • Leasing • Insurance & other services • Consumer and Industrial Product Sales • And many many more.





FEATURES: Menu driven • record capacity limited only by disk space • support contract available • not copy protected • color and monochrome • works with any standard (ACSII) word processor or Sidekick • on demand a customer-by-customer communication history • automatic sales follow-up • telephone follow-up management • User definable reporting • Sales strategies can be defined by department head & follow the critical path of closing a sale • Intelligent record updating option • correspondence follow-up management • target customer selection report • sales performance reporting - daily, weekly, monthly, etc. by region, location, salesman, product or product group • customized on screen script • direct mail management • automatic mailing including conditional mail merging & nesting of data items, sentences and paragraphs • label and envelope printing • definable target market profiles • recognizes standard SIC codes etc. • significant dates • profile of target market and response profile each with 20 definable fields • up to 14 powerful selection criteria & data base ranges.



Trademarks: IBM-International Business Machines, Sidekick - Borland International

• appointment management & lists • automated telephone followup • import/export interfaces • selection of sort criteria • follow-up lists • name & address lists • phone lists • runs on any IBM PC or compatible • its versatility and features will amaze you.

FOR IMMEDIATE SHIPMENT
In US and Canada

1-800-268-8588

In Hawaii & Alaska: 1-(416)-862-0017

ONLY $99.95 Shipping & Handling $5

SG+

SALES GENERATOR PLUS

Supersell Software Inc.
316 California Avenue, #990
Reno, Nevada,
89509

MASTERCARD
VISA
AMEX
DINERS CLUB

```
          POP       BP
          POP       DX
          POP       CX
          POP       BX
          POP       AX
          RET
INT2E     ENDP

START     LABEL     NEAR
          MOV       AH,04AH
          MOV       BX,OFFSET FREE
          DEC       BX
          MOV       CL,4
          SHR       BX,CL
          INC       BX
          INT       021H
          MOV       AH,00CH              ;<- GET INPUT STRING
          MOV       AL,00AH
          MOV       DX,OFFSET HOLDS
          INT       021H
          XOR       CH,CH
          MOV       CL,HAS
          JCXZ      ERROR1
          ADD       SET,CL
          MOV       SI,OFFSET INPUT
          MOV       DI,OFFSET SETTING
          REP       MOVSB
          XOR       BH,BH
          MOV       BL,SET
          MOV       SET[BX][1],00DH
          MOV       SI,OFFSET SET
          CALL      INT2E
EXIT      LABEL     NEAR                 ;<- EXIT
          MOV       AH,04CH
          INT       021H
ERROR1    LABEL     NEAR
          MOV       AL,1
          JMP       EXIT
FREE      LABEL     BYTE
CODE      ENDS
          END       IP                   ;<- REQUIRED FOR .COM FILE
```

.COM will have no memory at its disposal.) Then STRING executes DOS function 0C hexadecimal to clear the keyboard buffer and put keyboard input into the memory location starting at location INPUT.

The string is then moved to SETTING, where it finishes the SET STRING= string. In addition, the program corrects the length byte at location SET and ends the string with a carriage return. If the user doesn't enter a string, the program returns with an error level of 1. Otherwise, it executes the INT 2Eh routine (thus changing the value of the STRING variable) and returns to the calling program by executing DOS function 4C hexadecimal (terminate a process).

Using the Programs

Batch files were originally designed to eliminate the need to retype a routine set of commands, and the use of replaceable parameters lets you change what the batch file does *when you start its execution*. By using the programs in this article, you can now influence what the batch file does *while it's executing*, which is a more natural and potentially more powerful form of control. In addition, the STRING and TOPATH commands allow you to change to a new directory based on a user-specified directory name.

You can use these commands in several important ways. If you have a hard disk, you can use them to automate some of the inevitable housekeeping tasks. You can also create a menu-driven shell that lets you (or someone unfamiliar with the computer) perform certain tasks easily and without error. ■

BIBLIOGRAPHY

Disk Operating System Version 3.00, IBM Corp., Boca Raton, FL, 1984.

Shaw, R. H. "A User's Guide to the DOS Environment," *PC* magazine, March 31, 1987, page 287.

Shaw, R. H. "Expanding the DOS Environment," *PC* magazine, April 14, 1987, page 295.

Technical Reference Personal Computer AT, IBM Corp., Boca Raton, FL, 1984.

The Right Business Connections... Made Easy

If you use more than one computing device in your office, you and your colleagues probably waste many hours making connections and changing settings—hours you'd rather devote to real business.

A Newbridge *MainStreet*™ system takes care of the equipment connections—once and for all—while you take care of business.

With a single *MainStreet* controller you'll have a complete "Desktop Network" for up to 48 equipment connections.

Even larger interdepartmental networks can be created by linking controllers.

EasyStreet™ software brings the network to your PC, letting you "select and connect" to any equipment in the network, perform secure file transfers without interruptions to others, conduct split-screen conversations with colleagues, emulate terminals and queue for busy devices.

Whether your company is big or not so big, you'll be surprised how quickly your *MainStreet* desktop network leads to greater productivity. At an amazingly low price* ... and isn't that your bottom line?

Desktop Networking

NEWBRIDGE

Newbridge Networks Inc.
1-800-332-1080
13873 Park Center Road,
Suite 160
Herndon, VA 22071
(703) 834-3600

Newbridge Networks Corporation
1-800-267-6246
1051 Baxter Road
Ottawa, Ontario K2C 3P2
(613) 726-1601

Newbridge Networks Limited
0-800-373799
22 Welsh Street
Chepstow, Gwent, UK NP6 5LL
Tel. 02912 6896

* *About one tenth the cost of most LANs—another good reason to contact Newbridge today.*

For more information call from your modem ***703-834-1080***
(0 - 1200 baud, 8 bit, no parity, 1 stop bit) and enter the access code ***MAINST7*** when prompted.

Circle 150 on Reader Service Card

A close look at a growing lineage

IBM PC Family BIOS Comparison

by Jon Shiell

In the two years since the first BIOS comparison table appeared in BYTE [Editor's note: *See the "IBM PC Family BIOS Comparison" in BYTE's* Inside the IBM PCs, *Fall 1985*], the number of machines in the family has more than doubled. This article presents an expanded and updated BIOS table. Except for the Model 30, the PS/2 systems actually have two versions of the BIOS in their ROMs. The first version, for use in real mode and compatible with the BIOS in the prior PCs, is covered in this article. The other BIOS is for use with the OS/2 operating system and will not be covered here.

As the PC family has grown, there have been additions and deletions to the basic hardware set of the machines. (Because the Model 25 was only recently introduced, detailed information on its BIOS was not available for this article.) This article attempts to provide a comparative perspective of the various Basic I/O System (BIOS) features. If you wish to program on one machine, this article can help determine which functions apply across all machines in the family.

Table 1 gives the system configurations for IBM PC computers, with the exception of the 3270 PC. Table 2 describes the ROM BIOS interrupt vectors; table 3 lists BIOS video modes; and table 4 lists low-memory reserved addresses. Table 5 describes hardware interrupt request lines; table 6 covers Expanded Memory Specification (EMS) function-call interfaces. Table 7 covers multitasking hooks using interrupt 15; table 8 gives BIOS extension addresses; and table 9 lists the NETBIOS modifications and additions to DOS.

The purpose of the BIOS is to present a common interface to the program, be it an applications program or an operating system, to minimize the amount of code that must be rewritten when using different machines. The BIOS lets the programmer isolate hardware dependence to a single set of primitive routines. What you gain from this is portability and compatibility between different hardware environments. At the same time, you retain almost all the speed and control of direct hardware access.

The BIOS is made up of the code and programs that provide the device-level control for the major I/O devices in the system. In the IBM PC family, the BIOS is contained in ROM on the system board, along with cassette BASIC and a set of routines (called POST, for power-on self test) that check out the machine when you turn it on.

The BIOS creates hardware independence by providing a level of indirection and separation from the hardware. For example, when using a BIOS call to send a character to a printer, a programmer doesn't need to know what the I/O address of the printer port is or how to control it.

The BIOS is normally invoked via a set of interrupts vectored into various BIOS entry points. Other interrupt vectors are used to service hardware interrupts, such as "disk operation finished." In practical terms, the software invokes the BIOS by loading the appropriate registers in the microprocessor and issuing an INT instruction. For example,

```
MOV AH,0   ; Load AH with the BIOS
             function code for
             "print the character
             in register AL"
```

continued

Jon Shiell is a contributing editor at BYTE. He can be reached c/o BYTE, One Phoenix Mill Lane, Peterborough, NH 03458.

Illustration: Chris Spollen © 1987

Table 1: *System configurations for the IBM PC family of computers.*

Feature	PC	PC XT	PCjr	PC AT	PC AT 3x9	XT/2	XT/ 286	PC CVT	PS/2 30	PS/2 50	PS/2 60	PS/2 80	Comments
Model byte	FF	FE	FD	FC	FC	FB	FC	F9	FA	FC	FC	F8	@ FFFFE
Type byte	N/U	N/U	N/U	N/U	01	01	02	00	00	04	05	00	
BIOS level	N/U	N/U	N/U	N/U	00	00	00	00	00	00	00	00	
Hardware configuration	N/U	N/U	N/U	N/U	70	D0	70	38	FA	F6	F6	F6	Note 1
Processor type	8088	8088	8088	80286	80286	8088	80286	80C88	8086	80286	80286	80386	
Processor speed	4.77	4.77	4.77	6	8	4.77	6, 0 WS	4.77	8	10	10	16/20	Note 2; 1 wait state (WS) unless otherwise noted.
Num coprocessor speed	4.77	4.77	4.77	4	5.3	4.77	6	4.77	8	10	10	16/20	
DMA speed	4.77	4.77	4.77	3	4	4.77	3	4.77	4	10	10	8/10	
Bus width	8	8	8	16	16	8	16	8	8	16	16	32	Note 3
Maximum memory (in bytes)	640K	640K	640K	15M	15M	640K	15M	512K	640K	16M	16M	>16M	Note 4
8-bit DMA channels	4	4	0	4	4	4	4	3	4	4	4	4	Note 5
16-bit DMA channels	0	0	0	3	3	0	3	0	0	3	3	3	
Timer channels 0 and 2	●	●	●	●	●	●	●	●	●	●	●	●	Note 6
Timer channel 1	●	●	●	●	●	●	●	○	●	●	●	●	Note 7
System clock	○	○	○	●	●	○	●	●	●	●	●	●	Note 8
Number of function keys	10	10	10	10	10/12	10/12	10/12	12	12	12	12	12	Note 9

Notes:

All memory addresses and interrupts are in hexadecimal. The PC AT 3x9 models are the 319 and 339. The older models are the 099, 068, and 239. The PS/2 Model 25 has the same system configuration as the Model 30, except it has no hard disk and a different model version in the BIOS.

● = Yes; ○ = No; N/U = not used.

1. Configuration parameters, INT 15 (AH=0C0) returns a pointer to a block with the following format:

DW	8	Length of following table
DB	Model_byte	System model; see hardware table for specific values
DB	Type_byte	System model type
DB	BIOS_level	BIOS revision level
DB	HW_config	10000000 = DMA channel 3 used by fixed disk BIOS
		01000000 = Cascaded interrupt Level 2
		00100000 = Real-time clock available (RT/CMOS RAM chip)
		00010000 = Keyboard scan code hook 1A (PC AT and XT 286)
		00010000 = Keyboard intercept (INT 15, AH=41) supported (PC CVT and PS/2)
		00001000 = Wait on external event (INT 15, AH=41) supported (PC CVT); reserved on PS/2 systems
		00000100 = Extended BIOS data area allocated
		00000010 = Micro Channel system
		00000001 = Reserved
DW	0	Reserved
DW	0	Reserved

The PC XT 2 (Model 5160, model = FB, type = 01, BIOS date 01/10/86) returns an incorrect value for the configuration parameter. The incorrect value indicates that the level 2 interrupt is cascaded into another interrupt controller, and that DMA channel 3 is not used by the system BIOS when a hard disk is installed.

2. In the PCjr, the video is mapped into the lower 128K bytes of memory. Accesses to that area tend to be 50 percent slower than accesses for a normal PC. The PC CVT uses static CMOS RAM, so it needs no refresh cycles; thus, programs may run slightly (up to 5 percent) faster than expected. The PS/2 Model 80 comes in two speeds—16 MHz and 20 MHz, both with one wait state.

3. The PS/2 Model 30's memory is 16 bits, but the I/O bus is the 8-bit PC bus.

4. The PCjr has up to 128K bytes of internal memory; full expansion requires sidecars. The PS/2 Model 80 supports 32-bit memory addresses, so in theory you could put up to 4 gigabytes in one.

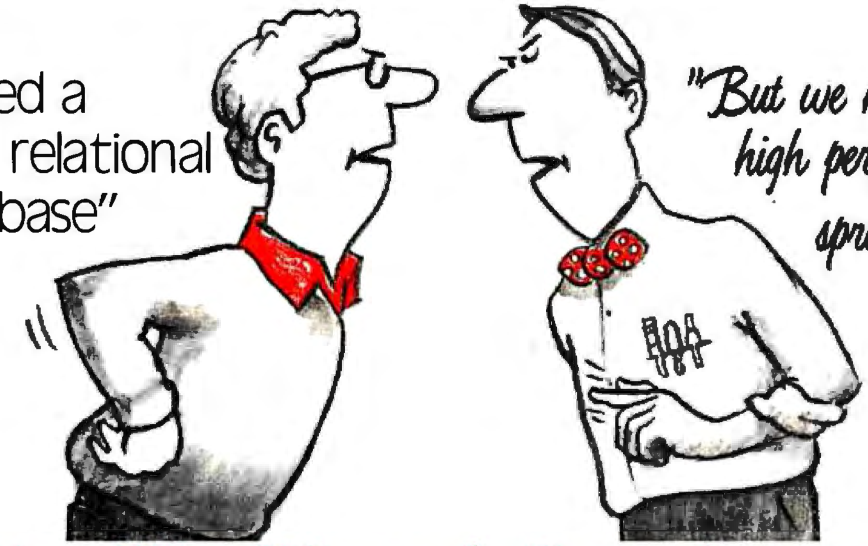
5. The PC CVT doesn't need to use one channel for dynamic RAM refresh, so its channel 3 acts like a PC's channel 4.

6. The PC CVT supports only modes 0, 2, 3, and 4 on channel 0.

7. Use depends on model (mostly refresh timing).

8. RT/CMOS RAM chip; PC CVT does not save configuration here.

9. The PC XT Models 089, 268, and 278 have the new keyboard; the other three models have the old keyboard. The PC AT Model 339 has the new keyboard, and the Model 319 has the old keyboard. The PC CVT generates function keys F11 and F12 with multiple keystrokes.



It's a Win-Win with Open Access II The Most Powerful Database and Spreadsheet Available in An Integrated Package

The arguments for integrated software used to be convenience, ease of use, and shortened learning curve. The argument against it was no power. The argument is over. According to our users, the primary reason for their purchase of Open Access II is the power of the relational database and the spreadsheet with 3-D graphics, followed by convenience and ease of use. And, by the way, this database and spreadsheet are integrated with a word processor, communications and desk accessories.

Nose-to-Nose Comparison Chart

	Database	Form Query	Report Generator	Query Processor	Relational Database	Graphics	3-D Graphics	Spreadsheet	Goal Seeking	Work Processor	Communications	Time Management
Open Access II	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Symphony	✓	✓	✓			✓		✓		✓	✓	
Framework	✓		✓	✓		✓		✓		✓	✓	

What's New in Open Access II

Meeting the challenge of advanced hardware technology, Open Access II update version 2.05 includes graphics drivers for the IBM Personal Systems/2, extended memory support, math co-processor support (8087/80287) and 37 other additions and alterations.

Developers Delight

The power and flexibility available in this package makes it an ideal software choice for developers of complex vertical applications, developers in need of large databases with sophisticated programming capabilities. With the addition to our product line of Runtime System and our upcomig compiler, Open Access II has strengthened its position as a software developer's power tool.

Accounting Access is now available for customized bookkeeping with Open Access II.

Also, Filling Your Network Needs

Local Area Network users around the world are singing the praises of Open Access II Network. So is the press, Infoworld calls it "an excellent value...may be one of the software bargains of the year." LAN Magazine says"...a sharp product...a unique and interesting spreadsheet locking scheme."

Test Drive - $19.95

We invite you to experience the power, flexibility, and performance available in this integrated package by test driving Open Access II with our Demo Tutorial. For only $19.95 you can experience this software, limited only by file size restrictions.

Call 800-521-3511
(if you're in California call 800-621-7490).
Order your Demo Tutorial today, it's a powerful experience.

Software Products International
10240 Sorrento Valley Road · San Diego, California 92121

Open Access II users call for update trade-in information.

Table 2: *ROM BIOS interrupt vectors.*

Interrupt	Function code	PC	PC XT	PCjr	PC AT	XT /2	XT /286	PS CVT	PS/2 30	PS/2 other	BIOS	Comments
0	N/A	●	●	●	●	●	●	●	●	●		Divide by zero trap
1	N/A	●	●	●	●	●	●	●	●	●		Single-step mode (used by debug)
2 NMI	N/U	●	●	○	●	●	●	○	●	●		Parity check routine
	N/U	●	●	○	○	●	○	○	●	○		Coprocessor interrupt (directly)
	N/U	○	○	○	●	○	●	○	○	●		Coprocessor interrupt (via redirection from Int 75, IRQ 13)
	N/U	○	○	●	○	○	○	●	○	○		Keyboard interrupt routine (with data ready)
	N/U	○	○	○	○	○	○	●	○	●		I/O channel check
	N/U	○	○	○	○	○	○	●	○	○		Disk-controller power-on request
	N/U	○	○	○	○	○	○	●	○	○		System suspend
	N/U	○	○	○	○	○	○	●	○	○		Real-time clock (periodic, update-ended, or alarm interrupt)
	N/U	○	○	○	○	○	○	○	○	●		System watchdog timer, time-out interrupt (IRQ0 interrupt missed)
	N/U	○	○	○	○	○	○	○	○	●		Uchannel DMA timer, time-out interrupt (DMA burst greater than 7.8 microseconds)
3	N/A	●	●	●	●	●	●	●	●	●		Breakpoint (used by debug)
4	N/A	●	●	●	●	●	●	●	●	●		Overflow trap
5	N/A	●	●	●	●	●	●	●	●	●		Print-screen function (uses address 50:0 for status)
6	N/U	○	○	○	○	○	○	○	○	○		Reserved
7	N/U	○	○	○	○	○	○	○	○	○		Reserved
8 (IRQ 0)	N/A	●	●	●	●	●	●	●	●	●		Timer interrupt handler (entered about 18.2 times per second)
9 (IRQ 1)	N/A	●	●	●	●	●	●	●	●	●		Keyboard interrupt handler
A (IRQ 2)	N/A	○	○	○	●	○	●	○	○	●		Cascade for IRQ8 to IRQ15
	N/A	○	○	○	○	○	○	○	○	●	EGA	Vertical retrace interrupt for EGA and VGA
B (IRQ 3)	N/A	●	●	●	●	●	●	●	●	●		COM2: controller (e.g., serial port) hardware interrupt entry
C (IRQ 4)	N/A	●	●	●	●	●	●	●	●	●		COM1: controller (e.g., serial port) hardware interrupt entry
D (IRQ 5)	N/A	●	●	○	●	●	●	○	●	○		Alternate printer LPT2: (PC AT's 80287 initial-interrupt entry)
	N/A	○	●	○	○	●	○	○	●	○		Hard disk controller
	N/A	○	○	●	○	○	○	○	○	○		Vertical retrace interrupt (display)
E (IRQ 6)	N/A	●	●	●	●	●	●	●	●	●		Disk-controller interrupt entry
F (IRQ 7)	N/A	●	●	●	●	●	●	●	●	●		Parallel printer, LPT1:
10 Video I/O	0	●	●	●*	●	●	●	●	●	●	EGA	Set CRT mode
	1	●	●	●	●	●	●	●	●	●		Set cursor type
	2	●	●	●	●	●	●	●	●	●		Set cursor position
	3	●	●	●	●	●	●	●	●	●		Read cursor position
	4	●	●	●	●	●	●	●	○	○		Read light-pen position
	5	●	●	●*	●	●	●	●	●	●		Select active display page
	6	●	●	●	●	●	●	●	●	●		Scroll active page up
	7	●	●	●	●	●	●	●	●	●		Scroll active page down
	8	●	●	●	●	●	●	●	●	●		Read attribute/character at current cursor position
	9	●	●	●	●	●	●	●	●	●		Write attribute/character at current cursor position
	0A	●	●	●*	●	●	●	●	●	●		Write character only at current cursor position
	0B	●	●	●*	●	●	●	●	●	●		Set color palette
	0C	●	●	●	●	●	●	●	●	●		Write dot
	0D	●	●	●	●	●	●	●	●	●		Read dot
	0E	●	●	●	●	●	●	●	●	●		Teletype write to active page
	0F	●	●	●	●	●	●	●	●	●		Return current video state
	10	○	○	●	○	○	○	○	●	●	EGA	Set palette registers
	11	○	○	○	○	○	○	○	●	●	EGA	Character-generator routine
	12	○	○	○	○	○	○	○	●	●	EGA	Alternate select
	13	○	○	○	●	○	●	○	●	●	EGA	Write string (with optional attributes)
	14	○	○	○	○	○	○	●	○	○		Load LCD character font or set LCD high-intensity substitute
	15	○	○	○	○	○	○	●	○	○		Return active-display type and parameters
	1A	○	○	○	○	○	○	○	●	●		Read/write display combination code
	1B	○	○	○	○	○	○	○	●	●		Return functionality/state information

Interrupt	Function code	PC	PC XT	PCjr	PC AT	XT /2	XT /286	PS CVT	PS/2 30	PS/2 other	BIOS Comments
	1C	○	○	○	○	○	○	○	○	●	Save/restore video state
11	N/U	●	●	●	●	●	●	●	●	●	Equipment determination; returns the EQUIP_FLAG from BIOS data area
12	N/U	●	●	●	●	●	●	●	●	●	Memory size determination; returns amount of memory in the system
13 Disk	0	●	●	●	●	●	●	●	●	●	Reset disk system
	1	●	●	●	●	●	●	●	●	●	Read status of last operation
	2	●	●	●	●	●	●	●	●	●	Read sectors into memory
	3	●	●	●	●	●	●	●	●	●	Write sectors from memory
	4	●	●	●	●	●	●	●	●	●	Verify sectors
	5	●	●	●	●	●	●	●	●	●	Format a track
	8	○	○	○	3x9	○	●	●	●	●	Return the current drive parameters
	15	○	○	○	●	●	●	●	●	●	Read disk type (none, disk-no-change line, disk, hard disk)
	16	○	○	○	●	●	●	●	●	●	Disk-change line status
	17	○	○	○	●	●	●	●	●	●	Set disk type for format
	18	○	○	○	3x9	●	●	○	●	●	Set media type for format
	N/A	●	●	●	●	●	●	●	●	●	Uses BIOS data area DISK_POINTER
	N/A	○	○	○	●	○	●	●	●	●	Uses disk-drive-media-type bytes at 40:90 and 40:91
	N/A	○	○	○	●	○	●	○	●	●	Uses rest of disk "state machine" at 40:92 to 95
13 Hard disk	N/U	●	○	●	○	○	○	●	○	○	Not used by PC, PCjr, or PC CVT
	0	○	●	○	●	○	●	○	●	●	Reset disk system
	1	○	●	○	●	○	●	○	●	●	Read status of last disk operation
	2	○	●	○	●	○	●	○	●	●	Read sectors into memory
	3	○	●	○	●	○	●	○	●	●	Write sectors from memory
	4	○	●	○	●	○	●	○	●	●	Verify sectors
	5	○	●	○	●	○	●	○	●	●	Format a track (not enhanced-small-device-interface [ESDI] disks)
	6	○	●	○	N/U	○	N/U	○	N/U	N/U	Format a track and set bad-sector flags
	7	○	●	○	N/U	○	N/U	○	N/U	N/U	Format the drive starting at the desired track
	8	○	●	○	●	○	●	○	●	●	Return the current drive parameters
	9	○	●	○	●	○	●	○	●	●	Initialize drive-pair character (uses INT 41 and 46, not ESDI disks)
	0A	○	●	○	●	○	●	○	D	D	Read long
	0B	○	●	○	●	○	●	○	D	D	Write long
	0C	○	●	○	●	○	●	○	●	●	Seek to desired track
	0D	○	●	○	●	○	●	○	●	●	Alternate disk reset (not ESDI disks)
	0E	○	●	○	N/U	○	N/U	○	D	D	Read sector buffer
	0F	○	●	○	N/U	○	N/U	○	D	D	Write sector buffer
	10	○	●	○	●	○	●	○	●	●	Test drive ready
	11	○	●	○	●	○	●	○	●	●	Recalibrate
	12	○	●	○	N/U	○	N/U	○	D	D	Controller RAM diagnostic
	13	○	●	○	N/U	○	N/U	○	D	D	Drive diagnostic
	14	○	●	○	●	○	●	○	D	D	Controller internal diagnostic
	15	○	○	○	●	○	●	○	●	●	Read disk type (none, disk-no-change line, disk, hard disk)
	19	○	○	○	○	○	○	○	●	●	Park heads
	1A	○	○	○	○	○	○	○	○	●	ESDI disks only; format unit
14 RS-232C I/O	0	●	●	●	●	●	●	●	●	●	Initialize communications port
	1	●	●	●	●	●	●	●	●	●	Send a character
	2	●	●	●	●	●	●	●	●	●	Receive a character
	3	●	●	●	●	●	●	●	●	●	Get port status
	4	○	○	○	○	○	○	○	●	●	Extended initialize
	5	○	○	○	○	○	○	○	●	●	Extended communications port control
		2	2	2	2	2	2	2	4	4	Number of serial ports supported (four ports via extended control functions)
15 Cassette I/O	N/U	○	●	○	○	○	○	○	○	○	Not used on PC XT
	0	●	○	●	○	○	○	○	○	○	Turn cassette motor on
	1	●	○	●	○	○	○	○	○	○	Turn cassette motor off
	2	●	○	●	○	○	○	○	○	○	Read from cassette
	3	●	○	●	○	○	○	○	○	○	Write to cassette
and System services	0F	○	○	○	○	○	○	○	○	●	ESDI drive only; format unit periodic interrupt (invoked once per cycle)
	20	○	○	○	●	○	●	○	○	●	AL=10, setup of SYSREQ routine $ (software only)

Interrupt	Function code	PC	PC XT	PCjr	PC AT	XT /2	XT /286	PS CVT	PS/2 30	PS/2 other	BIOS	Comments
		○	○	○	●	○	●	○	○	●		AL=11, completion of SYSREQ function $ (software only)
	21	○	○	○	○	○	○	○	○	●		Power-on self-test error log
	40	○	○	○	○	○	○	●	○	○		Read or write system (AL= 0,1) or modem (AL= 2,3) profile
	41	○	○	○	○	○	○	●	○	○		Wait on external event
	42	○	○	○	○	○	○	●	○	○		Request system power off (system suspend)
	43	○	○	○	○	○	○	●	○	○		Read current system status information
	44	○	○	○	○	○	○	●	○	○		Activate or deactivate internal modem power
	4F	○	○	○	3x9	●	●	●	●	●		BIOS keyboard interrupt hexadecimal 9 intercept
	80	○	○	○	●	●	●	○	●	●		Device open $
	81	○	○	○	●	●	●	○	●	●		Device close $
	82	○	○	○	●	●	●	○	●	●		Program termination $
	83	○	○	○	●	○	●	●	●	●		Event wait
	84	○	○	○	●	●	●	○	●	●		Joystick support
	85	○	○	○	●	●	●	●	●	●		AL=0, system request key has been pressed $
		○	○	○	●	●	●	●	●	●		AL=1, system request key has been released $
	86	○	○	○	●	○	●	●	●	●		Wait (for *n* microseconds)
	87	○	○	○	●	○	●	○	●	●		Move block of memory (can move to and from extended memory)
	88	○	○	○	●	○	●	○	●	●		Return amount of memory above 1 megabyte
	89	○	○	○	●	○	●	○	●	●		Switch processor to virtual mode
	90	○	○	○	●	●	●	●	●	●		Device-busy loop $
	91	○	○	○	●	●	●	●	●	●		Interrupt complete flag set $
	C0	○	○	○	3x9	●	●	●	●	●		Return system parameter pointer; see table 1 for definitions
	C1	○	○	○	○	○	○	○	●	●		Return extended-BIOS data-area segment address
	C2	○	○	○	○	○	○	○	●	●		Pointing-device BIOS interface
	C3	○	○	○	○	○	○	○	○	●		Enable/disable watchdog time-out
	C4	○	○	○	○	○	○	○	○	●		Programmable option select
16	0	●	●	●	●	●	●	●	●	●		Read next ASCII character
Keyboard	1	●	●	●	●	●	●	●	●	●		Set Z flag if buffer not empty
	2	●	●	●	●	●	●	●	●	●		Read shift status
	3	○	○	●	339	○	●	○	●	●		Set typematic rates
	4	○	○	●	○	○	○	●	○	○		Turn on/off keyboard click
	5	○	○	○	339	●	●	○	●	●		Place ASCII character/scan code combination in keyboard buffer as if from keyboard
	10	○	○	○	339	●	●	○	●	●		Extended read interface for the enhanced (101-/102-key) keyboard
	11	○	○	○	339	●	●	○	●	●		Extended ASCII status for the enhanced (101-/102-key) keyboard
	12	○	○	○	339	●	●	○	●	●		Return extended shift status in AX register (101-/102-key keyboard)
17 Printer	0	●	●	●	●	●	●	●	●	●	NB	Print a character
	1	●	●	●	●	●	●	●	●	●	NB	Initialize printer port
	2	●	●	●	●	●	●	●	●	●	NB	Return printer-port status
		3	3	3	3	3	3	3	2	2		Number of printer ports supported
18 BASIC	N/A	●	●	●	●	●	●	●	●	●		Points to resident (ROM) BASIC
19	N/A	●	○	○	○	○	○	●	○	○		Bootstrap loader, IPL the system from a disk
	N/A	○	●	○	●	○	●	○	●	●		Bootstrap loader, IPL the system from a disk or hard disk
	N/A	○	○	●	○	○	○	○	○	○		Bootstrap loader, IPL the system from a disk or cartridge
1A	0	●	●	●	●	●	●	●	●	●		Read current clock setting
Time of day	1	●	●	●	●	●	●	●	●	●		Set the current clock
	2	○	○	○	●	○	●	●	●	●		Read the real-time clock
	3	○	○	○	●	○	●	●	●	●		Set the real-time clock
	4	○	○	○	●	○	●	●	●	●		Read date from real-time clock
	5	○	○	○	●	○	●	●	●	●		Set date into real-time clock
	6	○	○	○	●	○	●	●	●	●		Set the alarm (24 hour maximum, goes off and causes an INT 4A)
	7	○	○	○	●	○	●	●	●	●		Reset the alarm
	8	○	○	○	○	○	○	●	○	○		Set real-time clock alarm activated power-on mode
	9	○	○	○	○	○	○	●	●	○		Read alarm time and status

continued

Interrupt	Function code	PC	PC XT	PCjr	PC AT	XT /2	XT /286	PS CVT	PS/2 30	PS/2 other	BIOS	Comments
	0A	○	○	○	○	●	○	○	●	●		Read system-timer day counter
	0B	○	○	○	○	●	○	○	●	●		Set system-timer day counter
Sound select	80	○	○	●	○	○	○	○	○	○		Set up sound multiplexer
1B	N/U	●	●	●	●	●	●	●	●	●		Keyboard break address
1C	N/U	●	●	●	●	●	●	●	●	●		Timer tick
1D	N/U	●	●	●	●	●	●	●*	●	●		Video parameters for 6845 initialization
1E	N/U	●	●	●	●	●	●	●	●	●		Disk parameters
1F	N/U	●	●	●	●	●	●	●	●	●		Graphics character extension for modes 4, 5, and 6
20		○	○	○	○	○	○	○	○	○		DOS program terminate
21		○	○	○	○	○	○	○	○	○	NB	DOS function call
22		○	○	○	○	○	○	○	○	○		DOS terminate address (don't issue this INT)
23		○	○	○	○	○	○	○	○	○		DOS control-break exit address (don't issue this INT)
24		○	○	○	○	○	○	○	○	○		DOS fatal-error vector
25		○	○	○	○	○	○	○	○	○		DOS absolute disk read
26		○	○	○	○	○	○	○	○	○		DOS absolute disk write
27		○	○	○	○	○	○	○	○	○		DOS terminate, stay resident (instead, use INT21, AH=31)
28		○	○	○	○	○	○	○	○	○		Used internally by DOS
29		○	○	○	○	○	○	○	○	○		Reserved for DOS
2A		○	○	○	○	○	○	○	○	○		Used by NETBIOS
2B		○	○	○	○	○	○	○	○	○		Reserved for DOS
2C		○	○	○	○	○	○	○	○	○		Reserved for DOS
2D		○	○	○	○	○	○	○	○	○		Reserved for DOS
2E		○	○	○	○	○	○	○	○	○		Reserved for DOS
2F		○	○	○	○	○	○	○	○	○		NB Interprocess multiplex interrupt (see notes)
	1	○	○	○	○	○	○	○	○	○		Call resident part of PRINT
	2	○	○	○	○	○	○	○	○	○		Call resident part of ASSIGN
	10	○	○	○	○	○	○	○	○	○		Call resident part of SHARE
	B7	○	○	○	○	○	○	○	○	○		Call resident part of APPEND
30 to 3F		○	○	○	○	○	○	○	○	○		Reserved for DOS
40	N/U	○	●	○	●	●	●	○	●	●		Points to disk BIOS entry
41	N/U	○	●	○	●	●	●	○	●	●		Pointer to first hard disk, parameter block (not ESDI disks)
42	N/U	○	○	○	○	○	○	○	●	●	EGA	Points to screen BIOS entry
43	N/U	○	○	○	○	○	○	○	●	●	EGA	Pointer to EGA initializing parameters
44	N/U	○	○	○	○	○	○	●	●	●	EGA	Pointer to EGA graphics character table
	N/U	○	○	●	○	○	○	●	●	●		Pointer to lower graphics character table
45	N/U	○	○	○	○	○	○	○	○	○		Reserved
46	N/U	○	○	○	●	○	●	○	●	●	○	Pointer to second hard disk, parameter block (not ESDI disks)
47	N/U	○	○	○	○	○	○	○	○	○		Reserved
48	N/U	○	○	●	○	○	○	○	○	○		Cordless keyboard translation
49	N/U	○	○	●	○	○	○	○	○	○		Nonkeyboard scan-code translation table address
4A	N/U	○	○	○	○	○	○	●	●	●		Real-time clock alarm
4B to 4F	N/U	○	○	○	○	○	○	○	○	○		Reserved
50	N/U	○	○	○	○	○	○	○	○	○		Periodic alarm interrupt from timer
51 to 59	N/U	○	○	○	○	○	○	○	○	○		Reserved
5A	N/U	○	○	○	○	○	○	○	○	○	CLU	Cluster adapter BIOS-entry address
5B	N/U	○	○	○	○	○	○	○	○	○		Reserved
5C		○	○	○	○	○	○	○	○	○	NB	NETBIOS entry point
5D to 5F	N/U	○	○	○	○	○	○	○	○	○		Reserved
60 to 66	N/U	○	○	○	○	○	○	○	○	○		Reserved for user program interrupts

Interrupt	Function code	PC	PC XT	PCjr	PC AT	XT /2	XT /286	PS CVT	PS/2 30	PS/2 other	BIOS	Comments
67	N/U	○	○	○	○	○	○	○	○	○		Reserved for user program interrupts, LIM EMS interrupt entry
68 to 6B	N/U	○	○	○	○	○	○	○	○	○		Not used
6C	N/U	○	○	○	○	○	○	●	○	○		System resume vector
6D to 6F	N/U	○	○	○	○	○	○	○	○	○		Not used
70 (IRQ 8)	N/U	○	○	○	●	○	●	○	○	●		Real-time clock INT
71 (IRQ 9)	N/U	○	○	○	●	○	●	○	○	●		Redirected to IRQ2
72 (IRQ 10)	N/U	○	○	○	●	○	●	○	○	●		
73 (IRQ 11)	N/U	○	○	○	●	○	●	○	○	●		
74 (IRQ 12)	N/U	○	○	○	●	○	●	○	○	●		PS/2 others: mouse interrupt
75 (IRQ 13)	N/U	○	○	○	●	○	●	○	○	●		Coprocessor, BIOS redirect to NMI interrupt (INT 2)
76 (IRQ 14)	N/U	○	○	○	●	○	●	○	○	●		Hard disk controller
77 (IRQ 15)	N/U	○	○	○	●	○	●	○	○	●		
78 to 7F		○	○	○	○	○	○	○	○	○		Not used
80 to 85		○	○	○	○	○	○	○	○	○		Reserved by BASIC
86 to F0		○	○	○	○	○	○	○	○	○		Used by BASIC when the BASIC interpreter is running
F1 to FF		○	○	○	○	○	○	○	○	○		Reserved for user program interrupts

Notes:

tv = Topview function.
NB = NETBIOS alters this function.
EGA = Enhanced graphics adapter (and video graphics array) function.
N/U = Not used.
D = Used for diagnostics only.

○ = Not supported.
● = Supported.
●* = A superset is supported.
$ = These INT 15 functions are just operating system hooks. They perform no BIOS-level functions.
3x9 = Only on PC AT Models 319 and 339.
339 = Only on PC AT Model 339.

All PC AT interrupts are valid for real mode only.
The PC Portable, PC/370, and 3270 PC all use the PC XT BIOS. The AT/370 uses the PC AT BIOS.
PS/2 other systems are the Micro Channel systems and the Models 50, 60, and 80.
The Typematic rate of the 84-key PC AT keyboard is programmable, but no explicit BIOS support is provided. Also, the AT's keyboard has an internal 16-key buffer.
When a hard disk is present, the INT 13 disk interrupt is rerouted to INT 40 and INT 13 points to the hard disk BIOS.
For the multiplex interrupt (INT 2F), AH contains the identification of the routine to be called, where IDs 00-7F are reserved for DOS and C0-FF are reserved for user applications. AL contains the function code.

Table 3: *BIOS video modes.*

Mode	Type	Max colors	Alpha format	Buffer start	Display size	Box size	Max pages	Supporting cards
0, 1	A/N	16	40×25	B8000	320×200	8×8	8	PCjr, CGA, PC CVT, EGA, and PS/2 others
					320×350	8×14	8	EGA and PS/2 others
					320×400	8×16	8	PS/2 Model 30
					360×400	9×14	8	PS/2 others
2, 3	A/N	16	80×25	B8000	640×200	8×8	4	PCjr, CGA, and PC CVT
					640×200	8×8	8	EGA and PS/2 others
					640×350	8×14	8	EGA and PS/2 others
					640×400	8×16	8	PS/2 Model 30
					720×400	9×16	8	PS/2 others
4, 5	APA	4	40×25	B8000	320×200	8×8	1	PCjr, CGA, EGA, and all PS/2 systems
6	APA	2	80×25	B8000	640×200	8×8	1	PCjr, CGA, EGA, and all PS/2 systems

continued

Mode	Type	Max colors	Alpha format	Buffer start	Display size	Box size	Max pages	Supporting cards
7	A/N	Mono	80×25	B0000	720×350	9×14	1	MDA and PC CVT
					720×350	9×14	8	EGA and PS/2 others
					720×400	9×16	8	PS/2 others
							1	
							1	
							1	PCjr
F								
							1	
							1	
							1	

Notes

APA = All points addressable (i.e., graphics mode)

A/N = Alphanumeric (i.e., text-only mode)

CGA = Color Graphics Adapter

MDA = Monochrome Display Adapter

EGA = Enhanced Graphics Adapter

The cursor is not displayed in APA modes.

Modes 0, 2, and 5 are identical to modes 1, 3, and 4, except color burst is not enabled. (This doesn't affect RGB displays.)

Table 4: *Low-memory reserved addresses.*

00000-002FF	System interrupt vectors
00300-003FF	System interrupt vectors, power-on and bootstrap stack area
00400-004EF	BIOS data area
00400-00406	COM1: to COM4: I/O port base addresses, one word each.
00407-00408	LPT1: to LPT4:* I/O port base addresses, one word each.
00409-0040F	Reserved
00410-00411	Equipment flag word, returned in AX register by INT 11.

Bits	Meaning
14-15	Number of printers attached (0 to 3, LPTs)
13	Internal modem installed (PC CVT) or serial printer installed (PCjr)
12	Joystick installed
9-11	Number of COM devices (0 to 4, COMs)
8	Unused (PCjr only; DMA chip present on system)
6-7	Number of disk drives (if bit 0 is 1; 00=1, 01=2, 10 and 11 not used)
4-5	Initial video mode 00 Unused 01 40 × 25 BW using color card 01 80 × 25 BW using color card 11 Monochrome card
2-3	Unused, or, in the PC, old PC XT, and PCjr, planar RAM size; 00=16K bytes, 01=32K bytes, 10=48K bytes, 11=64K bytes
1	Math coprocessor installed (unused on PCjr and PC CVT)
0	IPL disk installed.

00412	Reserved, except in PC CVT power-on self-test status
00413-00414	Memory size in K bytes (0 to 640)
00415-00416	Reserved
00417	Keyboard Control

Bits	Meaning
7	Insert locked
6	Caps Lock locked
5	Num Lock locked
4	Scroll Lock locked
3	Alt key pressed
2	Control key pressed
1	Left shift key pressed
0	Right shift key pressed

00418	Keyboard Control

Bits	Meaning
7	Insert key pressed
6	Caps Lock key pressed
5	Num Lock key pressed
4	Scroll Lock key pressed
3	Pause locked
2	System request key pressed
1	Left Alt key pressed
0	Left Control key pressed

00419	Alternate keypad entry
0041A-0041B	Keyboard buffer head pointer
0041C-0041D	Keyboard buffer tail pointer
0041E-0043D	32-byte keyboard buffer
0043E-00448	Disk drive data area
00449-00466	Video-control data area 1
00467-0046A	Reserved, except PS/2 others, 00472=pointer to reset code upon system reset when memory is preserved.
0046B	Reserved
0046C-0046F	Timer counter
00470	Timer overflow
00471	Break key state
00472-00473	Reset flag

Bits	Meaning
1234	Bypass memory test
4321	Preserve memory (PS/2 other only)
5678	System suspended (PC CVT only)
9ABC	Manufacturing test mode (PC CVT only)
ABCD	System POST loop mode (PC CVT only)

00474-00477	Hard disk drive data area
00478-0047B	LPT1: to LPT4:* time-out values, 1 byte each.
0047C-0047F	COM1: to COM4: time-out values, 1 byte each.
00480-00481	Keyboard buffer start offset pointer
00482-00483	Keyboard buffer end offset pointer
00484-0048A	Video control data area 2
0048B-00495	Disk drive/hard disk drive control-data area (XT/2, AT, XT/286, and all PS/2s)
00496	Keyboard mode state and type flags
00497	Keyboard LED flags
00498-00499	Offset address to user wait complete flag
0049A-0049B	Segment address to user wait complete flag
0049C-0049D	User wait count in microseconds, low word
0049E-0049F	User wait count in ms, high word
004A0	Wait active flag **Bits Meaning** 7 Wait-time elapsed and posted flag 6-1 Reserved 0 INT 15, AH=86, Wait, has occurred.
004A1-004A7	Reserved
004A8-004AB	Pointer to video parameters and overrides
004AC-004EF	Reserved
004F0-004FF	Applications program communication area
500	Print screen-status flag
504	Single-drive mode status byte
00510-00521	Used by BASIC
00522-0052F	Used by DOS for disk initialization
00530-00533	Used by MODE command
00534-005FF	Reserved for DOS

* PS/2 systems don't support LPT4:.

Table 5: *Hardware interrupt request lines.*

Hardware interrupt request line	PC and PC/XT and PS/2 Model 30	PCjr	PC CVT	PC AT and XT/286	PS/2 Models 50, 60, 80
NMI	See notes	Keyboard interrupt	See notes	Parity errors	See notes
IRQ 0	Timer	Timer clock interrupt	Timer output 0	Timer output 0	Timer output 0
IRQ 1	Keyboard	I/O channel (reserved)	Keyboard (buffer full)	Keyboard (buffer full)	Keyboard (buffer full)
IRQ 2	Reserved	I/O channel	Reserved	Cascade for 8 to 15	Cascade for 8 to 15
IRQ 3	Serial port 2	Serial port 2	Serial port 2	Serial port 2	Serial port 2
IRQ 4	Serial port 1	Modem or serial port 1	Modem or serial port 1	Serial port 1	Serial port 1
IRQ 5	Hard disk (not PC)	Display vertical retrace	Reserved	Parallel port 2	Reserved
IRQ 6	Disk control	Disk	Disk control	Disk control	Disk control
IRQ 7	Parallel port 1	I/O channel (parallel printer)	Parallel port 1	Parallel port 1	Parallel port 1
IRQ 8*				Real-time clock	Real-time clock
IRQ 9				Redirected to IRQ2	Redirected to IRQ2
IRQ 10				Reserved	Reserved
IRQ 11				Reserved	Reserved
IRQ 12				Reserved	Mouse
IRQ 13				Coprocessor	Coprocessor
IRQ 14				Hard disk controller	Hard disk controller
IRQ 15				Reserved	Reserved

* Interrupts 8 to 15 are not available on the PC, PC XT, PCjr, PC CVT, and PS/2 Model 30.

Notes:
PC, PC XT, and PS/2 Model 30 use NMI for parity errors and numeric coprocessor interrupt.

PC CVT uses NMI for I/O channel check, disk power-on request, keyboard, real-time clock alarm, or system suspend.

PS/2 Models 50, 60, and 80 use NMI for parity errors, I/O channel check, watchdog timer, and arbitrator time-out.

IRQ 3 and 4 (except in the PC CVT) may be used by SDLC or bisynchronous communication cards instead of serial ports.

Table 6: *Expanded EMS function-call interfaces. This covers version 3.2 and is accessed via interrupt 67.*

AH =	Function
40	Get manager status
41	Get page frame segment
42	Get number of pages
43	Get handle and allocate memory
44	Map memory
45	Release handle and memory
46	Get EMS version
47	Save mapping context
48	Restore mapping context
49	Get I/O port address
4A	Get logical-to-physical-page mapping
4B	Get number of EMM handles
4C	Get pages owned by handle
4D	Get pages for all handles
4F-5F	Reserved
60	Get physical window array

Table 7: *Multitasking hooks using Interrupt 15.*

Wait Function (AH=90, AL=type code). Used to tell the OS task dispatcher to dispatch another task while the current task waits for its I/O operation to finish.
Post Function (AH=91, AL=type code). I/O operation complete, which can be used to inform the OS task dispatcher that an I/O operation for a waiting task has completed, and the task should now be moved to the ready queue.

Type codes for Wait and Post functions

00–7F Serially reusable device; OS must serialize access.
- 00 Disk (time-out)
- 01 Disk (time-out)
- 02 Keyboard (no time-out)
- 03 Pointing device (time-out)

80–BF Reentrant devices; ES:BS points to a unique control block.
- 80 Network (no time-out), ES:BX points to network control block.

C0–FF Wait-only calls, no complementary Post function. Time-out on nonoccurrence of event.
- FC Fixed-disk reset, PS/2 only (time-out)
- FD Disk drive motor start (time-out)
- FE Printer (time-out)

Table 8: *BIOS-extension (ROM-area) addresses.*

C0000-C3FFF	16K bytes EGA BIOS
C4000-C5FFF	
C6000-C63FF	256 bytes PGA communication area
C6400-C7FFF	
C8000-CBFFF	16K bytes hard disk BIOS
CC000-CFFFF	
D0000-D7FFF	32K bytes cluster adapter BIOS
D8000-DBFFF	
DC000-DFFFF	
E0000-EFFFF	64K bytes expansion ROM space (PC AT)
F0000-FFFFF	64K bytes ROM BASIC and simple BIOS

Note: PCjr uses D0000-DFFFF for expanded cartridges and E0000-EFFFF for standard cartridges.

Table 9: *NETBIOS modifications and additions to DOS functions. This covers version 1.10.*

INT	AH	AL	
21	3D		Open file with sharing specified
	44	09	IOCTL, is device redirected?
		0A	IOCTL, is handle local or remote?
		0B	IOCTL, change sharing retry count
	59		Get extended error
	5A		Create temporary file with unique name
	5B		Create new file
	5C	00	Lock byte range
		01	Unlock byte range
	5E	00	Get machine name
		02	Set up printer-control string
	5F	02	Get assign list entry
		03	Redirect device to network
		04	Cancel redirection
2A	00		NETBIOS installation check
	03		Get device-shared status
	04		Execute NETBIOS
	05		Get network resource information
	06		Network print-stream control
2F	87	00	Append installation check
		02	Append version check
	88	00	Network command installation check
		03	Get server post address
		04	Set server post address
		09	Network version check

```
MOV AL,'B'; Character to be
            printed, in this case
            a 'B'
MOV DX,0  ; Print it on LPT1
            (Printer number minus 1)
INT 17    ; Printer BIOS entry
            interrupt.
```

The BIOS is extensible. When the POST routines run, as part of their operation they scan the ROM address space for add-on routines, which are then invoked so that they can install themselves. The IBM EGA, for example, extends the video interrupt INT 10, as indicated in table 2.

The rule for BIOS entries is one software interrupt per device. There may also be one or more hardware entries, and entries that point to tables or blocks of data used by the device driver.

The interrupt vectors, used as pointers to data instead of code, allow easy alteration to the environment, such as changing the character set displayed for 80 to FF by the CGA.

According to IBM, the only time you safely bypass the BIOS is when you access the following I/O ports: 21–interrupt mask registers; 61–sound control; 40–, 41–(Note: Don't change this port.), 42–(timer frequency will remain fixed at 1.19 MHz), and 201–game control adapter.

Regarding absolute memory locations, note the following: Some functions have been added to interrupt vectors (0:0 to 3FF), but no functions have been redefined. The video display memory maps (A000:0, B000:0 and B800:0) will not change for a given video BIOS mode of operation. If the bit map is altered, a new mode is defined to support it. ROM BIOS data areas (starting at 40:0) will retain their current definitions as long as the corresponding functions are defined. In other words, the definitions can change at the whim of IBM. ■

SECRET CULT


SideKick Main Menu
1 2 3 Access System
Enter 1-2-3   Lotus Worksheet/Graphics/Database program
C:\LOTUS\NOTES.
249

CTX

THE BEST 14" MONITORS IN BOTH WORLDS

U.S. HEADQUARTERS: CTX INTERNATIONAL, INC.
260 PASEO TESORO/WALNUT, CA 91789
TEL: (714) 595-6146 FAX: (714) 595-6293

EASTERN REGIONAL OFFICE
CONTINENTAL TECHNOLOGY, INC.
300 McGAW DRIVE
EDISON, NEW JERSEY 08837
TEL: (201) 225-7377 FAX: (201) 225-6355

NEW ENGLAND AREA
MICRO-REP
141 BROOKS ST., SUITE NO. 1
BRIGHTON, MA 02135
TEL: (617) 254-1469 FAX: (617) 783-4877

FACTORY
CHUNTEX ELECTRONIC CO., LTD.
ROOM 401, NO. 50 SEC.1,
HSIN-SHENG S. RD.,
TAIPEI, TAIWAN, R.O.C.
TEL: (02) 3921171 FAX: (02) 3919780

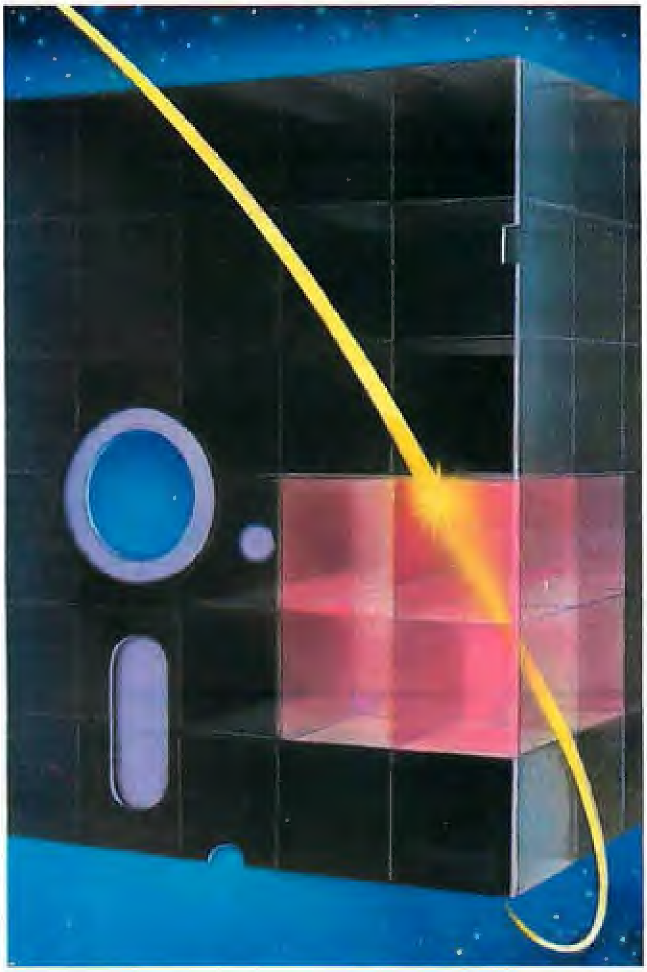
See us at COMDEX/Fall '87 November 2–6, 1987 Caesars Palace Las Vegas, Nevada

Circle 60 on Reader Service Card (Dealers: 61)

A new disk-allocation scheme in PC-DOS has important implications for long-term disk performance

Comparing Disk-Allocation Methods

Gregg Weissman



PC-DOS 3.0 and MS-DOS 3.0 and higher offer significant improvements to the speed and efficiency of disk-space allocation compared with previous versions. A careful study of the new and old methods, using simulation techniques, is instructive for operating-system designers and offers useful insights for anyone who uses a disk system for data storage.

Previous versions of DOS use a *first-fit* algorithm. Every time a new file is created or an existing file is extended, DOS starts looking for unused space at the beginning of the disk's FAT (file-allocation table), scanning forward until it finds a free cluster (the minimum unit of disk space that can be allocated). Version 3.0 and higher use the *next-fit* algorithm, in which DOS begins looking for a free cluster at the point where it last left off searching in the FAT.

In his book *The Art of Computer Programming, Vol. 1: Fundamental Algorithms* (2nd ed., Addison-Wesley, 1974), Donald E. Knuth uses these terms in relation to memory-allocation algorithms, but the implications of the methods are almost the same for memory and disk allocation. The only difference is that, when allocating disk space, DOS is not concerned about the fit of a requested block; clusters are allocated as they are found and chained together in the disk FAT. On the other hand, memory-allocation requests can fail if there are not enough bytes in any one contiguous block of memory to satisfy an allocation request.

Difficult mathematics are required for a theoretical analysis of the properties of first-fit and next-fit methods. Therefore, as Knuth did, I turned to simulation techniques. I wrote a program in Turbo Pascal to simulate disk activity and then ran simulations exploring the complex interactions of simulation and algorithm parameters.

The Algorithms Explained

The first step was to define the first-fit and next-fit algorithms.

Listing 1 is the pseudocode of the first-fit algorithm as I coded it in the simulations. The variable `FAT[]` is an array that represents a disk's FAT. `S` counts the number of times the routine has to look for a free cluster, a statistic used in analyzing performance. `Request` is the input parameter, the number of blocks that are requested.

The next-fit algorithm, presented in listing 2, is almost as simple. The only additions are a global or, if supported as in Turbo Pascal, a local static variable, `Last`, which points to the most recently used location in the FAT, and a little extra housekeeping to keep the FAT index within range.

Instead of beginning the search for free clusters at the beginning of the FAT each time, the simulation now starts just after the location where it last left off, as indicated by the `Last` pointer. When the index into the FAT reaches the end of the table, it is reset to the beginning.

The Simulation Goals

Before designing a simulation, you need to decide what you want to measure. Obviously, speed and efficiency are the key quantities, but how do you measure these?

Time to allocate space is directly proportional to the number of processor cycles needed to complete an allocation operation. Thus, the number of scan op-

continued

Gregg Weissman is a software consultant and partner at E-X-E Software Systems (205 East 78th St., New York, NY 10021).

erations (S in the programs) makes a suitable measure.

Fragmentation—the degree to which files are not in contiguous clusters on the disk—is another key factor in determining disk-access time; DOS and the disk drive have to do more work to read or write a given amount of data when the space is highly fragmented than when it is contiguous. Specifically, fragmentation is the ratio of contiguous clusters to total clusters allocated for every file on the disk.

I wanted the simulation to model the typical usage patterns of a personal computer user regarding the creation and deletion of files over time and the distribution of the user's file sizes.

I used a random uniform distribution of file creations and deletions. At each iteration of the simulation, I generated a uniformly distributed random number between 0 and 1 and performed a file create if the number was less than or equal to a threshold value. Otherwise, I deleted a file using a uniform random selection.

To model disk use over the long run, I designated upper and lower disk-capacity thresholds. When disk allocation exceeded the upper threshold, I decreased the file-create probability (p(create) in this article) to below 50 percent; this would free up space over time. When disk allocation fell below the lower threshold, I increased the file-create probability to above 50 percent to increase the disk utilization over time.

Another way to model space allocation/deallocation (one Knuth explores in his book) is to assign to each memory block (or file) a random lifetime; at each iteration in the simulation, blocks (or files) whose lifetimes have expired are deleted. Interested readers might wish to explore how the lifetime distribution affects performance of the first-fit and next-fit allocation methods.

Determining file-size distribution was more difficult. Examining the file sizes on my own hard disk showed the distribution represented in figure 1, with file sizes shown as a percentage of total disk capacity. As an approximation of this distribution, I assumed an exponential distribution

$$r = -\log(x) \times M,$$

where x is a uniformly distributed random variate between 0 and 1, and M is the desired mean file size.

Listing 1: *Pseudocode for the first-fit disk-allocation method.*

```
IF Request > Blocks_Free THEN exit {Create fails}
Index := 1; {Start of file-allocation table}
WHILE Request > 0 DO
 BEGIN
  IF FAT[Index] = Free THEN
   BEGIN
    FAT[Index] := Allocated;
    Request := Request - 1
   END
  ELSE
   S := S + 1; {Scans for free blocks, global variable}
  Index := Index + 1
 END {end while}
```

Listing 2: *Pseudocode for the next-fit disk-allocation method.*

```
IF Request > Blocks_Free THEN Exit {Create fails}
Index := Last; {Last is global, initialized to 1}
WHILE Request > 0 DO
 BEGIN
  IF FAT[Index] = Free THEN
   BEGIN
    FAT[Index] := Allocated;
    Request := Request - 1
   END
  ELSE
   S := S + 1; {Scans for free blocks, global variable}
  Index := Index + 1;
  IF Index > FAT_Size THEN
  Index := 1;
 END {end while}
Last := Index;
```

Listing 3: *Pseudocode for the allocation time simulation.*

```
For each iteration in the simulation DO
 BEGIN
  Adjust p(create) according to simulation criteria
  Select Delete or Create with uniform distribution
  IF Create THEN {RAND <= p(create)}
   BEGIN
    Create a file of exponentially distributed size with mean M
    Return scans_for_free_blocks
    If insufficient room or directory full, GOTO Fail
    u := Percent_of_disk_used
    MS[u] := MS[u] + Scans_for_free_blocks/Blocks_allocated
    Count[u] := Count[u] + 1
   END
  ELSE {it is a delete:}
   Select a created file with uniform probability and delete it
 Fail: (Come here to skip a creation that is impossible)
 END
{Simulation done: output results}
FOR u := 1 TO 100 DO
 WRITELN(MS[u]/Count[u])
```

Time Simulation

The first set of simulations explored the speed of the two algorithms, answering the question: "How do the two methods compare in speed as the disk fills up?" More precisely, I wanted to plot the mean number of scans for a free block per block allocated, as a function of the percentage of the disk used. Listing 3 gives the pseudocode for the time simulation.

To ensure an adequate sample population in each percentile of disk space used, I set the number of iterations to 5000. I varied the mean file size to examine its impact on performance and defined the FAT size as 1000 blocks. When the FAT was greater than 99 percent full,

continued

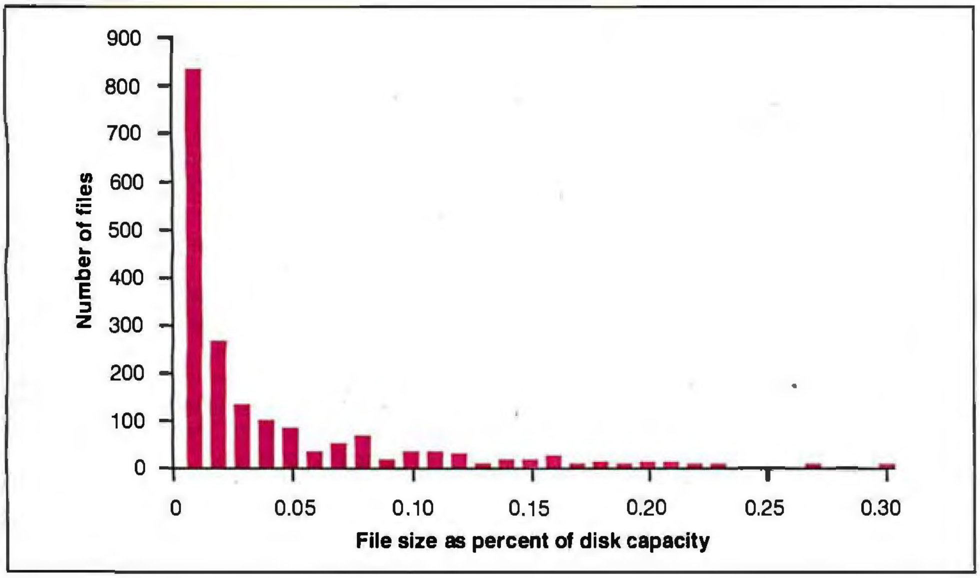


Figure 1: *Typical file-size distribution over a hard disk, expressed in percent of hard disk capacity.*

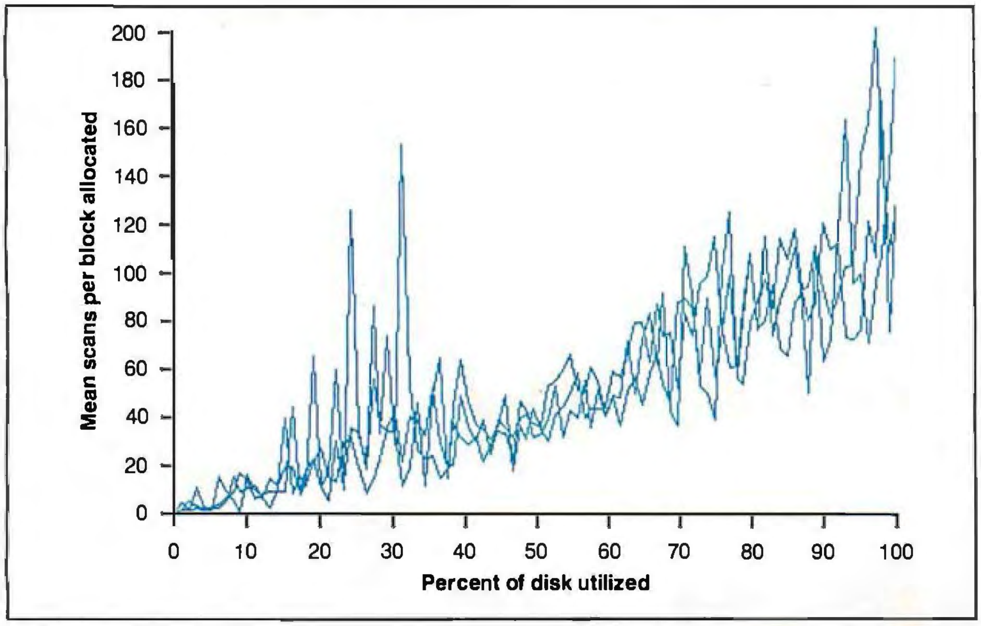


Figure 2: *A simulation plot showing the mean number of scans required per block allocated as a function of disk utilization; three runs are plotted. The first-fit algorithm was used. Note the direct linear relationship between disk utilization and scans required. In this simulation, disk capacity is 1000 blocks, and the mean file size is 1 percent of disk capacity.*

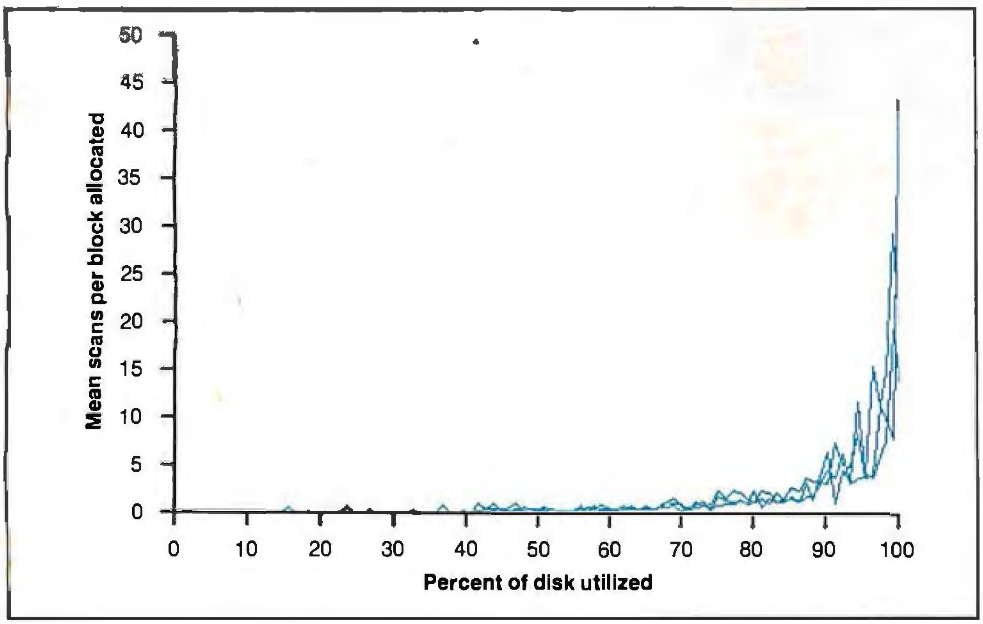


Figure 3: *The same as figure 2, but the next-fit algorithm was used, and mean file sizes for three runs are 0.5, 1, and 2 percent of disk capacity. Note that the number of scans remains at a constant low level until the utilization reaches 90 percent.*

p(create) was set to 0.25. When the FAT utilization dropped below 50 percent, p(create) was set to 0.55. Therefore, most activity took place with the disk between 50 and 100 percent full.

Figure 2 shows a plot of mean scans per block allocated as a function of the percentage of disk space used, for three runs of the simulation using the first-fit algorithm. The mean file size was 1 percent of disk space.

The trend of each plot is linear, indicating a direct relationship between disk space used and the time required to allocate more. I ran simulations with different mean file sizes and found that the rate of time increase was fastest with the smaller file sizes. Regression analysis gave slope coefficients of 1.56, 1.05, and 0.69 for mean file sizes of 0.5, 1, and 2 percent, respectively. The intuitive explanation for this inverse relationship is that larger files leave larger holes when they are deleted, making it easier to subsequently find free clusters.

Other simulations not illustrated here show that the rate at which performance degrades is directly proportional to FAT size. That's to be expected: The larger the FAT, the greater the number of clusters that must be searched.

Figure 3 shows the results for the same conditions as in figure 2, using the next-fit allocation scheme. The difference is extreme: The speed of allocation barely degrades until the disk is almost full. Even when the disk is almost completely filled (99 percent), the worst-case performance is better than first-fit by a factor of 10. This result is consistent with Knuth's results for memory allocation.

When only one free cluster is left, the average number of scans will be half the FAT size and may be as much as the FAT size itself. This explains the steep knee of the curve as disk utilization approaches 100 percent.

Subsequent tests I performed showed that the next-fit performance is unaffected by mean file size.

Listing 4: *Pseudocode for the fragmentation simulation.*

```
For each iteration in the simulation do
  BEGIN
   Adjust p(create) according to simulation criteria
   Select Delete or Create with uniform distribution
   IF Create THEN{RAND <= p(create)},
    BEGIN
     Create a file of exponentially distributed size with mean M
     IF insufficient room or directory full, GOTO Fail
     IF Iteration MOD 100 = 0 then
      WRITELN(Iteration,Percent_contiguous, percent_utilized)
    END
   ELSE {It is a delete}
    Select a created file with uniform probability and delete it
  Fail: {Come here to skip a creation that is impossible}
  END {end while}
```

Fragmentation Simulation

The question to be answered by the fragmentation simulation was: "After a large

continued

The *JET*-SETTER, laser printer.

The secret to my success.

In business, your first impression may be your last.

That's why I purchased the C. Itoh Jet-Setter laser printer. It's the first personal laser designed for achievers like me. And it's affordable!

First impressions are lasting impressions.
When I first started using the Jet-Setter to print daily reports, memos and budgets, my associates couldn't believe how I instantly gained esteem from top management.

But what really sold them on my talents was when I closed an account the firm had been working on for months with a business presentation printed on—you guessed it—the Jet-Setter.

Now everyone wants to get in on the action. And wait until they find out how affordable Jet-Setters are. They won't even need to purchase any additional software since it's compatible with the HP LaserJet™ series and uses optional Diablo™ and Epson™ emulation cartridges.

Anyone can use it.
My secretary already has an eye on the Jet-Setter to handle the volume of text generated on WordPerfect™ and other word processing files. And you should have seen the response I got when I presented my quarterly report with eye-opening charts and graphs instead of just numbers. Even our communications manager wants to purchase the Jet-Setter to publish internal newsletters using Ventura Publishing™ or PageMaker,™ desktop publishing packages.

With the C. Itoh Jet-Setter laser printer you can create professional business letters, presentations and newsletters.

The Jet-Setter grows with your business.
As more people catch on to the Jet-Setter revolution in my office, they will discover even more exciting features this printer offers, like its large library of font cartridges and expansion memory for full page graphics at 300 dots per inch. The secret to my success? The Jet-Setter from C. Itoh. How long can you afford to be without one of your own?

Helv, 10 point, pro
Roman, 10 point, p
Roman Italic, 10 po
Roman Bold, 10 po

For the C. Itoh dealer nearest you, call:
Toll-free (800) 423-0300

C.ITOH

C. Itoh Digital Products, Inc.
19300 S. Hamilton Av., Ste. 110 • P.O. Box 9085
Torrance, CA. 90508 • (213) 327-2110

Circle 67 on Reader Service Card (Dealers: 68)

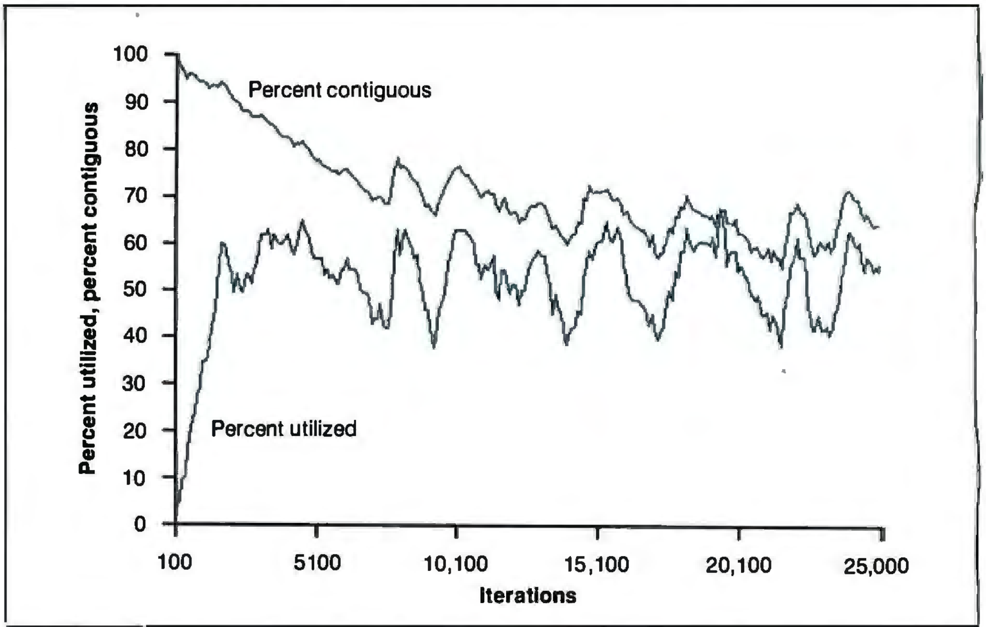


Figure 4: *File-fragmentation patterns over the course of 25,000 file-create/delete operations, using the first-fit algorithm. Note how quickly the percent contiguous drops to the 60 to 70 percent range. Disk capacity is 10,000 blocks, and the mean file size is 0.2 percent of disk capacity.*

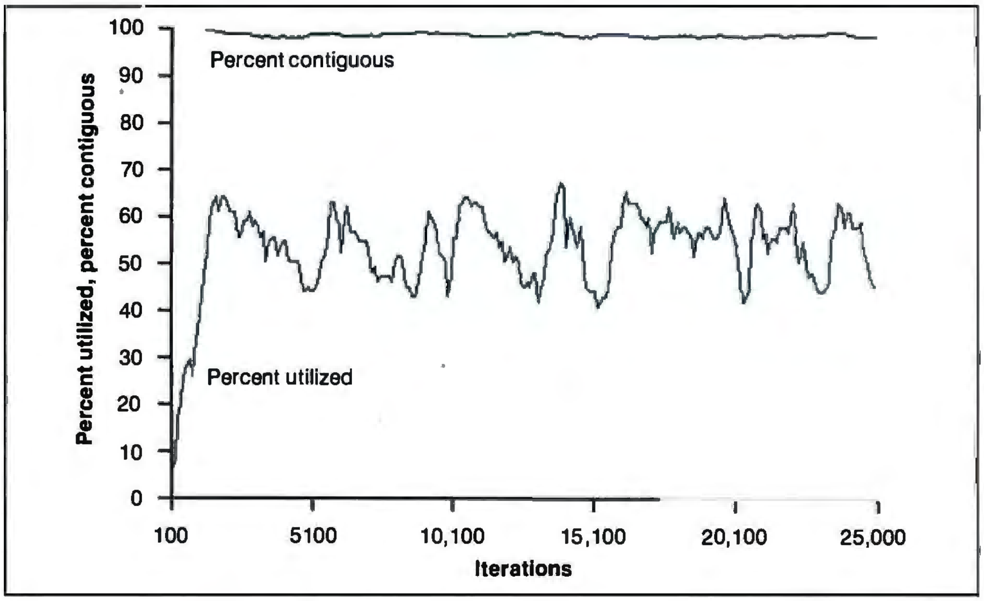


Figure 5: *The same as figure 4, but the next-fit algorithm was used. Percent contiguous remains at a constant high level under the usage pattern selected for this simulation. With other usage patterns (not shown), percent contiguous drops to 90 percent but remains relatively constant.*

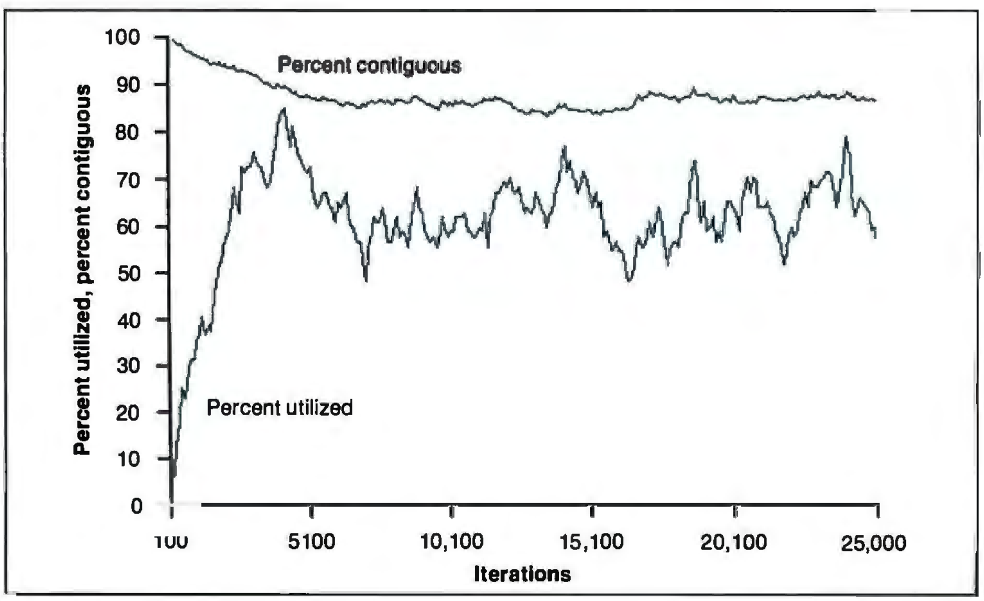


Figure 6: *In modified next-fit simulation, resetting the last-position pointer after every 100th iteration brings the percent contiguous down to around 85 percent, in close agreement with measurements on an actual disk.*

amount of random file creations and deletions, which algorithm minimizes file fragmentation?" To see the answer, I plotted the percentage of all allocated clusters that are contiguous as a function of the total number of disk operations.

To collect the data, I calculated the percent of contiguous clusters every 100 iterations and wrote the value to a disk file to analyze later. Listing 4 is the pseudocode for the fragmentation simulation.

All simulations ran for 25,000 iterations and used a mean file size of 20 blocks, or 0.2 percent of a 10,000-block FAT. I repeated the simulations for different usage patterns by varying `p(create)`, the probability of file creation, and the upper and lower thresholds at which `p(create)` is adjusted.

Figures 4, 5, and 6 show the simulation results for first-fit and next-fit allocation schemes. The x axis represents the number of iterations. Plotted against x are the percent of disk space used and the percent contiguous.

With the first-fit method (see figure 4), files become more fragmented (i.e., the percent contiguous decreases) fairly steadily as the number of file-create/delete operations increases. After 25,000 iterations, the percent contiguous drops to around 60 percent. My hypothesis says that as the number of iterations increases without bound, the percent contiguous stabilizes at around 50 percent.

With the next-fit method, the percent contiguous stays in the region of 90 percent and up, almost unperturbed by the random file-creation and file-deletion activity (see figure 5).

A Reality Check

The simulated next-fit performance seemed almost too good to be true, prompting me to verify the situation. I obtained statistics for my own hard disk, a 20-megabyte drive running under PC-DOS 3.1, and found that out of 7084 allocated clusters, 5858 were contiguous; the percent contiguous was 82.7. The disk had been in use for about a year without reformatting and was 68 percent full, so it should have been less fragmented according to the simulations.

However, the variable in memory that DOS uses to point to the next available cluster (`Last` in the pseudocode) is reset each time you turn off the computer—or, in the case of removable media, when you remove the disk. In effect, the allocation scheme reverts to first-fit whenever you turn off the computer or remove media. This accounts for a somewhat reduced percent contiguous.

I ran another set of simulations in which `Last` is reset every 100 iterations, equivalent to shutting a computer down after every 100 file-create and file-delete operations. Figure 6 shows the results of this modified simulation. Fragmentation reaches and remains at around 85 percent—close to the actual conditions of my own hard disk.

A Good Fit

The simulation indicates that operating-system designers have an easy choice of disk-allocation algorithms: Next-fit is the way to go. For hard disk users, the conclusion is just as obvious: Upgrade to DOS 3.0 or higher.

A number of utility programs offer to optimize your disk performance by reorganizing your files to make them contiguous. However, if your operating system uses the next-fit allocation scheme, you might not find such utilities beneficial. Since fragmentation will reach the 20 percent level fairly quickly, you gain little by restructuring the entire disk unless you do it fairly often. Considering that safe operation of these utilities requires making a full disk backup, the minor and short-lived performance improvement might not warrant the effort. ■

Announcing BYTE's New Subscriber Benefits Program

Your BYTE subscription brings you a complete menu of the latest in microcomputer technology every 30 days. The kind of broad-based objective coverage you read in every issue. *In addition*, your subscription carries a wealth of other benefits. Check the check list:

BONUSES

- ☑ *Annual Separate Issues:* In addition to BYTE's 12 monthly issues, subscribers also receive our annual IBM PC issue free of charge, as well as any other annual issues BYTE may produce.
- ☑ *BYTE Deck:* Subscribers receive five BYTE postcard deck mailings each year—a direct response system for you to obtain information on advertised products through return mail.
- ☑ *Reader Service:* For information on products advertised in BYTE, circle the numbers on the Reader Service card enclosed in each issue that correspond to the numbers for the advertisers you select. Drop it in the mail and we'll get your inquiries to the advertisers.
- ☑ *TIPS:* BYTE's Telephone Inquiry System is available to subscribers who need *fast response*. After obtaining your Subscriber I.D. Card, dial TIPS and enter your inquiries. You'll save as much as ten days over the response to Reader Service cards.
- ☑ *BYTE's BOMB:* BYTE's Ongoing Monitor Box is your direct line to the editor's desk. Each month, you can rate the articles via the Reader Service card. Your feedback helps us keep up to date on your information needs.

- ☑ *Subscription Service:* If you have a problem with, or a question about, your subscription, you may phone us during regular business hours (Eastern time) at our toll-free number: 1-800-423-8912 (in N.J., 201-981-1963). You can also use Subscription Service to obtain back issues and editorial indexes.

PAID SERVICES

- ☑ *BIX:* BYTE's Information Exchange puts you on-line 24 hours a day with your peers via computer conferencing and electronic mail. All you need to sign up is a microcomputer, a modem, and telecomm software. For further information and cost call 1-800-227-BYTE.
- ☑ *Program Listings:* Listings of programs that accompany BYTE articles are now available on BIX, on disks or in quarterly printed supplements (see reply cards in this issue for cost information), or call 1-800-258-5485.
- ☑ *Microform:* BYTE is available in microform from University Microfilm International in the U.S. and Europe. See Contents page for cost information.

DISCOUNTS

- ☑ 13 issues instead of 12 if you send payment with subscription order.
- ☑ One-year subscription at $22 (50% off cover price).
- ☑ Two-year subscription at $40.
- ☑ Three-year subscription at $58.
- ☑ One-year group subscription for ten or more at $18.50 each. (Call or write for details.)

TOLL-FREE NUMBERS FOR YOUR CONVENIENCE:

Subscriptions & Back Issues: 1-800-423-8912 (in N.J., 201-981-1963)

BIX: 1-800-227-BYTE

Program Listings Orders: 1-800-258-5485

And . . . welcome to BYTE country!

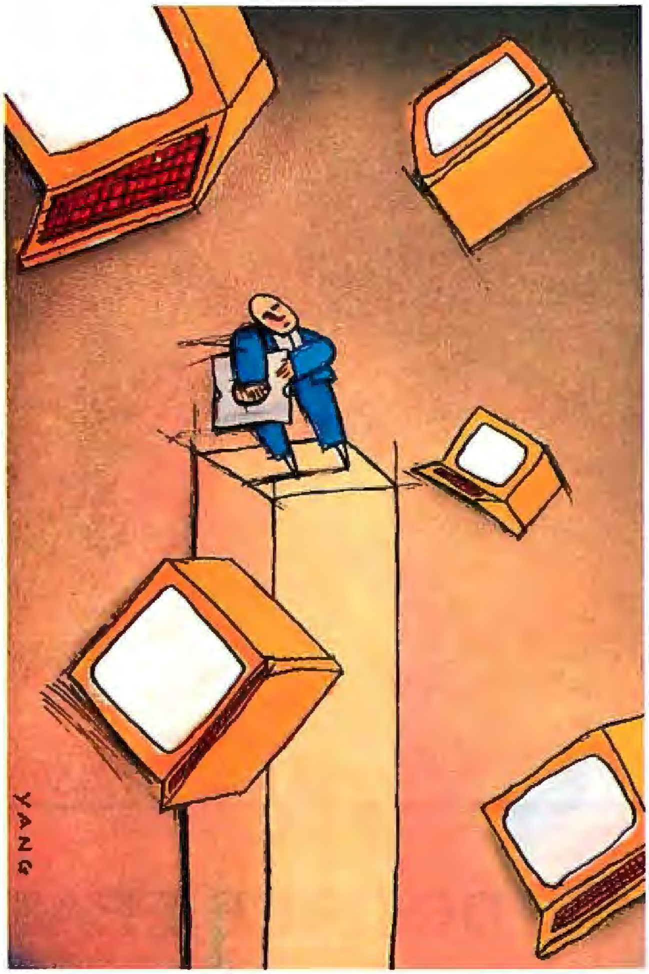
BYTE
THE SMALL SYSTEMS JOURNAL

McGraw-Hill

*Multidimensional analysis:
a new way to assess computer performance*

Rating the IBM Compatibles

Robert G. Brookshire



A potential buyer of an IBM-compatible computer faces a bewildering variety of claims about features, capability, and performance. Despite their similarity to the IBM family of computers, these machines differ in a number of dimensions, making comparisons between them difficult. I will present a summary ranking of 35 IBM-compatible computers based on a multidimensional scaling analysis of seven of BYTE's standard measures of performance. You should not use this ranking as the sole basis for evaluating these computers, but it will help to simplify the complex task of comparing them.

Measures of Computer Performance

Since June 1984, BYTE has used a standard set of benchmark tests in its personal-computer reviews. Two tests, the Sieve and Calculation benchmarks, measure the CPU's speed. Two others measure the speed with which the computer writes and reads a 64K-byte file to and from disk. A fifth test compares the times required to copy a 40K-byte file to a floppy disk using the COPY command. The reviewer conducts each test three times and averages the results. I did not use the results of the Disk Copy test in this analysis, since this test is not applied to all machines. [Editor's note: *For more details on these benchmarks, see "Benchmarking the Clones" by Jon R. Edwards and Glenn Hartwig in BYTE's* Inside the IBM PCs, *Fall 1985.*]

Two additional benchmarks test spreadsheet performance (using Microsoft Multiplan). These tests measure the speed with which the computers load and recalculate a spreadsheet with 25 rows and 25 columns, in which each cell is 1.001 times the cell to its left. Table 1 shows the results of the seven benchmarks, as well as summary statistics for 35 personal computers that use the MS-DOS or PC-DOS operating systems.

I omitted the Hewlett-Packard 110 and the Stearns Desktop Computer because not all their benchmark results were published in BYTE. Results for the IBM PC, which are routinely reported in each review, are included. The model number and clock speed of the CPUs used in these computers (as reported in the reviews) is also in table 1.

The seven benchmark tests give seven separate measures of performance for personal computers. Although the result of each test is informative in itself, it is difficult to compare two or more computers that have seven sources of performance variation. BYTE does not provide a summary measure of performance based on these tests. It is possible, however, to compare and summarize the benchmark results for a large number of computers by using multidimensional scaling.

The Method

Multidimensional scaling is a mathematical technique for representing the configuration of variates in one-, two-, or higher-dimensional space (see references 1 and 2 for a thorough explanation of the method). It is applied in the social and behavioral sciences to reveal the structure that underlies relationships between objects, or to reduce the dimensionality of a set of variables. The data that is input to a multi-

continued

Robert G. Brookshire holds a Ph.D. in political science from Emory University. He is director of academic computing at James Madison University and can be reached at the Academic Computing Center, James Madison University, Harrisonburg, VA 22807.

Table 1: *Benchmark test results and CPU characteristics for 35 personal computers. All times are in seconds.*

Computer	Disk Write	Disk Read	Sieve	Calculation	File Copy	Spreadsheet Load	Spreadsheet Recalculate	CPU Type	Clock Speed (MHz)
Epson Equity III	25.0	24.0	54.0	16.0	4.7	6.2	2.8	80286	8
Western AT	26.0	24.0	43.0	13.0	4.2	6.2	2.5	80286	8
Zenith Z-241	26.0	25.0	57.0	17.0	5.7	6.3	3.0	80286	6
Zenith Z-248	27.0	25.0	43.0	13.0	4.8	6.0	2.5	80286	8
IBM PC	56.0	46.0	191.0	69.0	5.8	8.1	11.0	8088	4.77
Epson Equity I	56.0	54.0	191.0	58.0	7.7	9.5	9.9	8088	4.77
Kaypro PC	55.0	30.0	186.0	57.0	8.2	8.6	10.0	8088	4.77
MPF-PC/700 D1	31.0	29.0	114.0	34.0	7.6	7.7	6.1	8088-2	8
Compaq Portable II	31.0	29.0	54.0	16.0	5.8	6.5	2.8	80286	8
Leading Edge Model D	56.0	30.0	190.0	58.0	8.2	8.1	10.0	8088	4.77
Xerox 6060	22.0	14.0	85.0	26.0	6.7	6.6	4.6	8086	N/A
NCR PC6	31.0	30.0	113.0	34.0	6.0	8.1	6.0	8088-2	8
Sperry PC/IT	9.0	8.8	44.0	14.0	.7	1.2	2.0	80286	7.16
ITT XTRA XP	10.0	10.0	60.0	20.0	4.8	2.9	3.2	80286	6
Conquest Turbo PC	30.0	30.0	130.0	40.0	9.5	9.4	7.5	8088-2	8
Deskpro 286	26.0	30.0	54.0	16.0	9.1	5.7	2.9	80286	8
Tele-286	25.0	24.0	54.0	16.0	3.5	4.9	2.8	80286	8
Executive Partner	31.0	29.0	90.0	28.0	9.1	6.1	5.1	8086-2	7.16
Kaypro 286i	24.0	23.0	73.0	22.0	8.2	7.3	4.1	80286	6
Canon A-200	57.0	29.0	132.0	41.0	13.0	8.0	7.4	8086	4.77
Color Fox	35.0	30.0	297.0	94.0	11.0	13.0	17.0	8088	3.6
AT&T PC 6300	32.0	30.0	87.0	27.0	10.0	7.0	4.9	8086-2	8
Data General/One	56.0	55.0	229.0	69.0	12.0	26.0	14.0	80C88	N/A
Sanyo MBC-775	30.0	29.0	154.0	35.0	8.0	7.7	6.3	8088-2	8
Ericsson	57.0	31.0	182.0	56.0	9.3	9.4	10.0	8088	4.77
Portable STM	31.0	30.0	80.0	24.0	5.7	5.1	3.7	80186	8
Kaypro 16	57.0	30.0	184.0	56.0	7.3	8.8	10.0	8088	N/A
Osborne 3	59.0	56.0	273.0	83.0	15.0	11.0	23.0	80C86	3.5
Tandy 1000	56.0	55.0	226.0	68.0	9.9	9.0	11.0	8088	4.77
TI Pro-Lite	34.0	33.0	155.0	51.0	11.0	7.2	9.1	80C88	5
Mindset	58.0	55.0	301.0	54.0	12.0	8.4	9.9	80186	6
Compaq Deskpro	30.0	29.0	93.0	29.0	7.4	6.1	5.1	8086	7.14
IBM PC AT	26.0	24.0	80.0	27.0	3.9	5.5	4.1	80286	6
ITT XTRA	33.0	32.0	185.0	56.0	8.8	10.0	10.0	8088	5
NEC APC III	30.3	29.4	86.0	28.9	6.5	8.4	6.7	8086-Z	8
Mean	36.5	31.2	130.6	39.0	7.8	7.9	7.2		
Standard Deviation	14.8	11.8	75.6	21.8	3.0	3.8	4.6		

Table 2: *Multidimensional scaling results for 35 personal computers.*

Computer	Rank	Coordinate
Sperry PC/IT	1	1.8452
ITT XTRA XP	2	1.4024
Tele-286	3	1.0418
Western AT	4	1.0104
Zenith Z-248	5	.9886
Epson Equity III	6	.9317
IBM PC AT	7	.7919
Zenith Z-241	8	.7868
Xerox 6060	9	.7659
Compaq Portable II	10	.7341
Compaq Deskpro 286	11	.6769
Kaypro 286i	12	.6245
Portable STM	13	.6085
Compaq Deskpro	14	.4188
Panasonic Executive Partner	15	.3473
NEC APC III	16	.3368
AT&T PC 6300	17	.2956
NCR PC6	18	.2876
Multitech MPF-PC/700 D1	19	.2333
Sanyo MBC-775	20	.1258
Conquest Turbo PC	21	−.0368
TI Pro-Lite	22	−.2730
ITT XTRA	23	−.3715
Canon A-200	24	−.4747
Kaypro 16	25	−.5154
Kaypro PC	26	−.5270
Leading Edge Model D	27	−.5312
Ericsson	28	−.6030
IBM PC	29	−.8110
Epson Equity I	30	−.9019
Tandy 1000	31	−1.2126
Mindset	32	−1.4164
Color Fox	33	−1.6591
Osborne 3	34	−2.4205
Data General/One	35	−2.4999

Young's S-Stress formula: 1: 0.086
Kruskal's Stress formula: 1: 0.112
R squared: 0.967

dimensional scaling program is a measure of either similarity or dissimilarity between each pair of objects or variables that are to be scaled.

For this analysis, I calculated measures of dissimilarity between each pair of personal computers. I standardized each of the seven performance measures for each computer by converting it to a Z score, taking the difference between each test result and the average of that benchmark for all the machines, and dividing by the standard deviation. I then calculated a Euclidean distance measure for each pair of computers by taking the square root of the sum of the squared differences between the Z scores. Next, I used Kruskal's nonmetric multidimensional scaling method (see references 3 and 4) as implemented in the ALSCAL procedure of the SPSS^x statistical package to reduce the resulting matrix of dissimilarities to one dimension.

Table 2 presents the results of the analysis. It reports each computer's coordinate on the dimension derived from the multidimensional scaling analysis, as well as the rank of the coordinate. The values for the measures of stress and R squared indicate a good fit of the model to the data; not much violence has been done to the data by reducing it to one dimension. Sperry's PC/IT is the best performer, while the Data General/One is the slowest overall.

The Results

The first dozen computers in table 2, with the exception of the Xerox 6060, are based on the Intel 80286 CPU. This processor is considerably faster than the Intel 8088 used in the IBM PC and most of the other computers in the analysis, and this accounts for the high performance rankings of these machines.

Within the 80286 machines, some of the factors that explain their relative rankings are the clock speed at which the CPU operates, the use of wait states by the

continued

CPU to synchronize timing with the memory chips or other devices, and the speed of the disk drives, all of which will affect the performance benchmarks. The Sperry PC/IT, for example, runs at a clock speed of 7.16 megahertz with no wait states and has a disk drive with a 30-millisecond average seek time. The IBM PC AT, in contrast, was benchmarked running at a 6-MHz clock speed with one wait state and using a hard disk drive with a 40-ms average access time, while the Zenith Z-248 used an 8-MHz clock with no wait states and a hard disk drive slightly faster than that of the PC AT. These factors, and probably others, have contributed to the large variation in the performance of the 11 IBM PC AT–compatible machines analyzed here.

Of the next five ranking computers, four use variants of Intel's 8086 CPU, while the Portable STM uses the 80186. Since the 80186 can be considered an intermediate chip between the 8086 and the 80286, the placement of the STM in the ranking is not surprising. It appears, based on its coordinate, to be more similar in performance to the 80286 machines than the 8086-based computers.

The Xerox 6060, which uses the 8086, outperforms all the other 8086 computers, as well as several of the 80286-based systems. This is due to the machine's rapid execution of the disk-performance benchmarks. Among the other 8086 machines, it is interesting that the AT&T and NEC computers, although operating at a clock speed of 8 MHz, are slower in overall performance than the Compaq Deskpro and Panasonic systems, which run at slower clock speeds. The machines with faster clock speeds did execute some benchmark tests more rapidly than the Compaq and Panasonic systems but were slower on other measures. This illustrates the usefulness of a multivariate analysis, which can summarize all seven performance measures simultaneously.

On the other hand, the coordinates on the derived dimension indicate no tremendous difference in the performance of these four 8086 machines (AT&T, NEC, Compaq Deskpro, and Panasonic). The computers ranking from 18 to 21 use Intel's 8088-2 processor, which can be set at either a 4.77- or 8-MHz clock speed. All benchmark results used in this analysis were those for the faster clock speed. Since these four machines use the same processor and clock speed, one might expect to find similar coordinates on the derived dimension due to similar results on the benchmark tests.

This is not the case. Considerable variation in the coordinates reflects variation on the underlying measures. As table 1 shows, these machines differ most on the Sieve benchmark, which measures processor speed. Even for computers with the same processor and clock speed, performance can differ significantly.

Nine of the remaining 14 computers use Intel's 8088 processor, including the IBM PC. As table 2 indicates, the IBM PC has relatively poor performance when compared with the other machines in this analysis. Even among the nine machines using the 8088, the IBM ranks sixth in speed. On the other hand, the IBM PC is the oldest machine on the list, and there are few dramatic differences between it and the other machines of its type.

The two lowest-ranking machines, the Data General/One and the Osborne 3, and the higher-ranking TI Pro-Lite, are "laptops," small and light enough to be held in your lap. Most laptop computers use variants of the Intel 80C88 processor, a CMOS version of the 8088 that uses significantly less electrical power than standard microprocessors, making it more suitable for battery-powered computers.

What you give up in exchange for portability can be performance. The TI Pro-

continued

← *Circle 89 on Reader Service Card*

Thirty 4000-line slides per hour.

Boardroom Quality Slides at Your Fingertips—Introducing the PFR Personal Film Recorder™ from Lasergraphics. Its 4000-line resolution and 30-slide per hour output make your PC a unique graphics presentation center that combines speed and convenience with complete confidentiality. And at only $4995, PFR can pay for itself in less than a year even if you make as few as 10 high quality slides a week.

Virtually all popular CADD or business graphics PC packages are compatible with PFR, driven by the powerful Rascol® II rasterizer (included). Even scanned images in TARGA® format can be incorporated, to put a whole new presentation tool in your hands. Only from Lasergraphics—the innovator in raster processors and film recorders.

Call 714-660-9497 for a free sample slide.

LASERGRAPHICS™

Turn your PC into a high-resolution graphics workstation for under $5000.

Circle 119 on Reader Service Card (Dealers: 120)

Lite performs slightly better than its 8088-based cousins, but the Data General and Osborne computers are extraordinarily slow. The other two slow systems, the Color Fox and the Mindset, have enhancements for graphics applications that apparently degrade performance.

Other Considerations

You should take many factors into consideration when purchasing a personal computer, including price, warranty terms, and availability of repair service. Also, you must consider software compatibility, accessibility of training and consultation, and performance.

The performance dimension has several sources of variation, however. These include the type of CPU used in the computer, the CPU's clock speed, the use of wait states, and the speed of the disk drives. All these sources of variation interact to affect performance in ways difficult to evaluate.

Using standard benchmarking programs is one approach to comparing the performance of computers. However, when you use several benchmarks, the effect can be to provide additional sources of variability at a higher level of abstraction. I've used multidimensional scaling to summarize seven benchmark tests and yield a rank order of 35 personal computers based on performance.

The ranking shows that, in general, computers using the Intel 80286 processor are the fastest, followed by those using the 8086, the 8088-2, and the 8088, with laptop computers using the 80C88 processor and its variants generally the slowest. Within each of these broad categories is considerable diversity. Some of it can be accounted for by CPU clock speed, wait states, or disk drive speed, but some differences in performance remain even among machines with almost identical designs.

The lesson to be learned from this analysis is that, in microcomputer performance, the proof is in the using. Those who are concerned about performance should measure the speed of the computers in performing tasks for which they are likely to use the machines. These performance tests are critical even for machines with supposedly equivalent architectures and components. A series of benchmarking programs can capture important differences in performance that design specifications do not reveal.

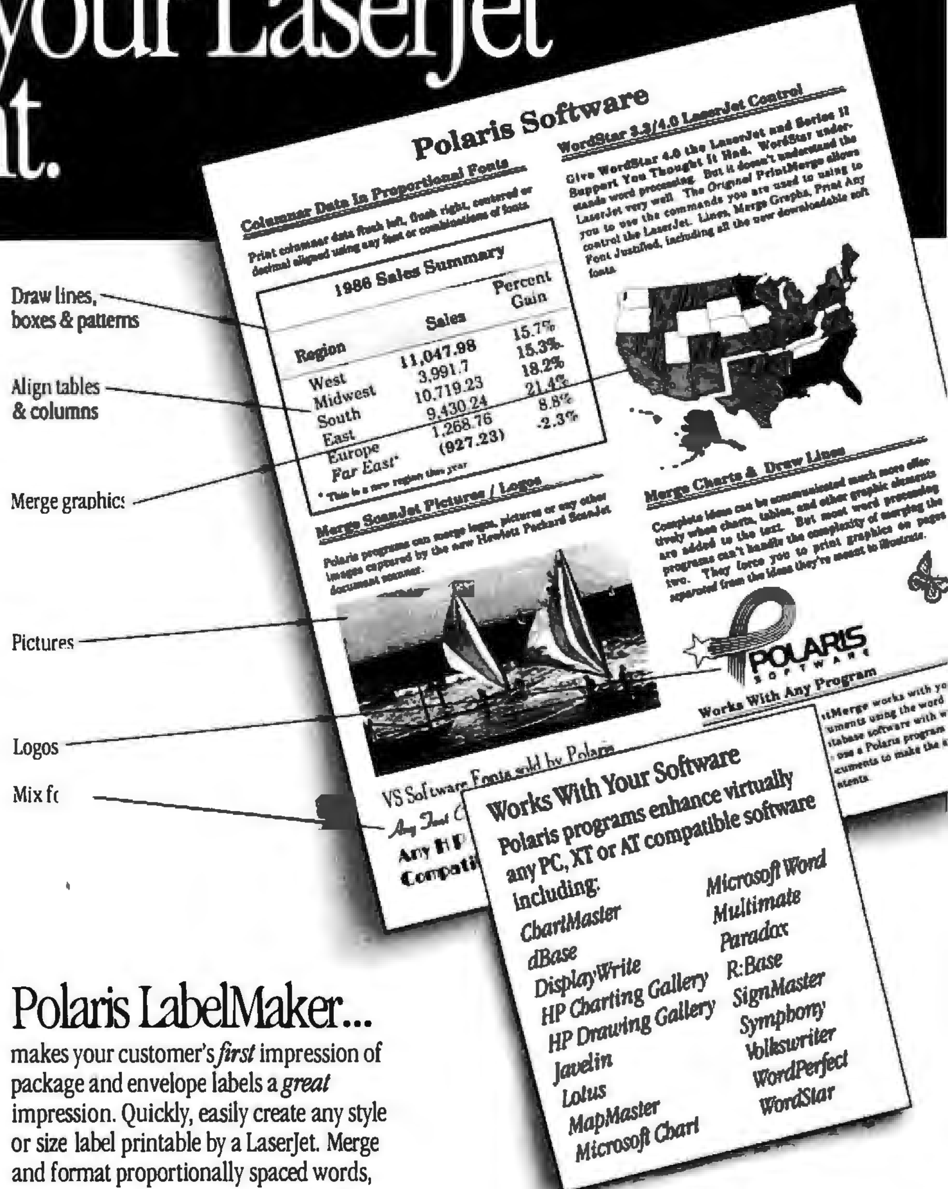
Finally, a few words of caution about interpreting this performance ranking. Although the multidimensional-scaling-analysis results in this article are valid for the machines analyzed, they could vary significantly if other computers, particularly computers differing substantially in performance from those examined here, were introduced into the analysis. Adding a new machine to the ranking would require the calculation of new dissimilarities and a new analysis of the entire set of data. It is not possible to infer from this particular ranking where a machine not included in this analysis would fall. ■

REFERENCES

1. Kruskal, Joseph B., and Myron Wish. *Multidimensional Scaling.* Sage University Paper Series on Quantitative Applications in the Social Sciences, no. 07-011. Beverly Hills, CA: Sage Publications, 1978.
2. Schiffman, Susan S., M. Lance Reynolds, and Forrest W. Young. *Introduction to Multidimensional Scaling.* New York: Academic Press, 1981.
3. Kruskal, Joseph B. "Multidimensional Scaling by Optimizing Goodness of Fit to a Nonmetric Hypothesis." *Psychometrika,* March 1964, pages 1-27.
4. Kruskal, Joseph B. "Nonmetric Multidimensional Scaling: A Numerical Method." *Psychometrika,* June 1964, pages 115-129.

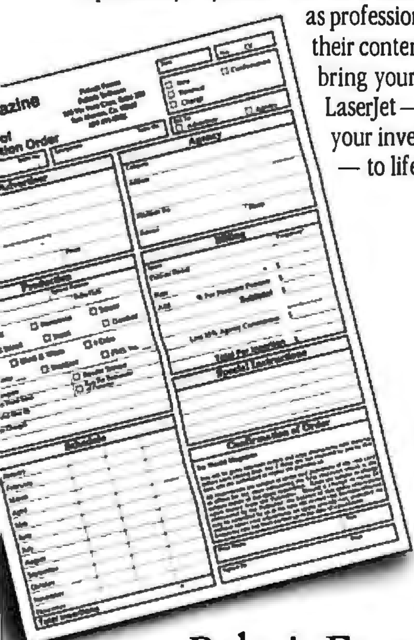
Don't waste your LaserJet investment.

You're not getting full benefit from your LaserJet or Series II investment. Because your wordprocessing, spreadsheet and database software only tap a fraction of its full potential. But POLARIS programs provide the enhancements you need to create documents that look as professional as their contents. Here's how.



Polaris PrintMerge...

is the general purpose program for memos and letters, reports and documents. Do true proportional spacing and alignment of text, tables, columns with *any combination* of old/new soft and cartridge fonts. Merge text with screen or output graphics from *any* program — draw lines, boxes, patterns — *anywhere.* It's the quick easy way to make documents look as professional as their contents. To bring your LaserJet — and your investment — to life.



Polaris LabelMaker...

makes your customer's *first* impression of package and envelope labels a *great* impression. Quickly, easily create any style or size label printable by a LaserJet. Merge and format proportionally spaced words, graphics, lines, boxes, bar codes for the master.

LabelMaker automatically positions and prints multiple copies. Even include serial numbering. Everything you need — without the aggravation.

Polaris Forms...

is a special program for quickly, easily creating professional-quality forms. Mix and position any combination of old/new cartridge and soft fonts anywhere. Justify copy, or set flush left, right, or centered in any width. Draw lines, boxes, patterns. Automatically repeat text, lines, graphics. Merge graphics and text from other programs. Make blank or completed, single or multiple page forms. Even have automatic serialization, date or time. Unique capabilities handle current and future needs without the time and expense of typesetters, graphic designers.

Order Direct Today...

to make sure you *are* getting the full benefit from your LaserJet or Series II investment.

800/338-5943

(In CA 800/231-3531)

for IBM PC, XT, AT & 100% compatibles.
*Add $6 shipping & handling. 30-day money back guarantee.

Polaris Ram-resident PrintMerge **$149***
Polaris PrintMerge for WordStar 3.3 & 4.0 **$124***
Polaris Forms **$149***
Polaris LabelMaker **$124***
Polaris Lines • Graphs • Symbols **$75***
Polaris Font Control for Word & WordStar 2000 (automatic PRD & PDF table generation) **$99***
Polaris ScanControl **$99***

POLARIS SOFTWARE
613 West Valley Parkway, Suite #323
Escondido, California 92025
619/743-7800

Circle 181 on Reader Service Card (Dealers: 182)

Call today for the name and number of your nearest CMS dealer.

CMS®

CMS Enhancements, Inc.
1372 Valencia Avenue, Tustin, CA 92680 (714) 259-9555
Telex (023) 371-8711 FAX (714) 549-4004

Circle 173 on Reader Service Card (Dealers: 174)

Add windowing to your BASICA programs using this assembly language subroutine

Windows for BASIC

John W. Ross

Most commercial programs that do any sort of on-screen interaction with the user have ads boasting that they "do windows." These applications programs are usually written in C or some other compiled language. In this article, I will show how you can add a windowing subroutine to programs written for the IBM PC's Microsoft BASICA interpreter and other fully compatible Microsoft BASICs.

The windowing must be done in assembly language to get the required speed from the interpreter. Depending on your viewpoint, this can be an advantage or a disadvantage. As a bonus, careful study of the listings will reveal some techniques that you can apply to other interpreter/assembler interfaces.

Video Basics

To implement this project, I will work directly with the PC's screen memory. In any project of this sort, you need a good understanding of the underlying hardware principles, so I will first present some background on the PC's video display system.

The IBM PC uses memory-mapped video; this means that character positions on the screen correspond to locations in the PC's memory. Memory locations A0000 to BFFFF (hexadecimal) are set aside for video display use. The exact correspondence depends on the type of graphics card or video adapter you use. The original IBM cards are the Monochrome Adapter (MA) and the CGA. Since these two cards are the most common, I will deal with them in this article.

The MA occupies 4K bytes of display memory starting at address B0000, while the CGA occupies 16K bytes from address B8000. While the MA can display only text and predefined graphics characters, the CGA, with its greater complement of memory, can operate in three graphics modes, two 40-column text modes, and two 80-column text modes.

I designed the windowing subroutine for the 80-column text mode, so it is compatible with both the MA and the CGA. DOS identifies the appropriate video modes as modes 2 (CGA gray), 3 (CGA color), and 7 (MA).

In 80-column text mode, the CGA has four times the MA's memory. This memory is divided into four "pages," numbered 0 to 3, only one of which appears on the video screen at one time. Usually, you work only with page 0, but you can make any of the four pages active, and both DOS and BASIC provide facilities for switching video pages. The windowing subroutine will detect which page is currently active and place its windows on the correct page.

Each character position on the screen occupies 2 bytes in video memory. The even-numbered bytes contain the code of the character displayed on the screen, while the following odd-numbered bytes contain the character's attributes. The 8 bits of the attribute byte have the meanings shown in table 1.

You can obtain the 256 possible combinations of these attributes by adding the values of the desired characteristics. For instance, an attribute value of 4 corresponds to red characters on a black background; adding 8, for an attribute value of 12, gives bright-red characters on a black back-

continued

John W. Ross is a senior scientific computing consultant at the University of Toronto and president of Jayar Systems, maker of a graphics package for the IBM PC. He can be contacted at Jayar Systems, 253 College St., Suite 263, Toronto, Ontario, Canada M5T 1R5.

ground, while a value of 140 (i.e., plus 128) gives you blinking, bright-red characters on a black background.

In monochrome mode, the attribute byte has a more limited interpretation, although it uses the same general coding scheme. The blinking and intensity bits still have the same meaning, but only four color combinations are valid: an attribute value of 7 produces normal white on black, a value of 1 produces underlined white characters on black, a value of 112 produces reversed (black-on-white) characters, and an attribute value of 0 yields invisible characters. All other attribute values show the same as 7—normal white on black.

Implementing Windows

I will start with a definition or description of a window from the user's point of view. Windows are rectangular regions that appear almost instantly on the screen to display information or allow interaction with the user. Crucial to their utility is the fact that, when the window disappears, the previous contents of that screen area are restored.

Given the memory-mapped nature of the IBM PC's screen, it's easy to see how a program could place a window on the screen. The program simply places blank bytes in the memory locations corresponding to the window's desired location on the screen and sets the attribute bytes to the desired color. To spiff things up a bit, the program could place box-drawing characters in the perimeter-window bytes to help delineate the window from the surrounding text.

This still leaves the problem of restoring the previous screen contents. To solve this problem, the program must supply a buffer into which it can copy the bytes from the screen where the window will be opened, before opening the window (remember that the buffer will have to provide storage for 2 bytes for each screen location). Then, when the window is closed, the original screen bytes are copied back onto the screen, leaving it just as it was before the window appeared.

If the buffer is big enough, you can have multiple windows open at once, and these windows can overlap one another to any degree. (In the case of overlapping windows, it is necessary to close them in the reverse order from which they were opened.)

Windows in BASIC

Since BASIC supplies PEEK and POKE commands that let you access any memory location, including screen memory, you might wonder why you couldn't implement windows in BASIC. In fact, you could do it with quite a compact bit of code. Unfortunately, when you open a window this way, it is excruciatingly slow. The window doesn't "pop," "flash," or "open" as windows are supposed to—it is painted on the screen byte by byte. This slowness is especially noticeable for big windows. Windows implemented in BASIC lose all their impact; if you try it, you will be unimpressed.

Even that old cure-all for BASIC's sluggishness, resorting to the compiler, doesn't help much. It's faster, but you can still see the windows being painted on the screen rather than just popping up—again, the effect is worse for large windows. Even if this were the answer to the speed problem, it wouldn't be an ideal solution. Not everyone has access to the Microsoft compiler, and resorting to the compiler means that you lose the advantages of using an interpreter in the first place.

This dilemma has only one solution—you have to resort to assembly language to implement the speed-critical subroutine. Although interfacing assembly language routines to interpreted BASIC programs can drive you to the brink of madness (if not clear over), if it works, the results are dramatic. In this case, the windows appear and disappear instantly, with no painting effect.

Assembly Language Windows

I've provided two ways for you to access the assembly language subroutine from your own programs. If you have the Microsoft Macro Assembler, type in the program in listing 1, WINDOW.ASM, which you can then assemble and link to produce WINDOW.EXE. This is the route to go if you want to experiment a bit. Alternatively, I've provided a generator program, WINDOW.BAS; this BASIC program generates the file WINDOW.EXE. [Editor's note: *WINDOW.BAS is available on BIX, on disk, and in print. See the insert card following page 208 for details. Listings are also available on BYTEnet. See page 4.*]

When you call the windowing subroutine from BASIC (I will discuss the format later), you supply the following arguments: a parameter that indicates to the subroutine whether you want to open or close a window, a starting row and column number for the window, its size in rows and columns, an attribute byte for the window, and the location of a buffer array that the subroutine can use to store the portion of the screen that the window will overlay. (You supply the buffer in BASIC to minimize the subroutine size and devote no more memory than necessary to the buffer.) The subroutine then places a window on the screen, outlined by a double line, in the position requested.

I will describe the subroutine's operation in more detail, but first I have to discuss a problem that anyone developing assembly language applications for BASIC has to overcome—where to put the assembly language subroutine. It has to be someplace where BASIC can find it so control can be transferred when necessary, but the location has to be secure enough so that the subroutine won't be damaged by BASIC, DOS, or any other gremlins that might be wandering around your computer's memory.

The neatest solution is to make the subroutine a resident program, which effectively makes it an extension of DOS. The subroutine is snug and safe there; you can forget about it and let DOS look after it. Once you install it, it will remain in place until you reset the system. The other advantage of having the subroutine permanently installed in DOS is that you don't have to explicitly include it in your BASIC programs.

When you call the window subroutine from BASIC, control is transferred to the section of code labeled `start` in listing 1. The program first saves BASIC's registers, then extracts the argument addresses passed by BASIC. The program next checks the current video mode. If it is not 80-column text mode, the program returns to BASIC without doing anything. Assuming the mode is acceptable, the program uses the input window location to compute the offset of the window from the start of display memory.

Depending on the function code passed to the routine, it either stores the contents of the screen area that will be covered by the window in the buffer supplied by BASIC, or it reverses the procedure and restores the screen contents from the buffer. If the program is closing a window (i.e., restoring the screen contents), it returns to BASIC at this point; otherwise, it places a window on the screen at the specified location. The internal subroutine `put_line` does this; it places one line of the window at a time on the screen.

The section of the program starting at `loader` executes when you install the program. It loads the subroutine in memory, then sets a pair of pointers to the subroutine in the interapplication communications area (ICA). This 16-byte area in DOS at location 0000:04F0h is reserved

continued

Listing 1: *The assembly language code that, when compiled and linked, produces the file WINDOW.EXE. When you run WINDOW.EXE, it installs the windowing subroutine in DOS.*

```
; Windows for interpreted            Copyright (C) John W.
  BASIC;                              Ross 1986;
 Calling format:
;   CALL WINDOW (BUFR%(I), IFN%, ROW1%, COL1%, NROW%,
     NCOL%, ATTR%)
;
;          BUFR%                     buffer array supplied
                                      by BASIC to hold
                                       screen contents
;          IFN%      function to      1 - place window on
                      perform:         screen
;                                     2 - restore screen
                                       contents
;          ROW1%                     row number of upper
                                      left corner of window
                                      (0-24)
;          COL1%                     column number of upper
                                      left corner of window
                                      (0-79)
;          NROW%                     height of window in
                                      rows
;          NCOL%                     width of window in
                                      columns
;          ATTR%                     screen attribute of
                                      window area
;-------------------------------------------------------
zseg       segment   at 0
           org       04f0h
comm_ofst  dw        ?               ;vectors to routine in
comm_seg   dw        ?               ; inter-apps comm area
zseg       ends
;-------------------------------------------------------
grp        group     dcseg,zloader   ;an artifice to force
                                       code
dcseg      segment   'code'          ; to load first
dcseg      ends
;-------------------------------------------------------
code       segment   para public     ;define code seg
                      'code'
;-------------------------------------------------------
windo      proc      far             ; this is the windows
                                       subroutine
           assume    cs:code,ds:code,ss:st-seg

;save BASIC's registers
start:     push      bp
           mov       bp,sp
           push      ds
           push      es
           push      ss
          jmp begin; stash our data in here
cpyrt      db        'Copyright (C) John W. Ross 1986'
                      ;brought to you by...
col1       dw        ?               ;upper left column
row1       dw        ?               ;upper left row
col_cnt    dw        ?               ;width in columns
row_cnt    dw        ?               ;height in rows
dsp_mod    dw        ?               ;current CRT display
                                       mode
page_no    dw        ?               ;current display page
attr       db        ?               ;window display
                                       attribute
ifn        dw        ?               ;function: 1 - open
                                       window
                                     ;          2 - close it
```

continued

Buy a modem from us and we'll throw in 300,000,000 new business offices worldwide.

When you leave your office, you shouldn't lose your data communications ability. Now, you can transmit and receive data from virtually anywhere in the world. Our WorldPort Series of modems allows you to connect in millions more locations than conventional modems. This means your modem can work where you do—hotel rooms, phone booths, internationally, even with digital PBXs.

The WorldPort 2400™ and the WorldPort 1200™ modems represent the cutting edge of modem technology. Their features go far beyond internal units, giving you both Bell and CCITT standards, direct connect and acoustic interface (300 and 1200 bps), battery power, and shirt pocket size. In fact, WorldPort may be the ultimate modem for both portable and desktop applications. And if you're impressed with the size, check our price.

For even greater versatility and value, the WorldPort 1200 can be easily upgraded to 2400 bps. And the WorldPort 2400 comes with Carbon Copy PLUS™ communications software, a real "double feature." Find out how doing business with Touchbase Systems can help you locate a few extra places to do business.

For more information about WorldPort modems and the name of your nearest dealer, call us today at 800-541-0345. In New York call (516) 261-0423.

Touchbase Systems, Inc.
160 Laurel Avenue
Northport, NY 11768
(516) 261-0423
TELEX: 6502848020
FAX: (516) 754-3491

Circle 226 on Reader Service Card

```
video     dw      0b800h,0b900h,0ba00h,0bb00h ;CGA
                   pages 0-3
mono      dw      0b000h          ;mono adapter
begin:
;get arguments
          mov     bx,[bp]+6
          mov     ax,[bx]
          mov     cs:attr,al
          mov     bx,[bp]+8
          mov     ax,[bx]
          mov     cs:col_cnt,ax
          mov     bx,[bp]+10
          mov     ax,[bx]
          mov     cs:row_cnt,ax
          mov     bx,[bp]+12
          mov     ax,[bx]
          mov     cs:col1,ax
          mov     bx,[bp]+14
          mov     ax,[bx]
          mov     cs:row1,ax
          mov     bx,[bp]+16
          mov     ax,[bx]
          mov     cs:ifn,ax
;save BASIC's data segment
          push    ds
;now set ds register to the code segment
          mov     ax,code
          mov     ds,ax;check current video state
          mov     ah,15
          int     10h             ;use the DOS interrupt
          cmp     al,2            ;black-and-white 80-
                                   column?
          je      crtok
          cmp     al,3            ;color 80-column?
          je      crtok
          cmp     al,7            ;is it mono?
          je      crtok
          pop     ds              ;none of the above
          jmp     exit            ; - fix stack and bail
                                   out
crtok:    mov     ah,0            ;get display mode from
                                   al
          mov     dsp_mod,ax      ; into a word
          mov     bl,bh           ;video page number
          mov     bh,0
          mov     page_no,bx
          shl     bx,1            ;index to video page
          cmp     dsp_mod,7       ;test display mode
                                   again
          jne     lab1            ;in case it's mono
          mov     bx,8            ;adjust video page
                                   number
lab1:     mov     dx,video[bx]    ;upper left of screen
          push    dx              ;save it for now
          mov     dh,byte ptr     ;define window
                   row1
          mov     dl,byte ptr
                   col1
          mov     al,80           ;pntr=80
          mul     dh              ;pntr=80*row1
          mov     dh,0
          add     ax,dx           ;pntr=80*row1+col1
          shl     ax,1            ;pntr=2*(80*
                                   row1+col1)
                                  ;point to window
                                   offset
          mov     bx,ax           ;now it's in bx
          pop     es              ;es now points to the
                                   screen
```

continued


```
;----------------------------------------------------.
;save or restore the screen image in the buffer
; provided by BASIC, depending on the value of IFN%
        mov     dl,byte ptr col_cnt
        mov     dh,byte ptr row_cnt
        mov     ax,ifn
        pop     ds              ;point to BASIC's data
                                 seg
        mov     di,[bp]+18      ;point to buffer in
                                 BASIC
        push    bx              ;save the screen
                                 offset;
        mov     cl,dh
        mov     ch,0
sl1:    push    cx;             do for col_cnt columns
        push    bx
        mov     cl,dl
        mov     ch,0
sl2:    push    cx
        cmp     ax,1
        je      down
        mov     cx,[di]         ;restore the screen
        mov     es:[bx],cx
        jmp     next
down:   mov     cx,es:[bx]
;save the screen image-byte & attribute
        mov     [di],cx
next:   inc     bx
        inc     bx
        inc     di
        inc     di
        pop     cx
        loop    sl2
        pop     bx
        add     bx,160
        pop     cx
        loop    sl1
        push    cs
        pop     ds              ;point to data (code)
                                 seg
        pop     bx              ;screen offset, next
                                 part
        cmp     ax,1
        jne     exit            ;in case IFN%=2
; set up registers to draw the top of the window
        mov     dl,201          ;left end-of-line
                                 character
        mov     dh,187          ;right end-of-line
                                 character
        mov     al,205          ;middle character of
                                 line
        mov     ah,attr         ;window attribute
        mov     cx,col_cnt      ;length of line
        call    put_line        ;do it
        add     bx,160          ;increment bx
;now for the body of the window
        mov     dl,186          ;left-end character
        mov     dh,186          ;right-end character
        mov     al,32           ;middle character
        mov     ah,attr
        mov     cx,row_cnt      ;do for NROW%s less 2
        dec     cx
        dec     cx
rowl:   push    cx
        mov     cx,col_cnt
        call    put_line
        add     bx,160          ;next line in window
        pop     cx
        loop    rowl
;finish it off
        mov     dl,200          ;left end
```

FOLD
HERE

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS MAIL PERMIT NO. 10 PETERBOROUGH, NH

POSTAGE WILL BE PAID BY ADDRESSEE

BYTE Listings
One Phoenix Mill Lane
Peterborough, NH 03458-9990

FOLD
HERE

```
        mov     dh,188          ;right end
        mov     al,205          ;middle
        mov     ah,attr         ;attribute
        mov     cx,col_cnt      ;length
        call    put_line
exit:   pop     ss              ;restore BASIC's
                                 registers
        pop     es
        pop     ds
        pop     bp
        ret     14
windo   endp                    ;end of main proc
;--------------------------------------------------
;put_line - puts a line on the screen
;assume upper left of screen area pointed to by es
;       bx points to offset of window in the screen
;       dl,dh end-of-line characters
;       al, ah midline character and attribute
;       cx contains line length put_line proc near push
         bx
;first, save bx
        mov     es:[bx],dl
        inc     bx
        mov     es:[bx],ah
        inc     bx
        dec     cx
        dec     cx
dl1:    mov     es:[bx],al
        inc     bx
        mov     es:[bx],ah
        inc     bx
        loop    dl1
        mov     es:[bx],dh
        inc     bx
        mov     es:[bx],ah
        pop     bx              ;recover bx ret
put_line        endp
;--------------------------------------------------
code    ends                    ;end of code segment
;--------------------------------------------------
st_seg  segment stack           'stack' ;define stack
                                 segment
        db      20 dup          ('stack ')
st_seg  ends
;--------------------------------------------------
zloader segment byte            'zload'
        assume  cs:zloader,ss:st_seg
endres  equ     $
;--------------------------------------------------
loader  proc    far             ;loader entry point
        push    es
        xor     ax,ax
        push    ax
        mov     byte ptr        ;set up to
                 es:1,27h
        xor     ax,ax           ;terminate and stay
                                 resident
        mov     es,ax
        assume  es:zseg         ;point to low memory
        mov     comm_ofst,offset
                 windo          ;subroutine entry
        mov     comm_seg,seg windo   ;within this
                                      segment
        mov     dx,offset grp:endres+100h   ;don't
                                      save loader
        ret
loader  endp
;--------------------------------------------------
zloader ends
;--------------------------------------------------
        end     loader
```

for programs that want to pass information to one another. Your BASIC programs will know to look here for the location of the window subroutine. Its address is stored as two 16-bit words: a segment address and an offset within the segment. After setting the pointers, the loader executes a terminate-but-stay-resident interrupt, which effectively incorporates the window subroutine, but not the loader, into DOS.

Using Windows

First, you have to generate the WINDOW.EXE file by one of the two methods mentioned earlier. When you begin a session in which you will use the window subprogram, you must first execute WINDOW to install the subroutine. From the DOS prompt, type `WINDOW`.

The `A>` prompt should return momentarily. At this point, the subroutine is installed and will remain so until you reboot the computer. All a BASIC program has to do is look at the ICA to see where the subroutine is located and call it.

A problem might arise if you attempt to keep the window subroutine in memory while running non-BASIC programs. Some of these other programs might use the ICA for their own purposes; this would destroy the address left by the window loader, and when your BASIC program came to call it again later, the results would be unpredictable and almost certainly undesirable.

If you have any doubts about whether your previously loaded window subroutine has been "lost" by another program using the ICA, repeat the installation process; it will cost you only another 1K byte or so of memory. To reclaim the memory, you must reset the computer.

Now look at what your BASIC applications program has to do to call the subroutine. I'll present some specific examples later, but for now I'll look at the more general steps. First, your program has to find out where the subroutine is located. The following statements should be executed before the first call:

```
10 DEF SEG=0
20 WINDO=PEEK(&H4F0)+
   (256*PEEK(&H4F1))
30 SUBSEG=PEEK(&H4F2)+
   (256*PEEK(&H4F3))
40 DEF SEG=SUBSEG
```

`WINDO` is the address of the memory-resident subroutine, and `SUBSEG` is its memory segment. To call the program, you set the segment using `DEF SEG` and issue a call to the subroutine address.

The format of the subroutine call is `CALL WINDO (BUFR%(I),IFN%,ROW1%,COL1%,NROW%,NCOL%,ATTR%)`. Note that the arguments are all integers, and they must all be initialized before the call.

`ROW1%` and `COL1%` values specify the location of the window on your screen, in terms of the position of its upper left corner. Row numbers run from 0 to 24, and column numbers from 0 to 79. To place a window in the upper left corner of your screen, set `ROW1%` and `COL1%` to 0.

`NROW%` and `NCOL%` parameters determine the size of the window in rows and columns. Valid ranges are 2 to 25 for `ROW1%` and 2 to 80 for `COL1%`.

`ATTR%` determines the window's background color and the color of its double-line outline.

`BUFR%(I)` is a buffer for the subroutine to temporarily store the screen contents. It must have a dimension of at least *n*, where *n* is the number of rows times the number of columns of the biggest window you will use. (Since an integer holds 2 bytes, one array element will hold one screen location: character plus attribute.) If you plan to have more than one window active at a time, the dimension of `BUFR%` must be large enough to handle the contents of all the open windows.

`I` is an index to the array element in `BUFR%` where you want to start storing screen contents. You must supply it. For instance, say you want to create 5-by-10 and 12-by-30 windows. `BUFR%` would be dimensioned as a 410-element array. You would call the window subroutine the first time with `I` pointing to `BUFR%(1)` and the second time with `I` pointing to `BUFR%(51)`.

`IFN%` is a function switch. If `IFN%` is 1, screen contents are saved starting at `BUFR%(I)`, and a window is opened on the screen. If `IFN%` is 2 (or any value other than 1), `NROW%` × `NCOL%` elements from the buffer are put back on the screen, effectively closing the window. The usual sequence of operation is to call `WINDO` with `IFN%` equal to 1 to fill `BUFR%`, then with `IFN%` equal to 2 to replace it. For special purposes, however, you could fill `BUFR%` with whatever you wanted, then execute a call to `WINDO` with `IFN%` equal to 2 to dump it quickly to a specific screen location.

I wish to stress several points. Remember that all parameters must be integers and must be initialized. The subroutine does not do any error checking for validity of the arguments. Your program

continued

Table 1: *Meanings of the bits contained in the attribute byte.*

Bit	Value	Meaning
0	1	Blue component, foreground
1	2	Green component, foreground
2	4	Red component, foreground
3	8	Intensity component, foreground
4	16	Blue component, background
5	32	Green component, background
6	64	Red component, background
7	128	Blinking component, foreground

Listing 2: *A BASIC program that calls the windowing subroutine to conduct a dialogue with the user.*

```
100  ' Example program 1 - Window Dialogue
110  'Get location of windows subroutine
120  DEF SEG=0
130  WINDO=PEEK(&H4F0)+(256*PEEK(&H4F1))
140  SUBSEG=PEEK(&H4F2)+(256*PEEK(&H4F3))
150  DEF SEG=SUBSEG
160  'Define a buffer to hold a 7-by-40 array
170  DIM BUFR%(280)
180  'Set up the subroutine arguments
190  ROW1%=2     ' window will appear in 3rd row,
200  COL1%=3     ' 4th column
210  NROW%=7     ' it will be seven rows deep and
220  NCOL%=40    ' 40 columns wide
230  ATTR%=80+3+8+128 ' magenta bknd, cyan fgnd, intense, blinking
240  'Open window
250  IFN%=1
260  CALL WINDO (BUFR%(0),IFN%,ROW1%,COL1%,NROW%,NCOL%,ATTR%)
270  'Extract foreground and background colors from ATTR%
```

continued

AT compatibility and protected mode software for your XT. . .

All This and OS/2!

The MotherCard 5.0

With the MotherCard 5.0 from SOTA Technology your XT or compatible can run Microsoft's OS/2 *and* all the next generation software.

The MotherCard 5.0 is a full-blown 80286 computer on a card. When installed in any 8088 based PC or XT, it becomes fully AT compatible. it also provides all the related hardware that is required to run OS/2 as well as the speed and memory needed to meet future computing demands.

MotherCard 5.0 Standard Features:

- 8, 10 or 12.5 MHz 80286 on board
- 80287 socket
- AT compatible Real-time clock
- Reconfigurable BIOS in static RAM and EPROM
- 640KB for DOS & 320 KB for EMS - expandable to 16 MB
- Software includes: RAM disk, Disk Cache, and Print Spooler

Compatibility: The card contains a reconfigurable BIOS stored in battery-backed RAM. This allows for an easy upgrade for a new BIOS, thus ensuring full compatibility for tomorrow's applications, including, of course, OS/2.

Speed: 8, 10, or 12.5 MHz - can run up to 12 times faster than your PC or XT and up to 2 times faster than an AT!

Memory: The board comes standard with 1 MB of memory (640 KB of DOS and 320 KB of EMS memory) Expandable up to 16 MB with an additional DaughterCard.

The MotherCard 5.0 works with all EGA, CGA, LAN, PC networks and tape backup systems without a hitch.

SOTA
STATE OF THE ART TECHNOLOGY™

SOTA TECHNOLOGY, INC.
657 N. Pastoria Ave., Sunnyvale, CA 94086

If you have an XT and feel the era of protected mode and OS/2 is going to pass you by, don't worry! The MotherCard 5.0 from SOTA Technology will breathe new life into your old machine *and* let you avoid all the hidden costs of buying a new AT system (learning a new machine, configuring your new system etc.). Remember the MotherCard 5.0 - it's an instant AT for a lot less!

SOTA Technology:
Turning the simple into the superlative.

The MotherCard 5.0 can run:

- Protected mode software: VDISK, Oracle's Professional ORACLE, Micro Focus COBOL/2
- Novell Advanced Netware
- 3Com 3+ Network
- EGA & EGA Monitors of all kinds
- LOTUS 1-2-3 and SYMPHONY
- Ashton-Tate dBASE III Plus, Multimate
- All word processing programs
- MicroSoft Windows

With reconfigurable BIOS, MotherCard 5.0 will support protected mode OS/2

The above screen was created with MicroSoft OS/2 software developer's kit in an IBM PC with the MotherCard installed.
IBM, OS/2, PC, XT, AT are trademarks of IBM Inc. All other products mentioned are trademarks of their respective companies.

See us at Comdex/Fall '87
Booth #C521

```
280  FG=ATTR% AND 15
290  BG=INT(ATTR%/16) AND 7
300  COLOR FG,BG
310  'Conduct dialogue in window
320  LOCATE ROW1%+3,COL1%+3
330  PRINT "What is your name "
340  LOCATE ROW1%+5,COL1%+3
350  INPUT NAM$
360  'Close window
370  IFN%=2
380  CALL WINDO (BUFR%(0),IFN%,ROW1%,COL1%,NROW%,NCOL%,ATTR%)
390  COLOR 7,0
```

Listing 3: *A BASIC program that places multiple, overlapping windows on the screen, then removes them.*

```
100  ' Example program 2 - Multiple Windows
110  '
120  ' get location of windows subroutine
130  DEF SEG=0
140  WINDO=PEEK(&H4F0)+(256*PEEK(&H4F1))
150  SUBSEG=PEEK(&H4F2)+(256*PEEK(&H4F3))
160  DEF SEG=SUBSEG
170  ' window data
180  DATA 3, 3, 20, 24, 44, 5, 5, 16, 30, 27, 2, 50, 15, 24, 109, 15, 40, 5,
     25, 79, 13, 38, 10, 20, 219, 0, 0, 25, 80, 62
190  MAXBUF=3800: MAXWND=5
200  DIM BUFR%(MAXBUF), R(MAXWND), C(MAXWND), NR(MAXWND), NC(MAXWND),
     NBYT(MAXWND)
210  NWND=0: NBUF=0
220  ' Open windows
230  FOR IDX=0 TO MAXWND
240    A$=INKEY$: IF A$="" THEN 240
250    READ ROW1%, COL1%, NROW%, NCOL%, ATTR%
260    GOSUB 1000
270  NEXT
280  NWND=NWND-1
290 ' close windows
300  FOR IDX=MAXWND TO 0 STEP -1
310    A$=INKEY$: IF A$="" THEN 310
320    GOSUB 2000
330  NEXT
340  END
1000  'Subroutine to open a window
1010  '
1020  ' check for invalid arguments
1030  IF NBUF+NROW%*NCOL% > MAXBUF THEN 1130
1040  IF ROW1% < 0 OR COL1% < 0 THEN 1130
1050  IF ROW1%+NROW > 25 OR COL1%+NCOL > 80 THEN 1130
1060  ' store window data
1070  R(NWND)=ROW1%: C(NWND)=COL1%: NR(NWND)=NROW%: NC(NWND)=NCOL%
1080  NBYT(NWND)=NBUF ' number of display positions occupied
1090  IFN%=1
1100  CALL WINDO (BUFR%(NBUF), IFN%, ROW1%, COL1%, NROW%, NCOL%, ATTR%)
1110  NBUF=NBUF+NROW%*NCOL%
1120  NWND=NWND+1
1130  RETURN
2000  'Subroutine to close a window
2010  '
2020  ' recover window data
2030  ROW1% = R(NWND): COL1%=C(NWND): NROW%=NR(NWND): NCOL%=NC(NWND)
2040  ' check for invalid arguments
2050  IF ROW1% < 0 OR COL1% < 0 THEN 2090
2060  IF ROW1%+NROW > 25 OR COL1%+NCOL > 80 THEN 2090
2070  IFN%=2
2080  CALL WINDO (BUFR%(NBYT(NWND)), IFN%, ROW1%, COL1%, NROW%, NCOL%, ATTR%)
2090  NWND=NWND-1
2100  RETURN
```

should do this. Also, make sure BUFR% is big enough to hold whatever you are going to put into it. If you try to stuff a big chunk of screen into a small buffer, you won't get a subscript out of bounds error; your program or BASIC will get clobbered, and it will be big-red-switch time.

Again, if you leave BASIC to run some programs that you suspect might alter the first two words of the ICA, it would be safe to install another copy of the subroutine.

In the next section, I'll present two BASIC programs that illustrate some techniques for using the windowing subroutine.

Sample Programs

Listing 2 demonstrates the use of windows to conduct a dialogue with the user. The idea is to place a window on the screen, then use LOCATE to position the cursor within the window for input and output. You could also capture input by using INKEY$. Note that WINDO numbers rows and columns from 0 to 24 and from 0 to 79, respectively; LOCATE numbers them from 1 to 25 and 1 to 80.

Line 230 shows how you can customize the window colors by adding together the desired characteristics. Lines 280 and 290 demonstrate a technique for extracting the foreground and background colors from the attribute value. This lets you set BASIC's text-output colors to match the window colors with the COLOR statement. Try this program when the screen is filled with text. For example, try it when the screen is filled with the program listing.

Listing 3 illustrates the use of multiple, concurrent windows. When you run this program, every time you hit a key, one of six windows appears sequentially on the screen. As you keep pressing keys, the windows are closed. The window characteristics and positions are read from the data statement in line 180.

This demonstration program shows how a large buffer provides storage for multiple windows. Note the check in line 1030 to make sure that the assembly language program doesn't overflow the buffer.

Not Just Window Dressing

In this article, I've presented a subroutine that lets interpreted BASIC do windowing, a technique that will give your programs a state-of-the-art look. Apart from the usefulness of the program itself, it demonstrates how you can extend interpreted BASIC's capabilities with assembly language subroutines and illustrates a technique for interfacing such subroutines to BASIC. ■

XyWrite III Plus: For People Who Write

People who write a lot have always depended on XyWrite to get the job done. Datapro Research Corporation said XyWrite III was "...the fastest word processor...its power and flexibility are unmatched." Now XyWrite III Plus introduces authoring tools that will make you even more productive than before.

XyWrite III Plus has the flexibility to handle all of your writing needs, from simple letters and memos to special applications such as audio-visual scripts with multiple independent columns, or complex research papers which require one-key switching between notes, outlines and drafts – up to nine different files in nine "windows."

The on-line *Spelling Checker* sports a 100,000-word dictionary to help prevent embarrassing, and costly, typos. Check your spelling as you work. Or, if you'd prefer not to break your train of thought, wait and check the entire document when you've finished writing it. You can even check multiple files with a single command. A unique shorthand feature saves valuable time by expanding predefined abbreviations; "asap" becomes "as soon as possible" instantaneously! And the renowned 220,000-word *Microlytics' WordFinder® Thesaurus* enhances creativity, putting alternative words at your fingertips.

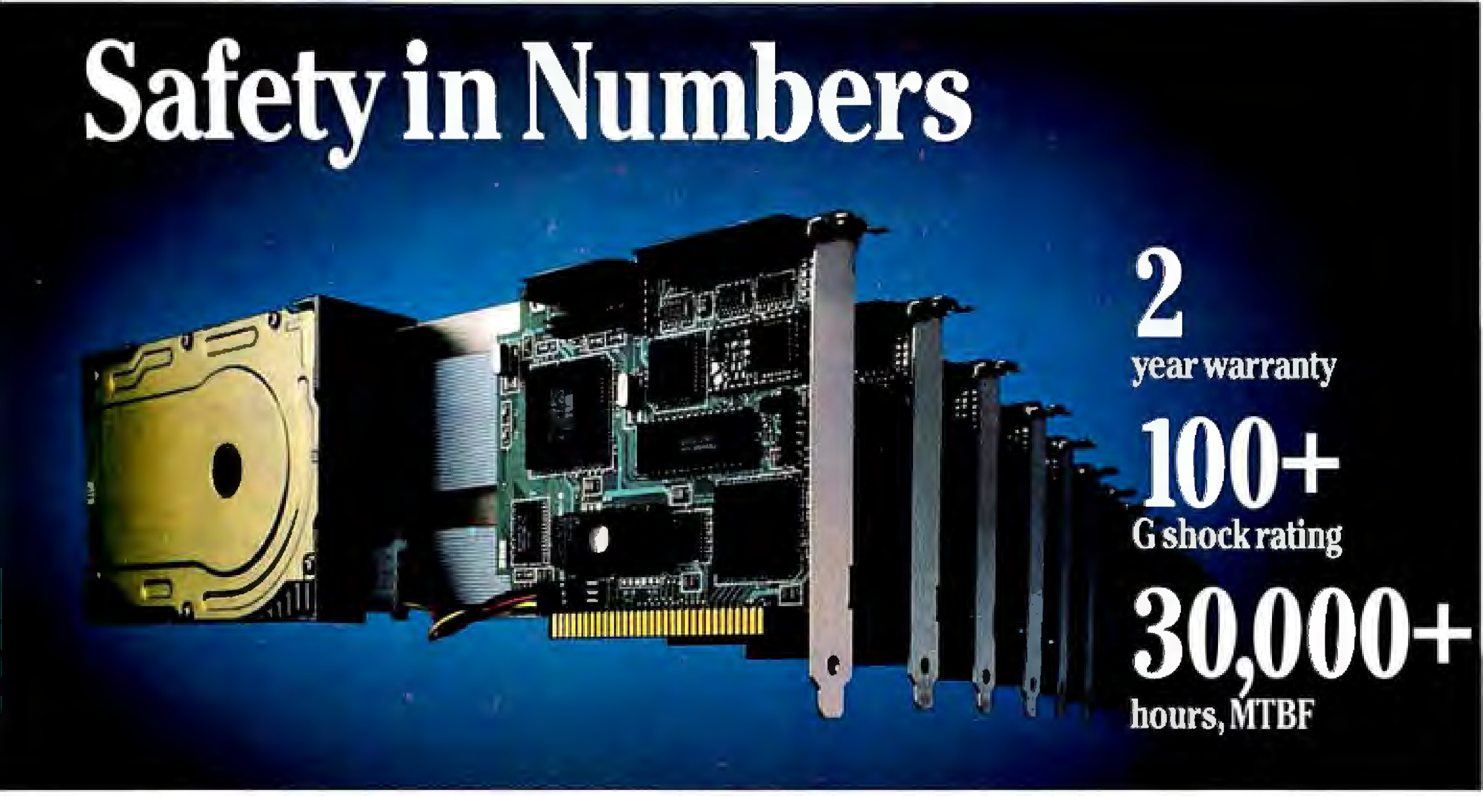
Use the *Redlining* feature to edit another writer's document. Mark suggested changes to the text but leave the original material intact. Combine *Redlining* with *Hidden Notes* and you have an ideal way to share thoughts with the original author.

XyWrite III Plus provides unmatched laser printer support (including PostScript) for desktop publishing. Whether you're using cartridge or downloadable fonts, or mixing different point sizes on a line, XyWrite III Plus will automatically hyphenate and justify...as you type. A host of vertical spacing commands give you maximum control over vertical movement as well. And XyWrite III Plus lets you *Include* ready-to-print files from other sources, eliminating the need to cut and paste.

The superior output capabilities of XyWrite III Plus, combined with an established reputation for speed and flexibility, clearly indicate that there is only one choice for word processing software. If you write, you need XyWrite III Plus. For more information and the name of a dealer near you, call or write today.

XYQUEST
3 Loomis St., Bedford, MA 01730
(617) 275-4439

Circle 249 on Reader Service Card



Reliability. The attribute that has made Silicon Valley Computer the world's leading manufacturer of ruggedized IBM™ compatible equipment. Reliability proven by the successful installation of thousands of disk and tape systems monthly. Superior engineering and meticulous care in manufacturing and testing have been the foundation of this reliability since 1982.

We offer a full selection of ruggedized hard-disk cards and high-quality commercial-grade disk-drive kits ranging in capacity from 21 to 512 megabytes. Also available are tape subsystems, 8- and 16-bit SCSI/ESDI/MFM/RLL disk controllers, Token Ring Ethernet controllers (Novell compatible), 3.5-inch floppies, 360 KB, 720 KB, 1200 KB, and 1440 KB floppies.

Tens of thousands of units are currently being used by the U.S. Navy, NASA, Pentagon R&D, major banking institutions, and numerous small businesses. All products meet or exceed stringent performance criteria. SVC established these standards to satisfy the needs of our growing list of sophisticated customers.

Compatibility.
Silicon Valley Computer drives have been interfaced with most IBM PC, XT, XT286, ATs and compatibles. Over 50 clones have been tested as 100% compatible with our drives using IBM PC-DOS 3.1/3.2/3.3.

We are also the first company in the industry to supply hard disk add-on units for the new IBM Personal System 2 Model 30, Model 60, and Model 80 machines.

Service.
Quality customer service is as important to us as the quality of our product. Ninety percent of all orders are shipped within 24 hours of receipt. Each order is accompanied by complete documentation, software, cables and mounting hardware. All products are available with our industry-leading two-year swap-out warranty. When you want the best product, service and reliability, at very competitive prices, call Silicon Valley Computer.

COMDEX/Fall '87
Booths 321 and 323

SILICON VALLEY
COMPUTER

140 Archer Street, San Jose, California 95112 • 408 288-8837
TELEX 821299 SILICON VLY • FAX 408-947-1353

Circle 202 on Reader Service Card

The Gold Card (hard-disk-card)

Silicon Valley Computer has the widest selection available.

RUGGEDIZED	
21R MB	80G shock • 65 ms w/auto head lifters • 30,000 hr MTBF (MFM)
32R	80G shock • 65 ms w/auto head lifters • 30,000 hr MTBF (RLL)
40R	100G shock • 29 ms w/auto head lifters • 30,000 hr MTBF (SCSI)
40AT	100G shock • 26 ms/hd lifters • 100% AT compatible • 30,000 hr MTBF
SILENT RUNNING	
21L MB	65 ms • Oxide media • 5,000,000 BPS • 10 W, 30,000 hr MTBF (MFM)
21F	65 ms • Plated media • 30 dB • 5,000,000 BPS • 25,000 hr MTBF (MFM)
32F	65 ms • Plated media • 30 dB • 7,500,000 BPS • 25,000 hr MTBF (RLL)
50F	62 ms • Plated media • 30 dB • 7,500,000 BPS • 25,000 hr MTBF (RLL)

Hard-Disk PC Kits (half height)

21L MB	65 ms • Oxide media • 5,000,000 BPS • 10 W • 28,000 hr MTBF (MFM)
32F	65 ms • 30 dB • Plated media • 7,500,000 BPS • 20,000 hr MTBF (RLL)
40R	29 ms • Plated media • 8,000,000 BPS • 10 W • 30,000 hr MTBF (SCSI)
40	26 ms • World's fastest 40 MB drive • 30,000 hr MTBF (MFM)
50F	62 ms • 30 dB • Plated media • 7,500,000 BPS • 20,000 hr MTBF (RLL)

Hard-Disk PC Kits (full height)

30 MB	40 ms • 30,000 hr MTBF • 5,000,000 BPS (MFM)
72	26 ms • 30,000 hr MTBF • 5,000,000 BPS (MFM)
120	28 ms • 30,000 hr MTBF • Novell compatible (MFM)
150	28 ms • 30,000 hr MTBF • 10,000,000 BPS (SCSI/ESDI)
234	28 ms • 30,000 hr MTBF • 7,500,000 BPS (SCSI)
317	28 ms • 30,000 hr MTBF • 10,000,000 BPS (SCSI/ESDI)
800	100 ms • Removable WORM • Optical/laser (SCSI)

IBM Compatible Devices

60 MB	Tape systems • Internal mounting, half-height DC 600 A cartridge
60 MB	Tape systems • External mounting power cabinet included
40 MB	Tape systems • Floppy interface DC2000 cartridge
	Token Ring compatible Ethernet boards • 10 MBPS
SCSI	Hard-disk tape controller • 8/16-bit switchable w/floppy
MFM	Hard-disk controller • 8/16-bit • 2 W • 2K buffer
RLL	Hard-disk controller • 8/16-bit • 2 W • 2K buffer
ESDI	Hard-disk/floppy/MFM controller • 8/16-bit • 10 MBPS

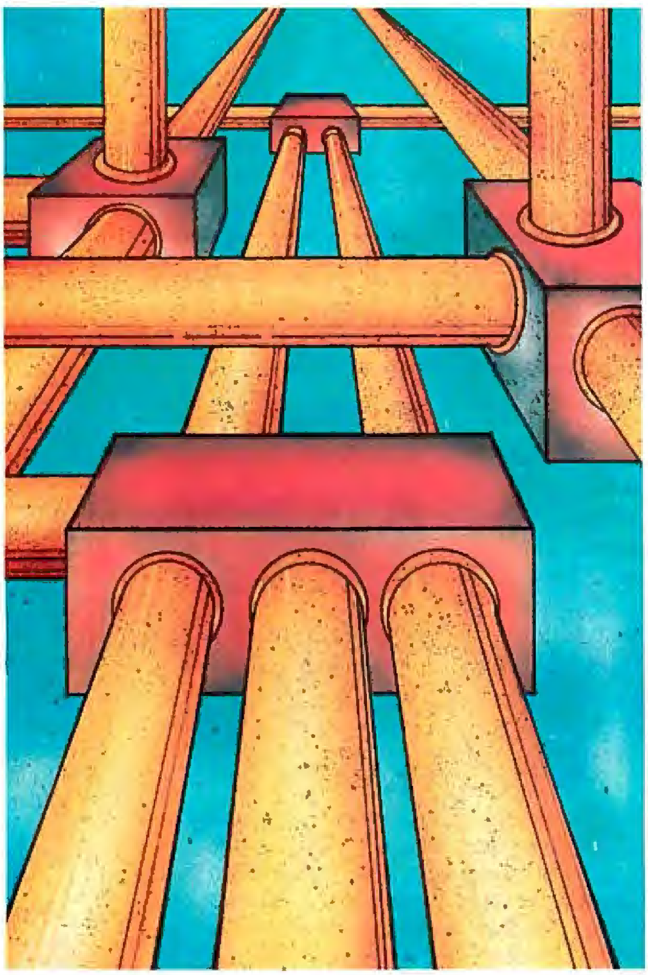
IBM is a registered trademark of International Business Machines Corp.

PRICES TOO LOW TO ADVERTISE!
(Volume users/resellers, call 408 288-8838.)

Creating filters to work with MS-DOS's pipe and redirection functions

Pipes and Filters

Paul Baker



Pipes, filters, and I/O redirection are standard features of Unix, and, with some creative programming, they can become powerful features for MS-DOS as well. Pipes let you string commands together, with the output of one becoming the input of the next. Filters perform utility-like functions while letting data flow through them. I/O redirection lets you modify the standard data source and destination devices. You can use them together to execute multiple sequential functions with little effort.

The three standard MS-DOS filters, FIND, SORT, and MORE, are designed to manipulate ASCII text files. FIND searches for a specific string of text and displays or counts those lines that include the specified text. SORT, as the name implies, sorts or alphabetizes a file by a particular column. MORE displays a file one complete screen at a time, pausing until told to continue.

Piping Programs Together

The pipe operator, a vertical bar (|), lets you give the system more than one command at a time. You place the pipe symbol between commands or program names within a string of commands, and the output of one program automatically becomes input to the next.

For example, the `DIR | SORT / +10` command groups and sorts a directory by file extension, starting at position 10. The pipe symbol between the two commands indicates that the output of the `DIR` command goes to SORT rather than to the screen. The SORT filter sorts the information based on the characters starting in position 10 of each line of the directory and sends the sorted data to the screen.

Redirecting I/O

The standard MS-DOS data source and destination are called standard input and standard output, respectively. The keyboard is the default for standard input, and the CRT screen is the default for standard output. However, you can send input and output to other devices with the redirection symbols: > redirects a program's output, < redirects a program's input, and >> adds a program's output to the end of a new or existing file.

The standard device for any error messages is called standard error. It is always the CRT screen, and you cannot redirect it. This is an important point, since you would not want error messages to appear in your redirected output.

The redirection symbols tell MS-DOS to temporarily substitute another device for standard input or standard output. For example, you might redirect standard output to go to the printer, a serial port, or a disk file, and you might redirect standard input to come from a serial port or a disk file.

Although redirection is a simple concept with a seemingly simple result, it is most important when you are using the pipe command. You usually use pipes and filters in MS-DOS without the aid of applications programs. If you didn't have the redirection symbols, you would have no way of getting a printout or saving your results without developing other programs to perform these tasks.

Programs that work with the pipe command will also work well with the redirection symbols. For my example, I displayed the sorted directory on the screen. However, I could have used the

continued

Paul Baker (Route 12, Box 461, Cleveland, TN 37311) is a telecommunications analyst.

output redirection symbol (>) to send the directory to the printer or to a disk file. For example,

```
DIR | SORT /+10 > PRN
```

sends the sorted output to the printer;

```
DIR | SORT /+10 > DIR.TXT
```

creates a file named DIR.TXT and sends the output to that file; and

```
DIR | SORT /+10 >> DIR.TXT
```

appends the sorted directory to the end of DIR.TXT, which can be a new or an existing file.

Filters versus Utilities

Filters are a form of utility program, but utilities and filters differ in how their input and output sections are designed. Utilities accept data as input and supply data as output, while filters let data flow through them.

Utility programs usually either stop and ask you for input and output filenames or let you include those filenames as parameters when you enter the program-execution command. For example, if I have a utility program named STRIP.COM that strips control characters from word-processing files, the command `STRIP INFILE.TXT OUTFILE.TXT` tells the program to accept input from INFILE.TXT and to place output in OUTFILE.TXT.

Filters work with the pipe and redirection commands. Input comes from the previous filter, from standard input, or from another input device via redirection. Output goes to the next filter or to standard output or is redirected to another output device. If you designed STRIP as a filter instead of as a utility, you could get the same result, and the program would work the same way. The command `STRIP < INFILE.TXT > OUTFILE.TXT` accepts data from INFILE.TXT and directs output to OUTFILE.TXT. Creating STRIP to work as a filter doesn't reduce its ability to operate in stand-alone mode; it lets you use STRIP in a string of commands.

For example, assume that INFILE.TXT is an ASCII text file of customer records with four fields in each record: name, address, state, and ZIP code. The command `STRIP < INFILE.TXT | FIND "TN" | SORT > OUTFILE.TXT` strips the control characters from INFILE.TXT, locates all people from Tennessee, sorts them by name, and saves the results in OUTFILE.TXT. You don't have to execute three different programs separately and supply I/O information to each one. If I had a label filter, I could add it after SORT, redirect the results to the printer, and print mailing labels.

Creating a Filter

A filter is a program that accepts a string of characters from the input, manipulates or modifies the data, and writes the revised information to the output. The input and output are the only sections of the program that are unique to filters. If you design a filter to work with pipes, it won't need to ask for input and output filenames; proper use of the pipe automatically specifies source and destination. In addition, since filters have a minimum of user interaction, they don't need extensive menus.

For example, look at INSERT.ASM, a filter that inserts data into a text file. [Editor's note: *INSERT.ASM is available in Microsoft Macro Assembler source code in print, on disk, and on BIX. See the insert card following page 208 for details. Listings are also available on BYTEnet. See page 4.*] The input data can include character strings, blanks, carriage returns, linefeeds, formfeeds, or other information. INSERT passes data from one filter to the next via temporary files that are created, maintained, and deleted by MS-DOS at the completion of the pipe operation. This process is inherent in the pipe command and is automatic as long as you design the filter to operate with the standard devices.

Listing 1, get_ready, is the first important code segment from INSERT.ASM. It determines where to find the input data. The code loads the handle or the identification of 00, the standard input. Then it calls the MS-DOS function to duplicate the handle. This function returns a new handle number in the AX register, which is to be used as the source of input. This handle could be for a temporary file, created by the pipe function, that contains the output from a previous filter; it could point to a file specified initially in the invoking command; it could reference a redirected device specified initially; or it could be the handle for the keyboard, the standard input. Then the program stores this handle in the BP register for later use and closes the duplicated file.

Listing 2, read_data, uses the duplicate file handle. It points to the input buffer and loads CX with 800 hexadecimal to tell MS-DOS to read a buffer's worth (2048 bytes) of data. Then read_data loads BX with the duplicate file handle from listing 1 that points to the input source, and the program calls the MS-DOS read function. If more data is to come from the input source, the program processes this current data and eventually returns to read_data to get more.

Output in either character or string format goes to the standard output device. Listing 3, send_byte, uses the MS-DOS service function 02h to send the character in DL to the screen. The output automatically goes elsewhere if you indicate redirection in the initial command.

All you have to design is each filter's processing section; the I/O sections could be similar in each one. Keep them small and dedicated.

Listing 1: *This section of INSERT.ASM determines where the program is to find the input data.*

```
get_ready: XOR  BX,BX          ; load handle 00
           MOV  AH,45h         ; get file duplicate
           INT  21h
           MOV  BP,AX          ; set BP to new handle
           MOV  AH,3Eh         ; close file
           INT  21h
```

Listing 2: *This segment of code reads 2K bytes into the input buffer.*

```
read_data: CLD
           MOV  DX,OFFSET data_buf   ; store in data_buf
           MOV  CX,800h              ; set to read 2048 bytes
           MOV  BX,BP                ; set BX to file handle
           MOV  AH,3Fh               ; read input data
           INT  21h
           OR   AX,AX                ; data read?
           JNZ  more_data
no_data:   IRET
```

Listing 3: *These three lines of code use the MS-DOS service call to send the output.*

```
send_byte: MOV  DL,AL          ; load character in DL
           MOV  AH,02h         ; send character to display
           INT  21h
```

Listing 4: *This section of code handles any error messages.*

```
bad_param: MOV  DX,OFFSET err1   ; point to error message
           MOV  CX,2Fh           ; send 47 bytes
           MOV  BX,0002h         ; point to standard error
           PUSH ES               ; save new data segment
           POP  DS               ; get new data segment
           MOV  AH,40h           ; send message to standard error
           INT  21h
```

Handling Error Messages

Almost by definition, error messages should go to the screen, even if you redirect the output elsewhere. This prevents error messages from ending up in your printout, or worse, nested into a data file. If you could send your error messages to a data file, you wouldn't know that you had errors until you needed to use the data. This could be frustrating.

Listing 4, bad_param, handles the error messages that might occur while you are using the INSERT.EXE filter. First it points to the error message and indicates how many bytes to send. Then it loads 0002h into the BX register; this specifies the handle related to the standard error device, which is almost always the CRT screen. Next, the program loads DS with the address of the data segment related to INSERT.EXE and calls the MS-DOS service function 40h to output the 47-byte error message to the standard error device.

The Sky Is the Limit

Once you are familiar with the input and output sections of filter programs, the sky is the limit regarding the variety of useful filters that you can develop. All you have to design is the filter's processing section; the I/O sections could be similar in each one. Remember to keep them small and dedicated, and therefore efficient. A study of INSERT.ASM can help with the programming techniques.

Small dedicated programs or filters that operate with the MS-DOS pipe and redirection symbols have a definite worth. With a little creativity and a few filters, you could handle many applications, including databases of phone and address lists, home inventories, and an index of what programs are on what disks, to mention a few. If you have any comments or suggestions about filter programs used with pipe and redirection functions, I would be interested in hearing them. ■

PRO SERVE GUARANTEES THE LOWEST PRICE FOR QUALITY COMPATIBLES OR YOUR MONEY BACK!*

PRO SERVE+™ XT TURBO

$629 COMPLETE!

NEW! *PRO SERVE+*™ 10 TURBO

$699 COMPLETE SYSTEM!

PRO SERVE+™ AT TURBO

$1169 COMPLETE!

386 Version ONLY $2799

STANDARD FEATURES:

IBM PC/XT Compatible
4/8 MHZ CPU speeds
640k RAM Memory
Two 360k Diskette Drives
8 I/O Slots
150 Watt Switching Power Supply
FCC Approved Chassis
AT Style Keyboard
High-Resolution Amber Monitor
Hercules Compatible Graphics Card
Parallel Port
User Manual
Complete and Ready to Run!

STANDARD FEATURES:

IBM PC/XT Compatible
4.77/10 MHZ CPU speeds
Keyboard selectable
640k RAM Memory
Two 360k Diskette Drives
8 I/O Slots
Reset Switch and Keyboard Lock
150 Watt Switching Power Supply
FCC Approved Chassis
AT Style Keyboard
High-Resolution Amber Monitor
Hercules Compatible Graphics Card
Parallel Port
User Manual
Complete and Ready to Run!

STANDARD FEATURES:

IBM AT Compatible
6/10 MHZ CPU speeds
1Mb RAM Memory
One High Density 1.2 Mb Diskette Drive
Hard/Floppy Disk Controller Card
8 I/O Slots
200 Watt Switching Power Supply
FCC Approved Chassis
Enhanced AT Style Keyboard
High Resolution Amber TTL Monitor
Hercules Compatible Graphics Card
On-Board Clock/Calendar
Parallel Port   User Manual
LAN & Multi-User Ready
Complete and Ready to Run!

**Many other configurations and options available. Call Pro Serve for quotes.
Requests For Bids on High Volume Orders Welcomed.**

HARD DRIVES

Seagate 20Mb w/contr. ST225	$315
Seagate 30Mb w/contr. ST238	369
Seagate 20Mb for AT ST225	289
Seagate 30Mb for AT ST4038	529
Seagate 40Mb for AT ST251	499
Seagate 80Mb for AT ST4096	929

MONITORS

Amber TTL w/Tilt Swivel Base	$89
Color RGB Monitor	269
EGA Monitor w/Card	569

ADD-ON BOARDS:

Mono Graphics w/Par. Port	$59
Color Graphics w/Par. Port	69
EGA Adapter	169
XT Multi-Function w/Clock and Ser., Par., & Game Ports	69
AT Multi-I/O w/ Ser., Par., & Game Ports	69
2MB AT RAM pansion	295
Serial Port	26

PRINTERS

Panasonic / Okidata / Citizen
Toshiba / Hewlett Packard

MOST ORDERS SHIPPED WITHIN 48 HOURS

* PURCHASE WITH CONFIDENCE!

Here is our commitment to you:

30 Day Money-Back Guarantee — *If you are not satisfied with any item you purchase from* **Pro Serve**, *you may return it within 30 days of the date of shipment for a full refund. Returns must be "as-new", not modified or damaged, with all warranty cards, manuals, and original packing intact, and must be pre-authorized by* **Pro Serve**, *and be shipped prepaid and insured.*

One Year Limited Warranty — **Pro Serve** *warrants the products it manufactures to be free from defects in materials and workmanship for a period of one year from the date of shipment. During this one year period,* **Pro Serve** *will repair or replace, at its option, any defective products or parts at no additional charge, provided the product is returned, shipping and insurance prepaid to the* **Pro Serve** *repair facility.*

TERMS: All prices subject to change without notice Shipping [illegible]B Phoenix, AZ. Minimum shipping charge is $2.50. UPS C.O.D. is $3.00 additional. Arizona residents only, add 6.7% sales tax. Allow extra two weeks shipping time when paying by personal or company check. No extra charge for MasterCard or Visa orders. Add 4% for American Express orders. All shipments sent insured. Defective merchandise may be returned for repair or exchange only. Merchandise returned for credit is subject to a 15% restocking charge. A Returned Material Authorization number obtained from **Pro Serve** in advance must be shown on the shipping label for all returns.

All systems are burned-in and tested for 24 hours. We configure systems boards, set up switches, format hard drives, perform all diagnostics, and check the system with your monitor. All systems are FCC approved: Class 15J, Part B.

IBM PC/XT and AT are registered trademarks of IBM Corp.

MasterCard / VISA / American Express

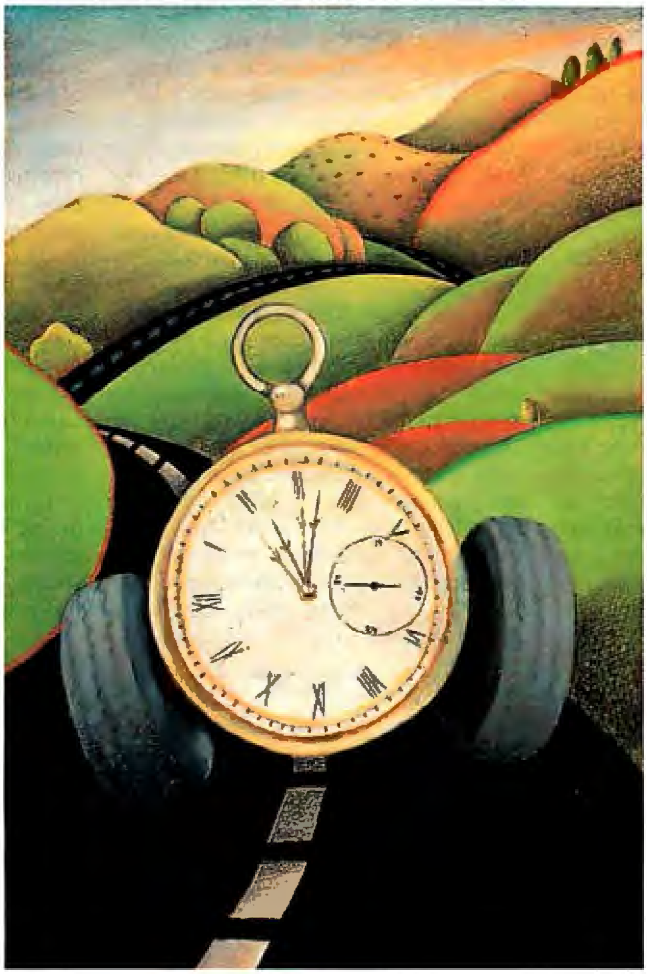
CALL TOLL FREE:
1-800-621-8285
(In Arizona, Call: 1-252-0017)
34 W. Monroe 300, Phoenix, AZ 85003
Circle 139 on Reader Service Card

A BIOS building block for compatibility

A Timing-Independent BIOS

Howard N. Cohen and John Hanel

Performance is a common way for IBM PC-compatible manufacturers to differentiate their machines from one another. The first important difference between the IBM PC AT and its clones was the processor's clock speed. Beginning with the original AT, which had a 6-megahertz clock speed, IBM and manufacturers of AT-class computers have increased performance by increasing the clock speed to 8, 10, and 12 MHz, and beyond.



A number of AT-class machines are available with user-changeable clock speeds. In these computers, the user can, through a switch on the computer or a command from the keyboard, change the clock speed among two or more possibilities. The most common switching is between a fast clock speed (10 or 12 MHz) and the 6-MHz speed of the original AT.

To accommodate these machines, and to deal with a proliferation of AT-class computers with different clock speeds, requires a BIOS that is not dependent on a specific machine clock speed for critical timing loops. In this article, we will examine problems that can arise from increasing the processor clock speed, and we will show how speed-independent BIOS facilities can handle these problems.

Timing Problems

Many early purchasers of high-performance AT-compatible machines found that some applications would not operate properly on these machines. Some of these problems were caused by the higher processor clock speed. The most common problem was with copy-protection schemes that were clock-dependent. To eliminate some of these problems, it became necessary to be able to switch speeds during machine operation.

Users, seeking to increase performance by replacing system clock crystals, also found problems using their existing BIOS. Upgrading machines from 6 to 8 MHz could cause intermittent errors, most notably in accessing floppy disk drives. These problems became so widespread that IBM inserted a timing check into its AT BIOS to stop this practice.

As technology increased capabilities, many manufacturers found themselves in the unenviable position of having to support several compatible machines, each requiring a different BIOS.

To help alleviate problems for AT-compatible manufacturers, BIOS software should provide several features missing from a BIOS designed to run at a single clock speed. First, keyboard speed-switching support should let users dynamically change the speed of their machines and get around speed compatibility problems.

Second, the BIOS should allow for timing independence. This lets one version of the BIOS run on many machines, each with different speed characteristics.

Speed Switching

A multispeed, timing-independent BIOS lets a user change system clock speed from the keyboard on machines that have hardware support for multiple processor clock speeds. Speed switching typically uses either Ctrl-Alt-1 or Ctrl-Alt-minus (keypad −) to switch to low speed and either Ctrl-Alt-2 or Ctrl-Alt-plus (key-

continued

Howard N. Cohen is president of Syenitic Software (10471 Lansdale Ave., Cupertino, CA 95014). John Hanel is supervisor of compatible software development for Award Software Inc. (130 Knowles Dr., Los Gatos, CA 95030).

Illustration: Tom Dolphens © 1987

Speed switching and timing independence are crucial AT-compatible BIOS features.

pad +) to switch to high speed. The speaker beeps once when switching to low speed and twice when switching to high speed. A change in the appearance of the cursor might or might not accompany the change in processor speed.

Internally, switching speeds is done by calling a Setspeed routine. You can modify this routine to perform whatever operations are necessary to make the hardware switch speeds. The default Setspeed routine used in the Award Modular BIOS switches speed by changing both bits 2 and 3 of the keyboard-controller output port (pins 23 and 24). This is compatible with most clone machines currently available that allow speed switching.

A user program can also call the Setspeed routine externally. To invoke the routine, a far call is made to F000:E826 hexadecimal. Prior to this call, the AL register must be set. If bit 7 = 1, the speaker beeps, signaling a speed switch. If it is 0, the speaker is silent.

The remaining 7 bits (0 through 6) are defined by the Setspeed routine installed for the particular machine. The default routine assumes a 7Fh for high speed and 0 for low speed. Other routines may support the other bit combinations for changing to additional speeds or setting the number of memory wait states.

In addition, you can execute Setspeed prior to running the power-on self-test (POST). You can configure the BIOS to run the POST in either low speed, high speed, or the default hardware speed. Also, you can configure the processor speed after the POST using the Setup program, which is resident in the BIOS.

Timing Independence

Many machine operations require an event to occur within a fixed amount of time. In most BIOSes, this time is normally computed by executing a loop of instructions a predetermined number of times. This is commonly referred to as a CPU timing loop. Figure 1 gives an example of a timing loop that will delay 3000 microseconds on an 8-MHz AT when called with CX set to 100.

When a machine runs at a different processor speed, the same timing loop will measure a different time interval. If the timing loop in figure 1 ran on a 12-MHz machine, it would delay for only 2000 μs with CX set to 100.

Using timing loops for critical timings has some problems. All the loop counters must be computed based on an expected processor speed. This means that a separate BIOS will have to be generated for each machine running at a different speed. Also, if a machine supports speed switching, the timing loops will not be correct for at least one of its speeds.

To eliminate these timing-loop problems, you can design a BIOS to be timing-independent when possible. You do this by using a hardware pulse that, on most machines, has a constant period. On the AT, every time a memory refresh occurs, bit 4 of I/O port 61H changes state.

```
; WAIT_CPU:
; This procedure waits a given period of time before
; returning to the calling routine. It is based on the
; speed of the CPU and will work correctly only on an 80286-
; based machine running at 8 MHz, one wait state, with the
; procedure starting on an even address.
;
;
;   INPUT: CX = number of 30-μs periods to wait
;   OUTPUT: CX destroyed

EVEN
wait_cpu                proc            near
wait_cpu_loop:
            push    cx
            mov     cx,15                   ; delay for 30 μs
            loop    $
            pop     cx
            loop    wait_cpu_loop           ; decrement cx and loop
            ret
wait_cpu    endp
```

Figure 1: *A CPU timing loop. (All examples of timing independence in this article are taken from the Award Modular BIOS developed by Award Software.)*

Most compatible machines do the same. Instead of counting instruction loops, the BIOS can count refresh pulses to allow time-outs in the BIOS to last the same amount of real time regardless of the CPU speed and number of wait states in memory. Figure 2 shows an example of how to use the refresh pulse for timing.

Unfortunately, not all compatible machines have similar refresh characteristics. However, there are a number of different ways to control the timing.

Short refresh (15- to 60-μs update) is used if the refresh pulses are constant and less than 60 μs apart. This is the most common refresh method because the AT operates this way.

On some machines (most notably the Intel 386 motherboard), the refresh pulse occurs much less frequently, sometimes as slow as 4 milliseconds. For these types of machines, use the long-refresh (greater than 60-μs update) timing method. This method is similar to the short refresh except that its actual timing routines have been modified to work with the longer pulse rate.

On some machines, the refresh pulse is either unavailable or not constant (e.g., the refresh rate changes when the processor speed changes). For these types of machines, you should use the CPU timing method, which uses timing loops based on the CPU speed and not on the refresh rate. This results in a BIOS that is not timing-independent.

To minimize the timing problems, the timing loops should be set for the fastest processor speed available. At lower speeds, the loops will execute longer, but this is usually less harmful than having a timing loop finish too quickly. Timing loops executing too quickly can result in a high number of incorrect data reads from disks or "device not ready" situations from the disk drive and I/O ports. An extra-long timing loop might affect system performance but generally will not result in a deluge of errors returned.

Whenever the BIOS reaches an area of critical timing, it first determines what timing method it should use. Based on what you have configured, the BIOS calls the appropriate timing routines.

Several parts of the BIOS have critical timing dependencies. The first is the refresh test, high range/low range. During the POST, the BIOS tests the refresh pulse occurring on port 61H bit 4 for a reliable rate. It does this by counting the number of refresh cycles that occur in 55 ms. It then compares the result it gets with the high and low range and expects it to be between these two figures. This test executes only if you are using the refresh timing.

continued

An important benefit is the ease with which system designs can be upgraded.

For example, if a machine has a refresh pulse every 15 μs, a "cycle" lasts 30 μs, and the number of cycles in 55 ms will be 55 (ms) / 30 (μs) = 1833.33 cycles. We recommend that you set the high value to this plus 25 percent, so in this example it will be 1833.33 + (0.25 * 1833.33) = 2291 refresh cycles. The low value should be set to 25 percent below the normal value.

The next part of the BIOS that has a critical timing dependency is the floppy disk motor spin-up. Whenever you turn on a floppy disk motor, you must let it spin up to speed before you try to read or write to the floppy disk. The length of this period of time is determined by multiplying the value from the floppy disk parameter table times ⅛ second.

The third part is the floppy disk head settle. After the head seeks to the appropriate track on a floppy disk, a short wait must occur for the read head to stop vibrating so that the read can occur reliably. The length of this period of time is determined by multiplying the value from the floppy disk parameter table by 1 ms.

Printer initialization must occur before printing any characters. The printer-initialization line is set and held low for a set period of time, then it is set high to complete the initialization. The recommended period of time is approximately 65 ms.

The keyboard check occurs during the POST. A command to initialize the keyboard is sent to the keyboard. The BIOS then waits for the keyboard to respond. If the keyboard does not respond in the time specified, the BIOS assumes a keyboard is not present. This wait should be approximately 100 ms.

The next part is the floppy disk command wait. Before the BIOS sends a command byte to the floppy disk controller, it should wait up to ½ second for the controller to signal that it is ready to receive a command.

The seventh BIOS part with critical timing dependencies is the floppy disk operation complete. After a command is issued to the floppy disk controller, the BIOS waits for the controller to complete the command for 1½ seconds.

Next is the floppy disk status wait. After the floppy disk controller has completed a command, the BIOS must read the status bytes from the controller. The BIOS should wait for ½ second for each status byte to come back from the controller before concluding that a controller malfunction has occurred.

The fixed disk controller busy is also timing-dependent. Before sending commands or reading status from the fixed disk controller, the BIOS must wait for approximately 8 seconds for the controller to signal that it is ready before an error condition is returned.

Next we have the fixed disk controller complete. After a command is issued to the fixed disk controller, the BIOS waits for the controller to complete the command for 10 seconds before signaling an error.

The next timing-dependent BIOS part is the fixed disk read/write long. When performing a read/write long on a fixed disk, the BIOS should wait 2 ms for the controller to accept/send the ECC (error correction code) bytes.

Next is the communication wait. When the BIOS performs data transfers over the RS-232C port, it should wait approximately 1.2 seconds for the external device to respond.

The last one is printer busy. When the BIOS prints characters to the printer, it waits until the printer is not busy before sending the character. The time-out value is looked up in the ROM BIOS data area and multiplied by 1 second to determine the time-out for the printer.

```
;WAIT_REFRESH:
; This procedure also waits a given time period before
; returning to the calling routine. It gets its timing from
; port 61H, bit 4, and will thus run on either 80286 or
; 80386 machines at any clock speed greater than 6 MHz,
; with any number of wait states in memory.
;
; Port 61H, bit 4, changes state every time a memory
; refresh occurs. Memory refresh normally occurs
; once every 15 µs, so port 61H, bit 4,
; completes a cycle once every 30 µs.
;
; The Modular BIOS uses this bit because very few
; applications will reprogram the memory-refresh time.
; Applications programs that want to be CPU-speed-
; independent should use timer channel 2. This will also
; let them run on 8086-based machines.
;
;
;   INPUT: CX = number of 30-µs periods to wait
;   OUTPUT: AL, CX destroyed
WAIT_REFRESH proc near

wr_state_0:
              in     al,61H ;wait while port 61H, bit 4
              test   al,010h ;is low
              jz     wr_state_0
wr_state_1:                 ;wait while port 61H, bit 4,
              in     al,61H ;is high
              test   al,010h
              jnz    wr_state_1
              loop   wr_state_0
                            ;30 µs have elapsed,
                            ;so decrement counter and wait
                            ;again.
              ret
wait_refresh endp
```

Figure 2: *A refresh timing loop.*

Building Block of Compatibility

Manufacturers of AT-class computers face the task of providing unique features for products that, in order to be salable, must provide a specific level of functionality. A BIOS that is not dependent on a particular clock rate allows the system designer freedom to alter the system clock to obtain peak performance without requiring massive custom programming. An important benefit is the relative ease with which system designs can be upgraded, because a level of uncertainty (will the software still work?) has been eliminated.

An understanding of the principles used in writing the timing-independent BIOS is a major step toward designing hardware and software compatible across a broad range of computers. ■


SELECT 5 BOOKS
for only $3.95
(values to $129.75)
80386 A Programming and Design Handbook
2937 $29.95
Counts as 2
Desktop Publishing
Business Programming
Graphics
Computer Languages
Hardware Interfacing
dBASE III PLUS Programmer's Reference A Sourcebook Of Programming Techniques
2856 $25.95
TRUE BASIC A COMPLETE MANUAL
1970 $22.95
1921P $16.95
LOTUS 1-2-3 SIMPLIFIED
2748 $21.95
ARTIFICIAL INTELLIGENCE THEORY, LOGIC AND APPLICATION
2671P $12.95
POWER TURBO PROLOG
2782 $22.95
UNIX & XENIX DEMYSTIFIED
1874 $21.95
GRAPHICS PROGRAMS IBM PC
1556P $15.50
Computer Technician's Handbook
1939 $25.95
2838 $19.95
2700 $49.95
Counts as 3
IBM PERSONAL COMPUTER
1496P $11.50
Programmer's Challenge
2637 $29.95
Counts as 2
Membership Benefits • Big Savings. In addition to this introductory offer, you keep saving substantially with members' prices of up to 50% off the publishers' prices. • Bonus Books. Starting immediately, you will be eligible for our Bonus Book Plan, with savings of up to 80% off publishers' prices. • Club News Bulletins. 14 times per year you will receive the Book Club News, describing all the current selections—mains, alternates, extras—plus bonus offers and special sales, with hundreds of titles to choose from. • Automatic Order. If you want the Main Selection, do nothing and it will be sent to you automatically. If you prefer another selection, or no book at all, simply indicate your choice on the reply form provided. As a member, you agree to purchase at least 3 books within the next 12 months and may resign at any time thereafter. • Ironclad No-Risk Guarantee. If not satisfied with your books, return them within 10 days without obligation! • Exceptional Quality. All books are quality editions especially selected by our Editorial Board.
WordPerfect BOOK
2757 $24.95
PROGRAM YOUR IBM PC TO PROGRAM ITSELF!
1898P $12.95
HIGH-SPEED ANIMATION AND SIMULATION FOR MICROCOMPUTERS
LEE ADAMS
2859 $29.95
Counts as 2
8-BIT & 16-BIT MICROPROCESSOR COOKBOOK
1643 $19.95
LISP THE LANGUAGE OF ARTIFICIAL INTELLIGENCE
2620 $25.95
Symphony
1948P $18.95
dBASE III PLUS
2808P $19.95
PROGRAMMING WITH R:base 5000
2666 $28.95
Counts as 2
MS-DOS & PC-DOS
1848P $15.95
When it's new and important in business or personal computing,
The Computer Book Club has the information you need . . .
at savings of up to 50% off publishers' prices!
C PROGRAMMERS UTILITY LIBRARY
2855 $24.95
PLAYING THE STOCK & BOND MARKETS WITH YOUR PERSONAL COMPUTER
1251P $10.25
CAD/CAM PERSONAL COMPUTERS
1852 $21.95
PROGRAMMING IN C
1873P $17.95
The Computer Book Club®
P.O. Box 80, Blue Ridge Summit, PA 17214
Please accept my membership in The Computer Book Club® and send the 5 volumes listed below, billing me $3.95 plus shipping and handling charges. If not satisfied, I may return the books within ten days without obligation and have my membership canceled. I agree to purchase at least 3 books at regular Club prices (plus shipping/handling) during the next 12 months, and may resign any time thereafter.
Name
Address
City
State/Zip
Phone
Valid for new members only. Foreign applicants will receive special ordering instructions. Canada must remit in U.S. currency. This order subject to acceptance by The Computer Book Club®
BY-1087
DOCUMENT YOUR SOFTWARE
1724P $13.50
SERIOUS PROGRAMMING IN BASIC
2650P $14.95
IBM PC EXPANSION GUIDE
1911P $17.95
30 CUSTOMIZED MICROPROCESSOR PROJECTS
2705 $22.95
LOTUS 1-2-3
2771 $25.95
COMPUTER PERIPHERALS THAT YOU CAN BUILD
2749 $22.95
C PROGRAMMING BUSINESS APPLICATIONS
2857 $25.95
TURBO PASCAL PROGRAMMING with applications
2627P $17.95
(Publishers' Prices Shown)
All books are hardcover unless otherwise indicated.
© 1987 THE COMPUTER BOOK CLUB®
Blue Ridge Summit, PA 17214

Put the World at Your Fingertips!

PC-GLOBE...the remarkable new program that gives you an **instant profile** of 177 different countries

ONLY **59^{95}**

Registered owners will be entitled to special pricing on annual updates.

Enhance Educational Programs

Obtain Instant Data For Marketing

Plan Vacations or Business Trips

Packed with up-to-the-minute data, hundreds of maps and full-color graphics, **PC-GLOBE** literally puts an atlas and encyclopedia on your computer screen.

PC-GLOBE has two fully integrated sections—Maps and Database. At the touch of a key...

- Pinpoint any of 177 different countries. The selected country will be highlighted for instant identification.
- Access maps of any of six different continents. On each map, locate countries or display the following overlays: 1) Major Cities, 2) Elevation and 3) Population Density.
- Access individual maps for any of the 177 countries. Overlays display Major Cities and Elevation.

The Database portion of the program offers the following information for each of the 177 countries—

- National population statistics & growth trends
- Major cities with their populations
- Time zones
- Health statistics
- Age distribution
- Political parties
- Government leaders
- Major industries & agricultural products
- Imports & exports
- GDP trends
- Balance of trade statistics
- World data & commodity rankings
- Literacy rate
- Percent urbanization
- Language & culture
- Tourist attractions & climate
- And much more!

The Database may be accessed from any point in the program, at any time. This portion of **PC-GLOBE** is specifically designed to give you the vital facts you need in the simplest, most practical format. The data appears in a colorful array of charts, tables and informational capsules, reinforced with useful and authoritative commentary.

All screen displays can be printed out or modified by any number of screen-capture programs. In this way, PC-GLOBE can be used to prepare customized country reports and presentations.

PC-GLOBE contains **six diskettes** and runs on IBM®-PC/XT/AT or compatible computers with two floppy drives or one floppy drive and a hard disk. The program works with DOS versions 2.0 or later and requires 256K RAM and a color graphics board.

CALL 1-800-255-2789
(orders only)
or contact your local dealer. For additional information, contact CSI.

CSI

Cornwell Systems, Inc.
P.O. Box 41852
Phoenix, AZ 85080
(602) 869-0412

Circle 258 on Reader Service Card (Dealers: 259)

Three construction examples that add a port to your IBM PC's bus

Three Bus Interface Designs for the PC

James R. Drummond

In this article I'll take a look at how to interface three popular parallel-interface chips to almost any IBM PC bus. Using the extra I/O ports these chips provide, you can add timing capabilities, A/D converters, motor drives, and many types of sensors to your IBM PC to expand its capabilities far beyond what its designers planned. I'll demonstrate one simple interface using the Intel 8255 chip, and two more complex interfaces using the Motorola MC6821 and the National Semiconductor NSC810A chips. The latter two examples take advantage of programmable-array-logic technology to reduce the interface design's complexity to a few chips.

I won't cover the characteristics of each chip except insofar as they refer to the bus interface. Each chip has its own advantages, but all the chips have two 8-bit parallel ports in common. All the interfaces described are designed to be capable of 16-bit access where possible. I have not provided explicit layout diagrams, but you should have enough information to adapt these designs to your specific needs. For detailed information concerning the capabilities of these chips and their registers, consult the manufacturer's literature.

The IBM PC Bus

The IBM PC bus consists of 62 connections, as shown in figure 1. Many of the pins are used for direct memory access and interrupt handling, and I won't consider them here. I'm concerned with the address lines (A0 to A19), data lines (D0 to D7), and control lines AEN, RESET DRV (which I'll call RESET for the remainder of this article), CLK, I/O CH RDY, I/O read ($\overline{\text{IOR}}$), and I/O write ($\overline{\text{IOW}}$). Power is available for peripherals on lines B1 and B31 (ground) and on lines B3 and B29 (+5 volts). All the bus lines are TTL-compatible.

Figure 2 shows some of the signals involved in standard IBM PC I/O cycles. It shows a word I/O cycle, which uses two I/O cycles to address successive I/O bytes. This simplifies programming by avoiding "byte shuffling" (i.e., the 2 bytes are read in the proper order—least-significant byte in the lowest address, most significant byte in the higher address for word organization). The signals on address lines A0 through A9 determine which card is selected on the bus. The small interval between the two cycles indicates that data throughput can be high, but not all peripheral chips can handle two accesses so close together in time.

The clock signal is a 2-to-1 mark space ratio with a low time of 140 nanoseconds and a high of 70 ns. A single byte I/O access requires five clock cycles and completes on the first T4 cycle. Therefore, byte I/O can complete in a minimum of 1.02 microseconds at 4.77 megahertz. "Turbo" IBM PC-compatibles can have a faster timing.

You can extend the PC's read/write cycle using the I/O CH RDY line. When a slow card is selected, it activates the tristate buffer attached to the I/O CH RDY line, and the PC senses this level on the leading edge of the T2 cycle. If I/O CH RDY is low (device not ready), the PC automatically inserts one wait state into the I/O cycle, and the line is sensed again. Figure 2 shows the PC adding one extra clock cycle (TW^1) in the first I/O cycle, since I/O CH RDY is

continued

James R. Drummond (University of Toronto, McLennan Labs, 60 St. George, Toronto M55 1A7, Canada) teaches electronics and microcomputer interfacing for physicists.

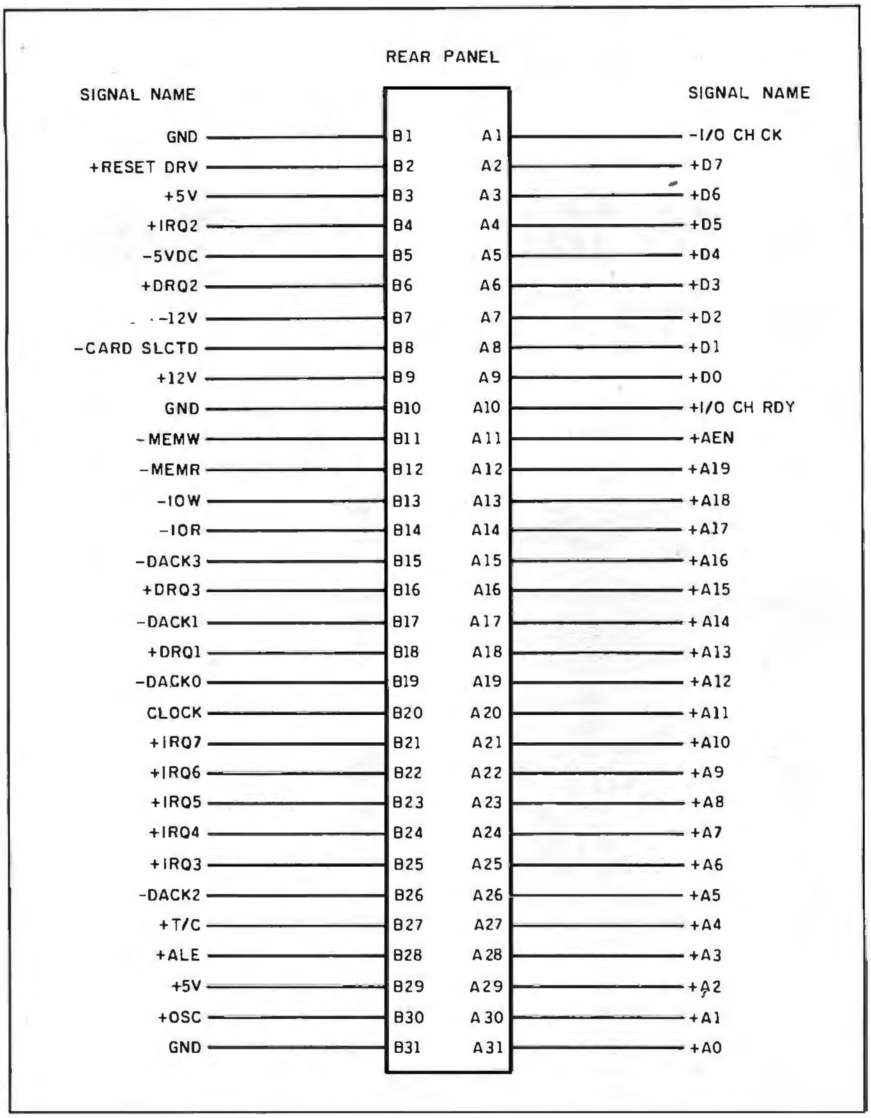


Figure 1: *IBM PC bus connections. All signals are TTL-compatible. (Figure courtesy of IBM Corp.)*

low at the end of the T2 cycle. Note that when the card tristates the driver for the I/O CH RDY line (shown in the diagram with the level at the halfway point), the line is actually pulled up to the ready state. Delaying the I/O cycle by more than a few microseconds will cause problems with the memory refresh system. The designs supplied here normally won't require more than a few extra clock cycles to operate properly.

The IBM PC has a most annoying restriction. Although the 8088 processor can use all 15 address lines to access 64K bytes of I/O space, only 10 address lines (A0 to A9) are actually decoded for I/O, resulting in a 1024-byte I/O space. IBM assigns standard I/O locations to many devices in this 1K-byte region, and these assignments are shown in table 1. At first glance, it might appear that there is free space in the "prototype card" region, which is addresses 300 through 31F hexadecimal. However, many third-party vendors use the prototype-card region for peripheral boards, so you might have to look elsewhere for I/O space. If you don't have many cards installed in your PC, one solution is to use the unoccupied assigned I/O locations. I've allowed for this situation by designing these circuits with switches to set the address so that you can move them to any unused region of the I/O space.

Unfortunately, 1024 I/O locations aren't enough when you have many installed cards splitting up the assigned I/O space, and some of the peripheral chips you might want to use have up to 26 registers each. Interfacing one such chip is usually easy, but when a design requires eight of these chips, the solution can get complicated. You can circumvent the problem by using the remaining six ad-

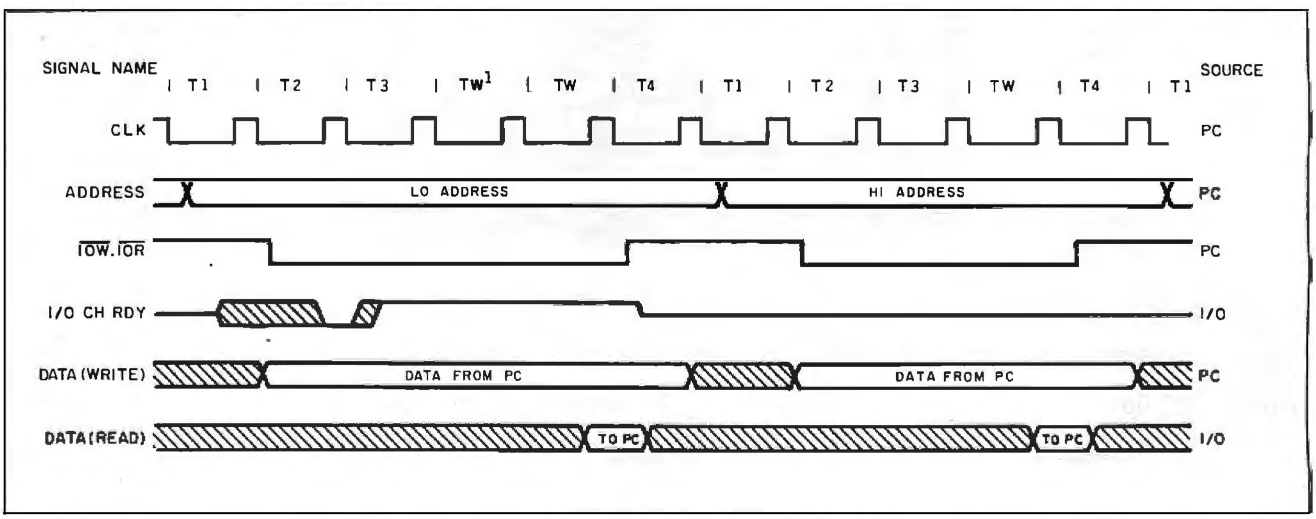


Figure 2: *Timing signals for an IBM PC word I/O cycle. The CLK interval is 210 ns. Note that the word cycle is actually two byte I/O cycles and that I/O CH RDY is used to add an extra clock cycle to the first I/O cycle.*

dress lines, regardless of what the IBM PC does. For example, the 26 registers of the peripheral chip mentioned earlier can be decoded with the least-significant address lines (A0 through A5), the card address decoded with the next address lines (A6 through A9), and the chip selected using the upper six address lines (A10 through A15).

This hardware trickery might seem awkward, but you can make it invisible to the programmer if the various addressing elements used by the software are defined symbolically. Thus, to access timer 1 on the fifth NSC810A chip on a card at address 300 hexadecimal using Turbo Pascal, you can define:

```
const
  CARD = $300;
  TIMER_1 = $012;
  NSC_5 = $2400;
```

To have the program read a 16-bit value, you write:

```
timer_value :=
  Portw [CARD or NSC_5 or TIMER_1];
```

or something similar. The actual values of the symbols are irrelevant to the programmer.

Table 1: *The assigned I/O locations in the IBM PC I/O space. (Table courtesy of IBM Corp.)*

Hex address range	Use
000–00F	DMA chip 8237A-5
020–021	Interrupt 8259A
040–043	Timer 8253-5
060–063	PPI 8255A-5
080–083	DMA page registers
0AX	NMI mask register
0CX	Reserved
0EX	Reserved
200–20F	Game control
210–217	Expansion unit
220–24F	Reserved
278–27F	Reserved
2F0–2F7	Reserved
2F8–2FF	Asynchronous communications (secondary)
300–31F	Prototype card
320–32F	Fixed disk
378–37F	Printer
380–38C*	SDLC communications
380–389*	Binary synchronous communications (secondary)
3A0–3A9	Binary synchronous communications (primary)
3B0–3BF	IBM monochrome display/printer
3C0–3CF	Reserved
3D0–3DF	Color/graphics
3E0–3E7	Reserved
3F0–3F7	Disk
3F8–3FF	Asynchronous communications (primary)

*Since addresses overlap, you cannot use both communications options at once.

Software Considerations

One of the overlooked difficulties of interface design is potential problems with software. For the IBM PC, the Intel 8088 processor uses special instructions (`IN`, `OUT`) to address the I/O space. Performing I/O won't be easy unless the language you use has a mechanism to access these instructions. For example, Turbo Pascal has a complete set of built-in I/O instructions using the two "arrays" `Port[0..$3FF]` and `Portw[0..$3FF]`, which access the I/O locations as bytes and words, respectively. Thus, to read a byte from I/O location 300h, the instruction is `k := Port[$300];`, with equally simple instructions for writing to a port. Some other languages have similar constructs, but some do not. You should check out a prospective language carefully before purchasing it for I/O programming.

Conversely, software that can perform I/O can be an aid to debugging hardware problems. I have written three programs that test designs using the 8255, MC-6821, and NCS810A, respectively. The programs are written in Turbo Pascal and simply write and read a word from an I/O port. This might not seem like much, but it can help you track down a lot of problems due to miswired address or data lines. [Editor's note: *T8255.PAS, T6821.PAS, and T810.PAS are hardware test programs written in Turbo Pascal. PAL6821.LST and PAL810.LST are PAL programmer equations. These files are available on disk, in print, and on BIX; see the insert card following page 208 for details. They are also available on BYTEnet; see page 4.*]

Hardware Design Considerations

Because the IBM PC bus can have many cards attached to it, you must drive it carefully, and each card must not load it excessively. In practice, this requires that you use no more than about two low-power Schottky TTL (LSTTL) inputs per slot and "bus buffer" chips to drive the data lines. A tristate LSTTL output can drive the I/O CH RDY line. All the interfaces described here use the 74LS245 for the bus buffers. Its logical pin-out makes the wiring easier.

The interface card must decode its own address, which you achieve by using a digital comparator that compares the address on the bus to that of the card. The card address is determined by the settings on a DIP switch. The 74LS682 is used for the comparator, since it has built-in pull-up resistors on one set of the inputs. Thus, to compare the corresponding address bit to a 1, the pin is left open, and to compare to a 0, the pin is grounded, requiring only a single-pole DIP switch. The AEN line should also be compared to a 0 on the comparator. This will eliminate the possibility of spurious address decoding during a bus DMA cycle, when AEN is high (1).

I built all these cards on standard wire-wrap cards using precut wire. I have found this technique quick and reliable, although the board does not look pretty if it is extremely dense. I also solder 0.1 microfarad decoupling capacitors across the power supply and ground pin of every IC socket on the board, which reduces power-supply noise considerably.

Many IBM PC clones run with a faster clock than the IBM PC, so a card design that works well on a standard 4.77-MHz IBM PC might not run at all on an 8-MHz PC. Some "turbo" PCs also alter the bus timing in various ways to make it look more like a standard PC bus. Hence, a large number of possible bus timings exist that you can't anticipate in your design. The best solution is to use high-speed chips wherever possible. Using fast chips, I've run most of these designs on many PC variants, but I can't try all of them.

Now that I've covered the basics of the PC bus signals and some of the support-

continued

software and hardware considerations, I want to look at the interface chips and see how you can connect them to a PC bus.

The Intel 8255

The Intel 8255 programmable peripheral interface (PPI) is a fairly simple parallel-port chip. Its main advantages are a simple interface and three bidirectional 8-bit ports. Figure 3 shows the chip pin-outs, and table 2 lists the register assignments. The values you place in the 8255's control register determine which groups of lines are inputs and which are outputs. For more details of the operation of the various registers, consult the Intel data books.

You would expect that an Intel chip would interface easily on the IBM PC bus, which is effectively the bus for an Intel 8088 processor in "maximum mode." This is indeed the case: The various control lines go straight onto the pins of the 8255,

Table 2: *The register map for an Intel 8255. Ports A and B cannot be accessed as a word because of the 8255's slow cycle time.*

A1	A0	Register	Access
0	0	Port A	R/W
0	1	Port B	R/W
1	0	Port C	R/W
1	1	Control	W

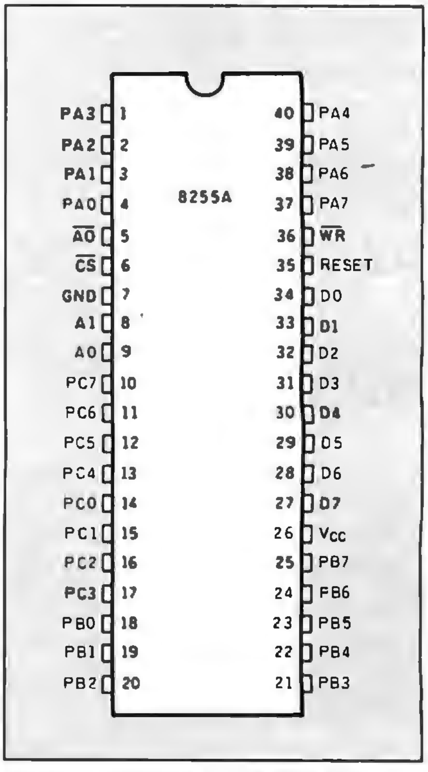


Figure 3: *Pin-outs for the Intel 8255.*

as shown in figure 4. This circuit uses a minimal IBM PC interface consisting of an 8255 chip, a 74LS682 used as the address decoder, and a 74LS245 used as a bus buffer. You don't need to delay the I/O cycle, so you don't use the I/O CH RDY line. Note that, although the bus buffer is activated by the board select signal ($\overline{BS}$) from the decoder, the bus buffer is arranged to direct data onto the card unless a read cycle is in progress ($\overline{IOR}$ low). This prevents the possibility of a bus collision with the output from another card, which can occur when the 74LS245 decodes a bus transient as its board address and activates $\overline{BS}$.

There is a slight deficiency of the interface: since it does not use address line A2, eight I/O locations are decoded, even though the 8255 has only four registers. The effects of this address ambiguity are small: a slight loss of I/O space and duplication of the registers in the upper 4 bytes of the decoded space.

If you would like more than one 8255 on an interface card, you must do some additional address decoding using a 74LS138 decoder. You can take the additional address lines required from either just above the register selects (address line A2 and up), as shown in figure 5, or from above the PC's normal I/O decoding area (address line A10 and up), as shown in figure 6. In the former case, the I/O space occupied by the card expands from eight to 32 addresses; in the latter case, it doesn't. The availability of I/O space (or the lack of it) will determine which design option to use.

The one disadvantage of the 8255 is that you cannot use the word addressing as discussed for the PC bus. Why? The 8255 requires a minimum of 850 ns be-

continued

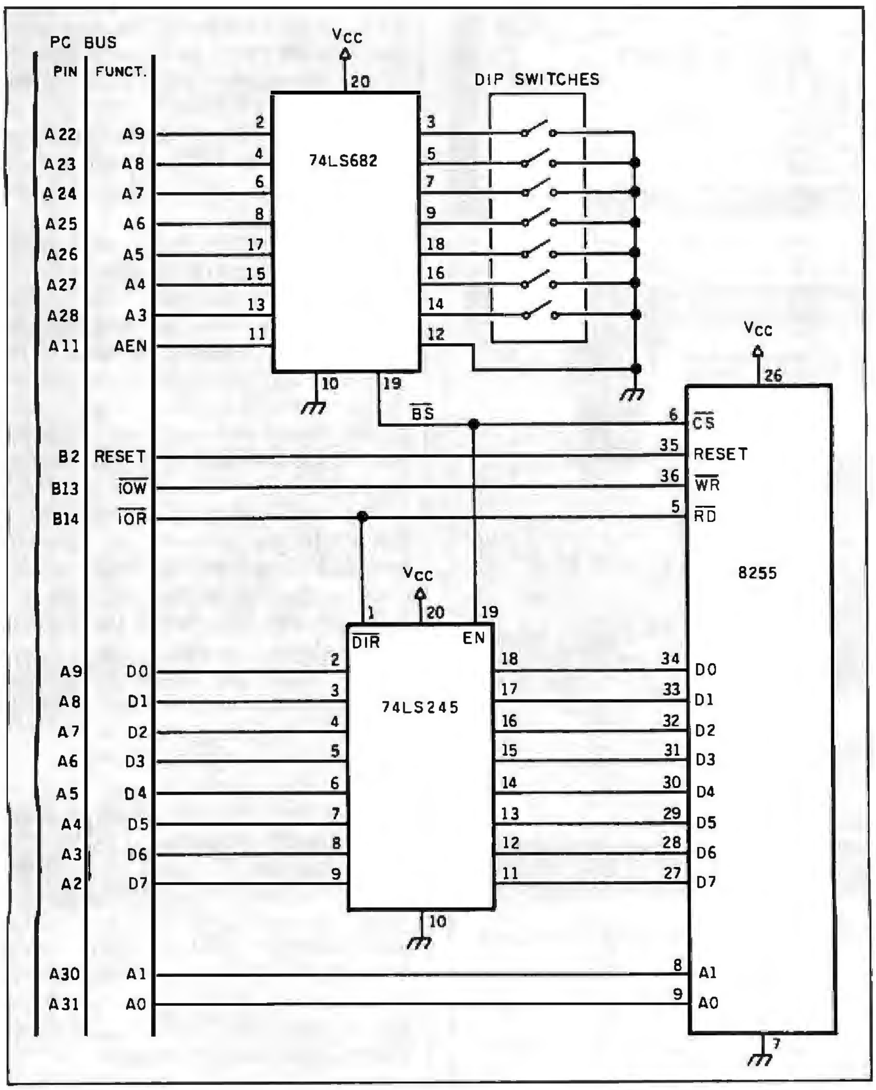


Figure 4: *A minimal interface circuit using the 8255. The 8255 response time is sufficient, so you don't need to use I/O CH RDY to extend PC I/O cycles. You do not use address line A2.*

tween successive I/O cycles, and the IBM PC allows 420 ns—far too little. Figure 7 shows how to avoid this restriction: The 8255s are used in pairs, with one chip corresponding to the high byte and the other chip corresponding to the low byte of the address. Table 3 shows the register map for word I/O decoding with this circuit.

You don't need to buffer the bus lines, even if many 8255s are on the card, because the 8255 represents an almost negligible load to the system bus compared to an LSTTL load. On a normal IBM PC, a standard 8255 will be adequate to meet the timing requirements, but a PC with a faster clock requires an 8255-5.

The Motorola MC6821

The Motorola MC6821 peripheral interface adapter (PIA) is a venerable chip that is still useful and inexpensive. It is a fairly simple parallel-port chip with two bidirectional 8-bit ports. The pin-outs for this chip are shown in figure 8. Note the three chip select lines: CS0, CS1, and $\overline{CS2}$. By using CS0 as the master select line and taking an address line to CS1 on one 6821 and to $\overline{CS2}$ on another 6821, you can address the two chips without additional circuitry.

Table 4 shows the register map for the MC6821. Again, you should consult the manufacturer's data books for detailed information on using and programming this chip. If you reverse the sense of the RS0 and RS1 register select lines as shown in table 5, the two I/O ports occupy successive I/O addresses. This lets you use the MC6821 as a 16-bit port on the PC bus.

The disadvantage of the MC6821 is that it's designed to work with a continuous clock signal (E). You simulate the E clock by dividing the IBM PC clock by 4 with a control PAL, which yields a 1.2-MHz E clock. This signal is suitable for a 6821A, which can handle a maximum clock frequency of 1.5 MHz. For an 8-MHz PC, the PAL generates a 2-MHz E clock that requires an MC6821B.

Figure 9 shows the bus timings for an MC6821 read cycle. Because the E clock counter is not synchronized to the PC's bus cycles, the PC might start an I/O cycle at an improper point in an E clock cycle. The control PAL achieves synchronization by using the I/O CH RDY line to delay the PC's I/O cycle until an MC6821 access begins. At this point, the control PAL releases I/O CH RDY, and the PC I/O cycle is allowed to complete.

For a read cycle, the data is held on the bus by holding the E clock high until the end of the PC I/O cycle. For a write cycle, the write is completed by releasing the E

continued

Osborne/McGraw-Hill's

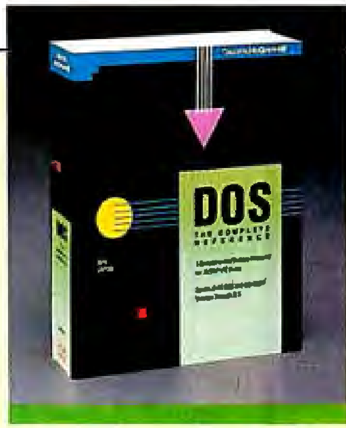
Indispensable Complete Reference Series

1-2-3®: The Complete Reference

by Mary Campbell

Every Lotus® 1-2-3® command, function, and procedure is thoroughly explained and demonstrated in "real-world" business applications. Includes money-saving coupons for add-on products. 892 pages.

$22.95, A Quality Paperback,
ISBN 0-07-881005-1



DOS: The Complete Reference

by Kris Jamsa

Has all the answers to all your questions on DOS through version 3.X. This essential resource is for every PC-DOS and MS-DOS® user. 1045 pages.

$24.95, A Quality Paperback,
ISBN 0-07-881259-3

dBASE III PLUS™: The Complete Reference

by Joseph-David Carrabis

Conveniently organized so you can quickly pinpoint all dBASE III® and dBASE III PLUS™ commands, functions, and features. 768 pages.

$22.95, A Quality Paperback,
ISBN 0-07-881012-4

C: The Complete Reference

by Herbert Schildt

For all C programmers, beginners and seasoned pros, here's an encyclopedia of C terms, functions, codes, and applications. Covers C++ and the proposed ANSI standard. 740 pages.

$24.95, A Quality Paperback,
ISBN 0-07-881263-1

AVAILABLE NOW at Fine Book Stores and Computer Stores Everywhere.

Or Call Our Toll-Free Order Number **800-227-0900**
800-772-2531 (In California)

Available in Canada through McGraw-Hill Ryerson, Ltd. Phone 416-293-1911

Osborne **McGraw-Hill**
2600 Tenth Street
Berkeley, California 94710

Trademarks: Lotus and 1-2-3 are registered trademarks of Lotus Development Corp. dBASE is a registered trademark and dBASE III PLUS is a trademark of Ashton-Tate. MS-DOS is a registered trademark of Microsoft Corp.

Figure 5: *A circuit design that interfaces more than one 8255 to the PC bus.*

Table 3: *The register map of the two 8255s in the circuit of figure 7. This circuit allows word access, since the two chips can be addressed in word I/O sequence.*

A2	A1	A0	Register	Access
0	0	0	Chip #0 Port A	R/W
0	0	1	Chip #1 Port A	R/W
0	1	0	Chip #0 Port B	R/W
0	1	1	Chip #1 Port B	R/W
1	0	0	Chip #0 Port C	R/W
1	0	1	Chip #1 Port C	R/W
1	1	0	Chip #0 control	W
1	1	1	Chip #1 control	W

Table 4: *The register map for the Motorola MC6821.*

A1	A0	Register	Access
0	0	Port/DDR A	R/W
0	1	Control A	R/W
1	0	Port/DDR B	R/W
1	1	Control B	R/W

Table 5: *The register map for the Motorola MC6821 with the register select signals RS0 and RS1 reversed. The chip can be accessed with a word I/O cycle.*

A1	A0	Register	Access
0	0	Port/DDR A	R/W
0	1	Port/DDR B	R/W
1	0	Control A	R/W
1	1	Control B	R/W

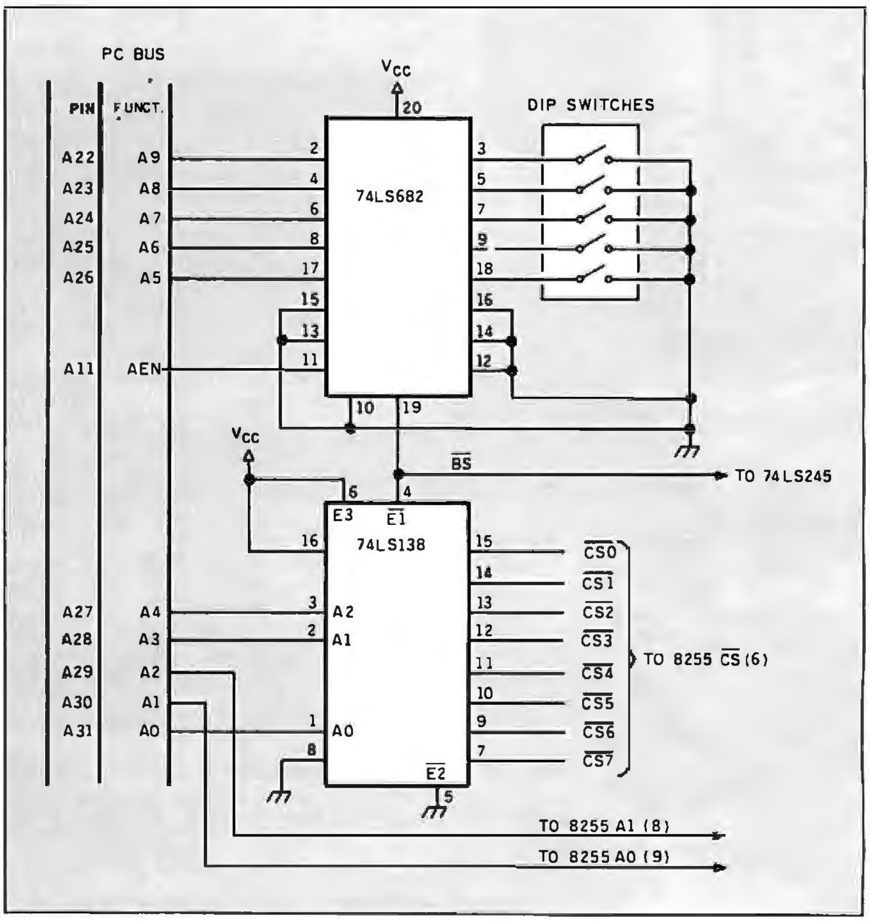


Figure 7: *Circuit design lets you access two 8255 chips as a word.*

Your logical choice for printer & data sharing.

It's your choice.
Now you can give every PC user in your office *freedom of choice:* the choice to send their output to any printer — serial or parallel — anytime.
And all you need is *The Logical Connection*™ Version 3.0.

Instant switching.
Just "pop-up" the *memory resident* switching menu, cursor-point to the device you want to switch to, and press *ENTER*.
The Logical Connection handles all the print spooling, protocols and parallel/serial conversion automatically.

Preset configurations.
10 Preset configurations give multiple computers unlimited "switch and share" access to printers, plotters and modems — and each other.
Just cursor-point to a graphic "picture" of the configuration you want and download it with a couple of keystrokes, using any IBM-PC™ or compatible. If you need a *custom* configuration, you can easily modify one of the *presets* — or create your own.

Why network if you don't need to?
For peripheral sharing applications *The Logical Connection* can outperform a LAN, right out of the box — at a far lower cost.
The Logical Connection lets minicomputers and mainframes share peripherals with PCs. Or, let many devices share a single expensive modem connection.
And for really *big* applications, you can "daisy-chain" up to 45 *Logical Connection* boxes together, up to 3/4 of a mile apart.

> PC MAGAZINE EDITOR'S CHOICE
> In April, 1987 *PC Magazine* named **The Logical Connection** "Editor's Choice among *all* printer sharing devices reviewed." Author Winn L. Rosch called it ". . . more than logical — maybe inspired!"

An easy choice.
The Logical Connection is easy to set up and easy to use. Best of all, it's easy on the pocketbook. For price per—formance and ease of use, nothing else even comes close.
So if you're looking for the *best* way to share your valuable computers and peripherals, there's only one logical choice. Get the logical connection today.
For the name of your nearest dealer, or to order direct, call *Fifth Generation Systems at* **1-800-225-2775.**

Fifth Generation
SYSTEMS, INC.
Innovative Products Using Today's Technology
11200 Industriplex Blvd., Baton Rouge, LA 70809
SALES:
2691 Richter Ave., Suite 107, Irvine, CA 92714
(800) 225-2775 • (714) 553-0111

*The Logical Connection is a trademark of Fifth Generation Systems, Inc. • IBM-PC is a trademark of International Business Machines, Inc.
*This product is in no way associated with or has originated in Fisher Scientific or Allied Corporation.

clock signal before the end of the PC cycle. This ensures that the PC does not remove the write data before the 6821 has latched it. The PAL also enforces a delay of two complete cycles of the E clock between reading bytes during a word access.

The use of a 16R4 PAL greatly simplifies the operation of this card so that the design appears barely more complicated than the 8255 design. This illustrates the great advantage of using PALs in a circuit design wherever possible. The complete circuit is shown in figure 10a, and the PAL equations are shown in PAL-6821.LST.

By letting the card decode eight byte locations, you can use the A2 line as shown to decode two MC6821 chips. If you need more than two 6821s, a 74LS-138 decoder is necessary, as in the 8255 card design. Figure 10b shows how this decoder is used in a design that can address up to eight MC6821s.

The National NSC810A

The NSC810A is the most complex chip I've considered so far. It has one 6-bit and two 8-bit bidirectional ports, two 16-bit multimode timer/counters, and 128 bytes of RAM. Although you can't access the RAM with this design, you can program the I/O ports for several access modes, including individual bit set and reset.

To minimize the number of pins on the

continued

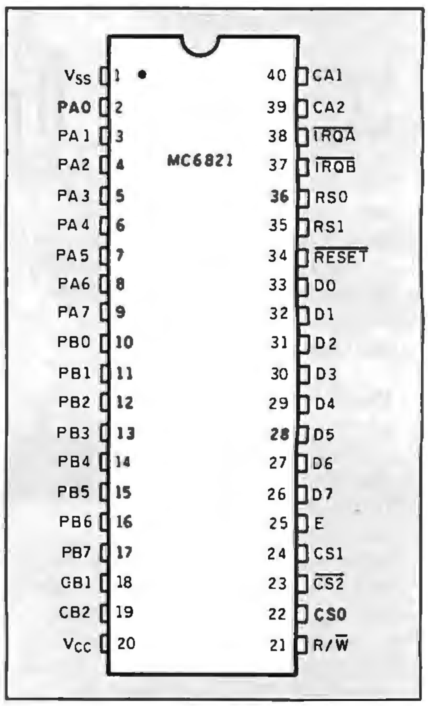


Figure 8: *Pin-outs for the Motorola MC6821.*

chip, the data bus pins and register select pins are multiplexed, keeping the pin count to 40, as shown in figure 11. An address latch enable signal (ALE) is used to latch the register address onto the chip early in the I/O cycle. The disadvantage of this technique is that an I/O cycle requires more control signals.

The NSC810A uses 23 registers, as shown in table 6. Note that 26 bytes of contiguous space are used in the I/O map, but three of the addresses are not used. The timer ports must be accessed as low-address byte, high-address byte,

continued

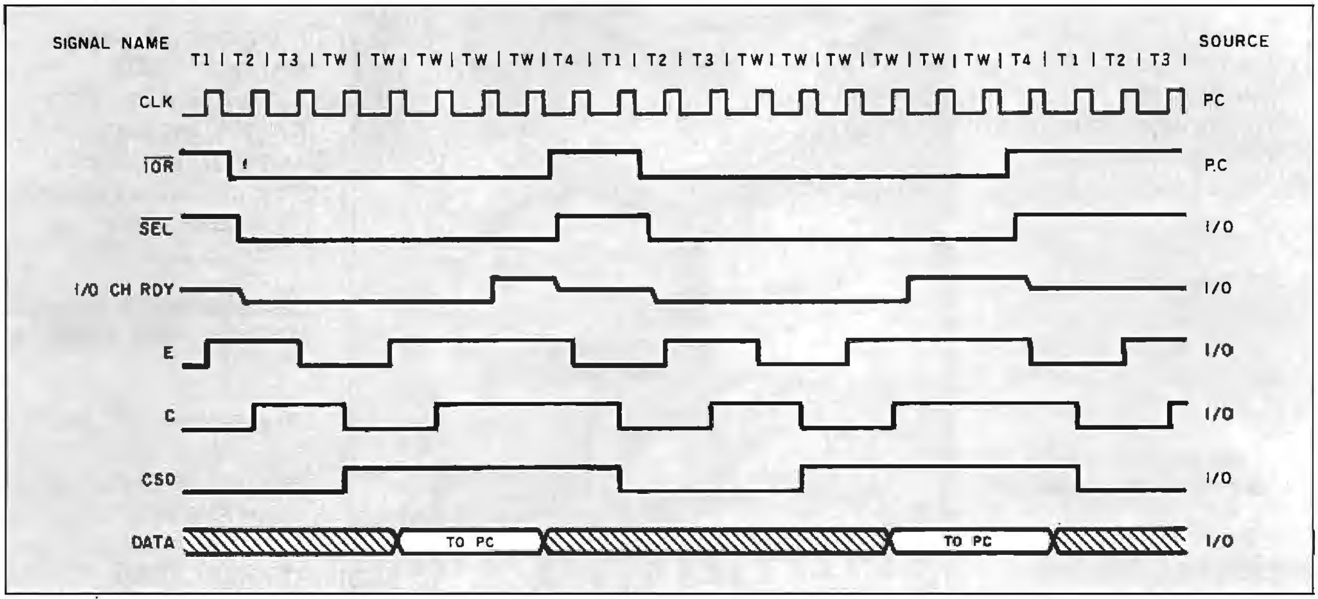


Figure 9: *Timing signals for an MC6821 word I/O read cycle. I/O CH RDY is used to synchronize the MC6821 to the PC I/O cycle. The E clock is held high to hold data on the bus until the PC finishes accessing it.*

which matches the IBM PC's word I/O operations.

The interface design uses a 16R6 PAL to decode only the 26 register locations required, leaving the upper six locations for perhaps another card with a few registers on it. The PAL sequencer allows for all features discussed above. In addition, it lets you use NSC810A chips on any IBM PC bus for both byte and word access. The timing for a read cycle is shown in figure 12.

The NSC810A uses multiplexed address and data lines, so you must allow sufficient time to clock both address and data into the chip during I/O accesses, even on an 8-MHz PC. This requires that the read/write pulses be at least three cycles long, and the ALE must be high for at least two cycles. The PAL remultiplexes the bus from the static address lines and uses the I/O CH RDY line to delay the PC while it clocks an I/O access through at a speed that the NSC810A can handle. The PAL equations are in PAL-810.LST.

The NSC810A is a CMOS chip, which reduces power consumption enormously. This is a significant advantage in multichip boards. It also means that the NSC810A requires CMOS logic levels, whereas the IBM PC bus uses TTL. More precisely, a CMOS input high level is required to be 3.5 volts or greater, while an LSTTL output at full load is required to be 2.4 V or greater. This means that all bus lines to the NSC810A must be buffered, even though the load is very low. The best buffers in this case are HCT series because they are input TTL-compatible and output CMOS-compatible. However, the drive capability of a 74HCT245 is only about 35 percent of a 74LS245. So, you take the unusual step of using two buffer chips in series: a 74LS245 off the data bus and then HCT245 to the NSC810A.

If a simple TTL output drives the NSC810A, you can use a 2.2K-ohm pull-up resistor to increase the logic high out-

continued

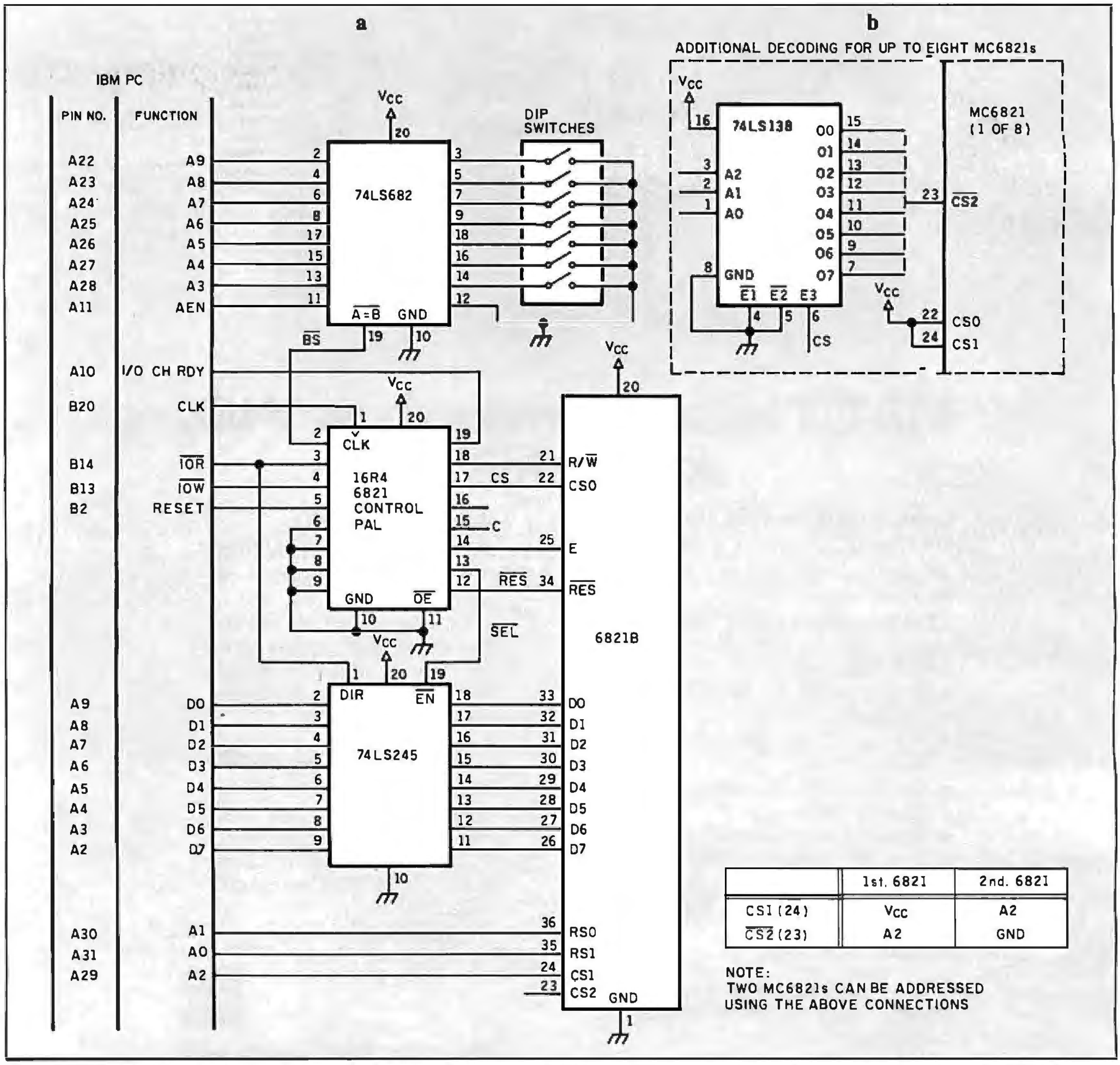
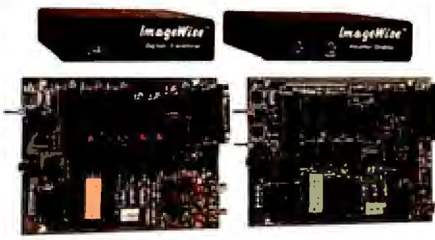


Figure 10: ***(a)*** *A circuit design that uses the MC6821 as a word I/O port. The note tells you how to connect two MC6821 chips.* ***(b)*** *Use of a 74LS138 decoder to interface up to eight MC6821 chips to the circuit.*

MICROMINT'S Gold Standard in Single Board Computers & Controllers

Announcing

ImageWise™
Serial Digital Imaging System



MICROMINT INTRODUCES A STAND-ALONE SERIAL DIGITAL IMAGING SYSTEM. The MICROMINT **ImageWise® Serial DigitalImaging System** is the most cost effective and versatile high performance grayscale video digitizing system on the market today. The ImageWise system has been designed to function as a standalone digitizer or as an integral component of a complete tele-imaging system. **ImageWise™**'s serially bit mapped digitized pictures give it almost universal compatibility with any computer capable of attaching to a modem or terminal. It is ideally suited for CAD/CAM. esktop Publishing, automatic inspection, and security applications. Critical-System functions such as image resolution and picture update can be controlled and commanded remotely. Images are transmitted and received serially, either compressed or uncompressed, and can be displayed, transmitted, stored, edited, or processed for use in a variety of industry standard application software.

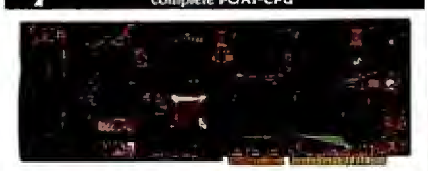
IMAGEWISE SYSTEM SPECIFICATIONS
- NOT bus dependent—can function standalone
- Truetrame grabber—uses a high speed flash A/Dconverter and 64K bytes of static RAM to capture an image in 1/60th second.
- Accepts any B/W or color NTSC video signal.
- Stores pictures as 244 lines of 256 pixels, 64 levels grayscale.
- Resolution of transmitted image is software selectable, all images are represented in 64 levels f gray scale.
  Selectable Resolutions:
  High: 256x244x6
  Med: 128x122x6
  Low: 64x61x6
- Video Input. 1 volt peak-t -peak, B&W or color 75 ohm termination
- Video Output. 75 Ohm, 1.5 V peak-to-peak NTSC composite video
- Serial Input/Output. RS-232 – 8 bit, one stop bit, no parity 300 bps– 57.6Kbps selectable data rate – Xon / Xoff Handshaking Switch selectable data compression (on/off)
- Modem compatible: Easily functions as a video telephone to send video images anywhere.
- Video processing, disk upload/download, and display utilities provided for PC-DOS and MS-DOS machines

Optional PC Utilities Disk converts ImageWise™ files for use with popular Desktop and Paint Programs.

DT01	ImageWise Digitizer / Transmitter	$349.00
DR01	ImageWise Receiver / Display	$349.00

Announcing

OEM-286 — $775.00
complete PC/AT-CPU



Low Power!
Expansion Card Form Factor!
100% AT Compatible!

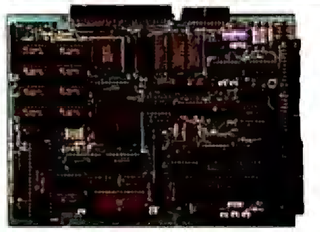
MICROMINT's **OEM-286** is a complete PC/AT-CPU and more. The **OEM-286** is the first low power, 100% AT compatible which has been specifically designed for OEM use within the industrial and business sectors. The **OEM-286** features the Zymos CMOS POACH set and 100% compatible Award BIOS. The development of the POACH chip set has allowed the 129 IC's on a standard AT to be reduced to 23 IC's and two SIMMS. What this means for you is:
1) The overall size of a standard AT/CPU has been condensed into the expansion card form factor (13 ¼" x 4 ⅞").
2) Power requirements are less then 1 A @ 5 volts
3) **OEM-286** plugs into a passive backplane for easy connection to other expansion peripherals

The **OEM-286** is available in both 8 and 10 Mhz versions and comes with the Award BIOS.

OEM-286 FEATURES
- 100% AT compatible
- 80286 Microprocessor. 8 or 10Mhz
- 80287 o-Processor opti nal
- 64 Kbytes of ROM, can accommodate 128 Kbytes
- 512 Kbytes of RAM
- Keyboard Controller
- Expansion card sizefactor
- Standard interface to the System Expansion Bus
- battery backed real time clock
- Award BIOS inclu ed

OEM-286/8	8 MHz AT/CPU	$775.00
OEM-286/10	10 MHz AT/CPU	$825.00
OEM-286/BP	8 SLOT PASSIVE BACKPLANE	[illegible]

SB180FX — $409.00
Single Board Computer



SB180FX TECHNICAL SPECIFICATIONS

PROCESSOR
- Hitachi HD64180, an 8-bit CPU in a 68 pin PLCC package
- Superset of Z-80 instruction set, including hardware multiply
- Integrated Memory Management Unit with 512K bytes address space
- Dynamic RAM refresh
- Wait state generator
- Clocked serial I/O port
- 2 channel Direct Memory Access Controller
- 2 channel Asynchronous Serial Communication Interface
- 2 channel 16-bit Programmable Reload Timer
- 12 Interrupts
- Dual bus interface to 68xx and 80xx support chips
- 6.144MHz, 9.216MHz, and 12.288 MHz system operation

MEMORY
- 512K bytes dynamic RAM on board
- Memory externally expandable to 4 Mbyte RAM
- Either an 8K 2764, 16K 27128, or 32K 27256 EPROM usable
- Full function 8K ROM resident monitor

INPUT/OUTPUT
- Console RS-232 serial port with auto-baud rate select to 38,400 baud
- Peripheral RS-232 serial port, full handshaking, 150-38,400 baud
- Line printer parallel I/O port
- 24 bits bi-directional parallel I/O
- 19-bit address decoding, I/O port decoding, and dual bus interface brought out to expansion bus connector
- Can be directly attached to GT180 640 x 480 color graphics adapter
- Fully implemented SCSI hard disk and communications bus interface

FLOPPY/HARD DISK INTERFACE
- Uses Standard Microsystems 9266 disk controller
- Compatible with NEC 765A controller
- On-chip digital data separator
- Can control 3½", 5¼", and 8" floppy disk drives—up to 4 in any combination
- Handles both FM encoded (single density) and MFM encoded (double density) data
- NCR 53C80 SCSI bus controller for hard disk or network communication

SOFTWARE COMPATIBILITY
- CP/M, ZRDOS, ZCPR3 Compatible

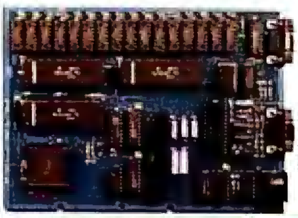
SB180FX-1	SB180FX - 6.144 MHz computer board populated w/ 256K bytes RAM, 8K byte ROM monitor, without SCSI chip. Add $50.00 for 9 MHz.	[illegible]
SB180FX-1-30	SB180FX-1 computer board as described above with Z-System software including ZRDOS, ZCPR3, editor, utilities, ZAS assembler, and ZDM debugger, BIOS and ROM monitor sources, and BIOS for SCSI hard disk. Supplied on five 5¼" SB180 format DSDD disks.	[illegible]

SB180 — $299.00
Single Board Computer

SB180-1	SB180 - 6.144 MHz single board computer w/256K bytes RAM and ROM monitor. Add $50.00 for 9 MHz	$299.00
SB180-1-20	Same as above w/ZCPR3, ZRDOS BIOS and ROM sources	$399.00
COMM180-S	SCSI Hard Disk Interface	$150.00

SB180-1 OEM 100 QUANTITY PRICE $195.00

GT180 — $395.00
Graphics Display System



The GT180 has both TTL RGB (red-green-blue intensity) and analog RGB outputs, is totally compatible with virtually all standard and multiscan CRT monitors, and has a connector for an IBM PC keyboard.

The GT180 offers these features:
- Advanced graphics controller provides intelligent link between computer and user
- Only 5.75" x 8", piggy-backs on either an SB180 or SB180FX computer
- High resolution at a low cost: 640X 480 with 16 of 4096 colors
- Hardware drawing commands: LINE, RECTANGLE, POLYLINE, POLYGON, CIRCLE, ELLIPSE, ARC, FILLED RECTANGLE, PAINT, PATTERN, WINDOW, and COPY to name but a few
- Automatic translation of logical X-Y coordinates to physical frame buffer addresses
- Fast drawing speed of 2 million pixels per second
- Provides fully programmable horizontal split screens and window screen
- Fully supported by GT180 Graphix Toolbox written in Modula-2

GT180-1	Graphic Display Expansion Board (TTL RGB only)	$395.00
GT180-2	Graphic Display Expansion Board (TTL RGB and Analog RGB)	$449.00

SB180
Software and Accessories

SB180-U	Uniform Disk Format Conversion Software	$ 69.95
SB180-ZMSG/TKBBS	Z-MSG Bulletin Board Software	$100.00
SB180-MOD2	Turbo Modula 2	$ 69.00
SB180-MOD2G	Turbo Modula 2 w/Graphix Toolbox	$ 89.00
SB180-CASE	Four half height 5¼" drive enclosure w/power supply, mounting brackets and hardware for the SB180 or SB180FX	$197.00
SB180-CABLE	Set of 4 cables including power, term, disk and print	$ 79.00

BCC22 — $249.00
Term-Mite Smart Terminal

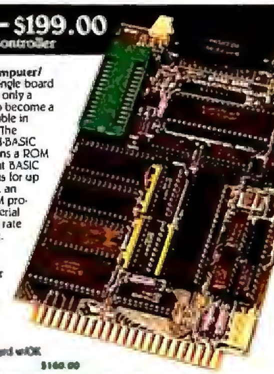
Why pay $500 or more for a smart terminal? The **TERM-MITE ST** offers you all of the following on a single board for less than ½ the price

- Dimensions: 4" x 6½"
- 128 displayable characters
- 24 lines x 80 characters
- Separate transmit & receiver baud rates (110-19,200bps)
- CRT refresh at 50 - 60 Hz
- Supports scanned and encoded keyboards
- 11 Graphic characters
- 25th line reverse video status display
- 21 escape functions
- 14 control functions
- Directly drives composite video or separated sync monitor
- All functions are firmware controlled. Source code available

EDITING FEATURES: typeover, clear to screen to space or null, erase to end of page, erase to end of line, absolute cursor addressing
VIDEO ATTRIBUTES: reverse video, half intensity, double height, double width, underlined, blinking and blank characters
BUS CONFIGURATION: MICROMINT BCC compatible or no bus connection neccessary for stand-alone operation with parallel keyboard.

BCC22	TERM-MITE Smart Terminal Board	$249.00
BCC22K	PARALLEL Encoded ASCII KEYBOARD plugs directly into TERM MITE	$ 79.00

BCC52 — $199.00
BASIC 52 Computer/Controller



The MICROMINT **BCC52 Computer/Controller** is a stand-alone single board microcomputer which needs only a power supply and terminal to become a complete system programmable in BASIC or machine language. The **BCC52** uses the Intel 8052AH-BASIC microprocessor which contains a ROM resident 8K byte floating point BASIC interpreter. It contains sockets for up to 48K bytes of RAM/EPROM, an "intelligent" 2764/128 EPROM programmer, 3 parallel ports, a serial terminal port with auto baud rate selection, a serial printer port.

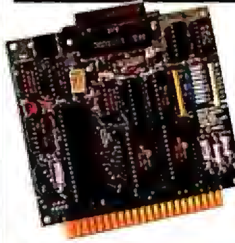
BCC52*
Basic-52 Computer/Controller
$199.00

OEM 100 QUANTITY PRICE
$149.00

BCC53
Multi Function Expansion Board w/0K adds 6 ports and 62K bytes $160.00

* NOW AVAILABLE IN INDUSTRIAL TEMPERATURE RANGE – CALL FOR INFORMATION

BCC11 — $139.00
Basic Controller



Features:
- Uses Z8 single chip microcomputer
- On board tiny basic interpreter
- 2 on board parallel ports & serial port
- 6 interrupts (4 external)
- Just connect a CRT and write control programs in Basic
- 6K Bytes of RAM or EPROM memory on board
- Baud rates 110-9600 bps
- ata and address bus available for 56K memory and I/O expansion.
- Consumes only 1.5 watts at +5, +12 & −12V.

BCC11*
BASIC System Controller
$139.00

* NOW AVAILABLE IN INDUSTRIAL TEMPERATURE RANGE – CALL FOR INFORMATION

BCC40 — $159.00
Power I/O Expansion Board

The MICROMINT **BCC40 POWER I/O Expansion Board** provides on/off control and monitoring of eight 115-230 VAC or 5-48VDC devices. Up to 16 POWER I/O boards may be used in a system for a total of 64 inputs and 64 outputs

BCC40
POWER I/O Expansion Board (with no power modules) $159.00
with 8 Output Modules $249.99
with 8 Input Modules $229.99

BCC52 & BCC11
Software and Accessories

BCC52-ROM A	ROM A Utilities-BASIC extensions	$ 49.95
BCC52-ROM A/B	ROM A and B Utilities BASIC extensions and Assembler	$100.00
BCC52/41	ROM C Utilities-Real Time Clock and Power I/O firmware	$ 34.00
BCC52-0K-CLK	SMARTIME BCC52 Clock and ROM C	$ 69.00
BCC52-8K-CLK	SMARTIME BCC52 Clock w/8K RAM and ROM C	$ 79.00
BCC53	Multi Function Exp. Board w/0K-adds 6 ports and 62K Byte	$160.00
BCC08	Serial I/O Expansion Board	$149.00
BCC09	RS-232 to 20ma Converter	$ 55.00
BCC13	8 Bit, 8 channel A/D Board	$129.00
BCC30	12 Bit, 16 channel A/D Board	$197.00
BCC25-4	4x20 LCD Display	$159.00
BCC25-8	8x40 LCD Display	$229.00
BCC22K	Parallel Enclosed ASCII keyboard	$ 79.00

MOTHERBOARDS—CARD GAGES—POWER SUPPLIES

MB02	5 Slot mini-mother board (not compatible with CC01, CC02)	$ 69.00
MB08	8 Slot full mother board (compatible with CC01/CC02	$ 85.00
MB44	44 pin gold card edge connector	$ 6.00
CC01	10 inch Card Cage for one MB08	$ 59.00
CC02	19 inch Card Cage for two MB08s	$ 79.00
UPS11	Universal Power Supply-used for single board applications .9A @ 5V, .3A @ +12V and .1A @ −5V	$ 19.00
UPS05	Card Cage Power Supply-designed for a 2 or 3 board system 1A @ 5V, .3A @ +12V, .1A @ −12V.	$ 79.00
UPS10	Heavy Duty Switching Power Supply—used for an entire BCC system 5.5A @ 5V, .7A @ +12V, .3A @ −12V	$ 69.00
UPS21	21 Volt Programming Power Supply—used for EPROM programming on the BCC52 21V +/− 2% @ 200ma	$ 49.00

To order call
1-800-635-3355
for technical information 1-(203)-871-6170
TELEX: 643331

Micromint, Inc.
4 Park Street
Vernon, CT 06066

MICRO MINT

Table 6: *The register map of the National NSC810A. The timer ports must be accessed low byte first, followed by the high byte. This matches the order for access for a PC word I/O cycle.*

Bits						
A4	**A3**	**A2**	**A1**	**A0**	**Register**	**Access**
0	0	0	0	0	Port A (data)	R/W
0	0	0	0	1	Port B (data)	R/W
0	0	0	1	0	Port C (data)	R/W
0	0	0	1	1	Not used	—
0	0	1	0	0	DDR - port A	W
0	0	1	0	1	DDR - port B	W
0	0	1	1	0	DDR - port C	W
0	0	1	1	1	Mode-definition register	W
0	1	0	0	0	Port A - bit clear	W
0	1	0	0	1	Port B - bit clear	W
0	1	0	1	0	Port C - bit clear	W
0	1	0	1	1	Not used	—
0	1	1	0	0	Port A - bit set	W
0	1	1	0	1	Port B - bit set	W
0	1	1	1	0	Port C - bit set	W
0	1	1	1	1	Not used	—
1	0	0	0	0	Timer 0 (LB)	*
1	0	0	0	1	Timer 0 (HB)	*
1	0	0	1	0	Timer 1 (LB)	*
1	0	0	1	1	Timer 1 (HB)	*
1	0	1	0	0	STOP timer 0	W
1	0	1	0	1	START timer 0	W
1	0	1	1	0	STOP timer 1	W
1	0	1	1	1	START timer 1	W
1	1	0	0	0	Timer 0 mode	R/W
1	1	0	0	1	Timer 1 mode	R/W

* Write to modulus register, read from timer.

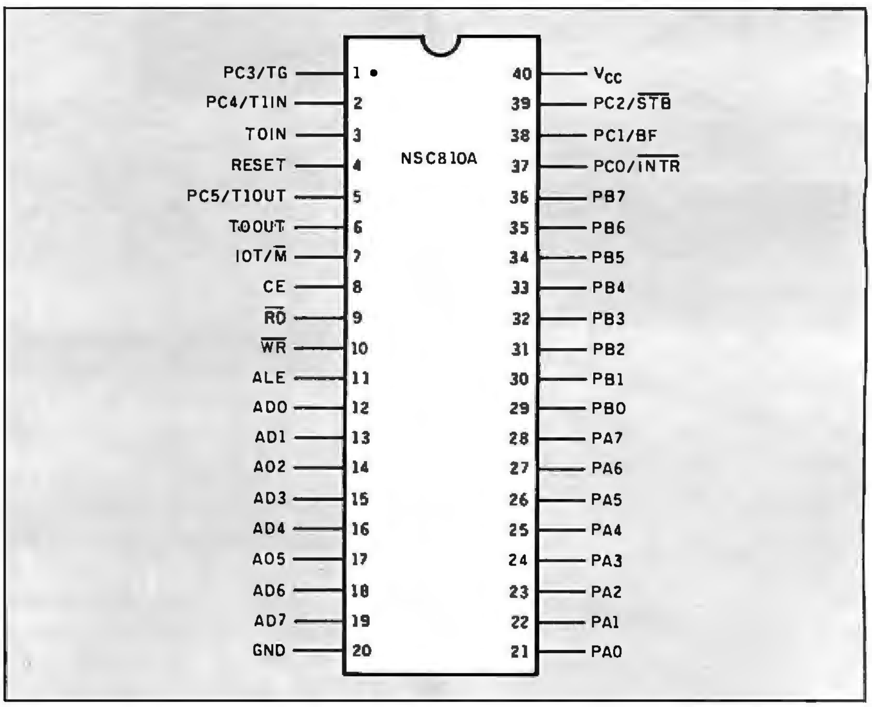


Figure 11: *The pin-outs for the NSC810A.*

American Semiconductor
Gives you the prices and products that have quickly made them,
TOP GUN
SAY CHARGE IT! W/AMERICAN SEMICONDUCTOR CHARGE CARD
ANTHONY
MICROSOFT Flight Simulator
Complete systems for Less than ANYBODY!
And we can FINANCE YOU!
Our revolving charge makes owning the system you want easy!
Over a $BILLION of FINANCING AVAILABLE!!
NOT A LEASE ★ YOU OWN IT!
★FREE!! FLIGHT SIMULATOR LIMITED TIME OFFER!!
WITH ANY COMPUTER PURCHASE
$29.mo.
TURBO XT MODEL #5301
• Selectable 4.77 & 8 MHz Processing Speed • 64K RAM Memory • Bios • One 360K Floppy Drive • Floppy Controller w/Cable • Eight Expansion Slots • PC/XT Compatible Keyboard
RETAIL VALUE 1099.
OUR PRICE 499.
$39.mo.
TURBO XT MODEL #5302
• Selectable 4.77 & 8 MHz Processing Speed • 640KRAM Memory on Main Board • Bios • 16 Bit 8088-2 CPU • One 360K Floppy Drive • Multi I/O (w/Floppy Controller, Clock/Calendar, Parallel, Serial, & Game Port) • Eight Expansion Slots • PC/XT Compatible Keyboard
RETAIL VALUE 1599.
OUR PRICE 739.
$45.mo.
TURBO XT MODEL #5303
• Selectable 4.77 & 8 MHz Processing Speed • 640K RAM Memory on Main Board • Bios • 8088-2 CPU • One 360K Floppy Drive • Multi I/O (w/Floppy Controller Clock/Calendar, Parallel, Serial & Game Port) • Eight Expansion Slots • One 10MB Hard Drive (w/Controller) • PC/XT Compatible Keyboard
RETAIL VALUE 1999.
OUR PRICE 1019.
$49.mo.
TURBO XT MODEL #5304
• 8 MHz Processing Speed • 640K RAM Memory on Main Board • Bios • 16 Bit 8088-2 CPU • Two 360K Half/Height Floppy Drives • Multi I/O (w/ Floppy Controller, Clock/Calendar, Parallel, Serial, & Game Port) • One 20MB Hard Drive (w/Controller) • TURBO Normal Mode Either Software or Hardware Selectable • Eight Expansion Slots • PC/XT Compatible Keyboard
RETAIL VALUE 2499.
OUR PRICE 1295.
$49.mo.
TURBO XT MODEL #5305
• 8 MHz Clock Speed • 640K RAM Memory on Main Board • Bios • 16 Bit 8088-2 CPU • Two 360K Half/Height Floppy Drives • Multi I O (w/Floppy Controller, Clock/Calendar, Parallel, Serial, & Game Port) • 30MB Hard Drive (w/controller) • TURBO Normal Mode Either Software or Hardware Selectable • Eight Expansion Slots • PC/AT Compatible Keyboard
RETAIL VALUE 2999.
OUR PRICE 1479.
EGA
$59.mo.
TURBO XT MODEL #5306
• 8 MHz Processing Speed • 640K RAM Memory on Main Board • Bios • 16 Bit 8088-2 CPU • Two 360K Half/Height Floppy Drives • Multi I/O (w/ Floppy Controller, Clock/Calendar, Parallel, Serial, & Game Port) • 20MB Hard Drive (w/Controller) • TURBO Normal Mode Either Software or Hardware Selectable • Eight Expansion Slots • PC/XT Compatible Keyboard • IBM Compatible Enhanced Graphics Adapter (EGA) • High Resolution EGA Monitor
RETAIL VALUE 3899.
OUR PRICE 1949.
$5.mo.
VIDEO GRAPHIC PACKAGES
MONOCHROME
COLOR
EGA ENHANCED
WITH MOST SYSTEM PURCHASES
HARD DRIVE KITS from 10 mb. to 360 mb.
NEW 386
$85.mo.
MODEL #5701
• 16 MHz Processor Speed • Intel 80386 Micro Processor • 32 Bit architecture • Compatible with 8 MHz 80286 Hardware & Software • Phoenix 386 Bios • Four Speed Selection (16, 8, 6, 4.77 MHz) • Relocation of Bios • EGA Bios to 32 Bit High Speed RAM for Lightning Execution • Real Time Clock • 4, 8 or 10 MHz 80287 Co-processor Socket • Standard 32 Bit High Speed Memory Board 1MB of RAM Memory Installed, Connectors For Attaching Up to 10MB of Memory • 1.2 MB Half/Height Floppy Drive • Eight Expansion Slots • Serial/Parallel Adapter • AT Compatible Keyboard
RETAIL VALUE 8999.
OUR PRICE 3272.
$59.mo.
AT PORTABLE TURBO MODEL #5509
XT640 RAM Memory • AT 512 RAM Memory • 1/2 High 360 Floppydisk drive • 8088 CPU/80286 CPU • RAM 640K • Operating system: CP/M86 & UCSD-P • Interface cards: multi I/O card, color graphics card or monochrome card • 9" Dual frequency monitor or monochrome monitor • Power: 135W power supply (for XT) 200W power supply (for AT) • AT/XT switchable keyboard (Cherry Switch) • Size: 48 cm (W) x 44 cm (D) x 19 cm (H) • Weight: 14 kgs
RETAIL VALUE 2995.
OUR PRICE 1749.
AVAILABLE IN XT • MODEL #5309 • $49 mo.
RETAIL VALUE 2450 • OUR PRICE 1249.
$69.mo.
TURBO AT MODEL #5503
• 10MHz Clock Speed • 512K RAM Memory (1MB Option Available) • Bios • Intel 80286 Microprocessor • 1.2MB Half/Height Floppy Drive • 40MB Hard Drive (w/Controller) • Eight Expansion Slots • Serial/Parallel Adapter • AT Compatible Keyboard • Performance 16, 12, 10, 8, 8 MHz Available.
RETAIL VALUE 5999.
OUR PRICE 2099.
$59.mo.
BABY AT TURBO MODEL #5507
• 6-8-10 MHz Processing Speed • 8 MHz • 640K (Expandable to 1MB on Main Board) • Bios • 80286 Based CPU • 360K Half/Height Floppy Drive • Hard Drive (w/Controller) • Multi IO (w/Floppy Controller, Clock/Calendar, Parallel, Serial, & Game Port) • 20MB Hard Drive (w/Controller) • AT Style Keyboard
RETAIL VALUE 3999.
OUR PRICE 1759.
$59.mo.
TURBO AT MODEL #5501
• 512K RAM Memory (Expandable to 1MB on Main Board) • 80286 Based CPU • Bios • 1.2MB Half/Height Floppy Drive • 20MB Hard Drive (w/Controller) • Eight Expansion Slots • Parallel Port • AT Style Keyboard • Performance 16, 12, 10, 8, 6 MHz Available.
RETAIL VALUE 3999.
OUR PRICE 1719.
$69.mo.
TURBO AT MODEL #5502
• 10MHz Clock Speed • 512K RAM Memory (1MB Option Available) • Bios • Intel 80286 Microprocessor • 1.2MB Half/Height Floppy Drive • 30MB Hard Drive (w/Controller) • Eight Expansion Slots • Serial/Parallel Adapter • AT Compatible Keyboard • Performance 16, 12, 10, 8, 6 MHz Available.
RETAIL VALUE 4499.
OUR PRICE 1959.
CHEAPER PRICES? AMERICAN SEMICONDUCTOR had to make a choice between building our products as cheaply as possible and selling it as a "get by" product, or building quality into our products for service, durability, and your long lasting enjoyment.
NO OTHER COMPANY CAN OFFER YOU A 5 YEAR WARRANTY ON ITS XT & XT TURBO MOTHER BOARDS AND INSTANT CREDIT BESIDES!
THAT'S WHY WE SELL SO MANY!
INSTANT CREDIT NOW!
President, American Semiconductor, Inc.
Anthony R. E. Hall
Even if you have a computer... We'll sell you add-ons from our TOP QUALITY products, and FINANCE them too!
MONITORS ARE ADDITIONAL. SURGE PROTECTOR POWER DISTRIBUTION CENTERS ARE OPTIONAL AND AVAILABLE FOR ONLY $39.95
SAY CHARGE IT!
ADD 3.95%
ORDER TOLL FREE!
1-800-237-5758 EXT 823
CUSTOMER SERVICE (813) 961-5584
IMPORT EXPORT
(813) 961-9444
OPEN MON–FRI 8 AM–7 PM
OPEN SATURDAY 9 AM–5 PM (EST)
DEALERS! call for the up to minute prices for DRAM'S, 8087's, EXPANSION CARDS, IC'S and much, much more!
1-800-825-SAVE
7283
NORTHEAST REGIONAL SALES OFFICE 1-301-933-3523
WASHINGTON, D.C. AREA
1-800-942-9488
OFFICES NOW OPEN: TAMPA – MIAMI – LOS ANGELES – WASHINGTON, D.C. – TAIPEI, TAIWAN – TOKYO, JAPAN
CORPORATE HEADQUARTERS: 16520 N. FLORIDA AVE., LUTZ, FLORIDA 33549, (813) 961-9444
PRICES SUBJECT TO CHANGE DUE TO AVAILABILITY AND MARKET FLUCTUATIONS. VOID WHERE PROHIBITED. $25.00 CREDIT APPLICATION FEE.

put voltage level. This technique is used for the PAL output lines, as shown in figure 13. The chip select of an NSC810A is active high, so a 74HCT238 is used to decode the chip selects from address lines A10 through A12.

A New Degree of Freedom

I hope that this article has given you enough details to show you that interfacing to the IBM PC bus is not difficult, even when several manufacturers' products are involved. Although many companies produce quality peripheral cards for the PC, Murphy's Law decrees that these cards never do precisely what you want. Conversely, you might find a peripheral card that has just the features you need, but it might also be loaded with features

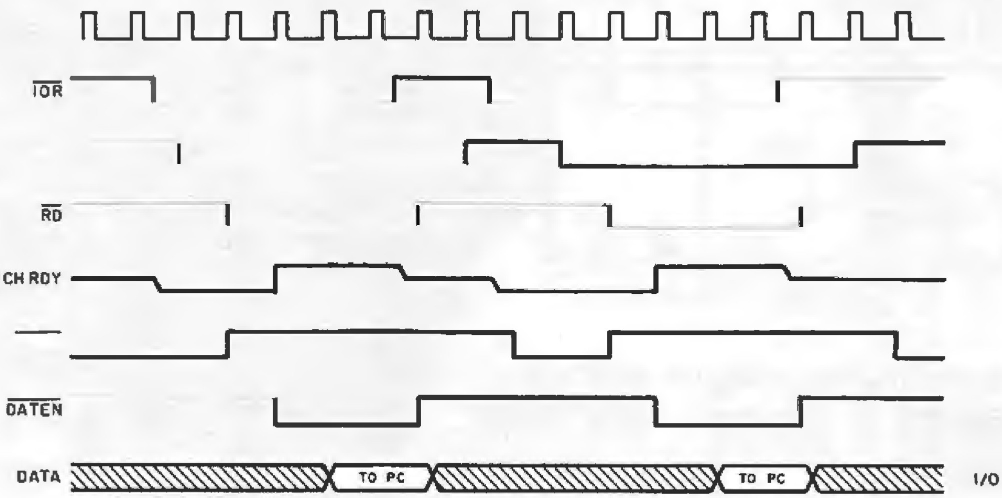


Figure 12: *Timing signals for an NSC810A word I/O cycle. I/O CH RDY is used to extend the PC I/O cycles.*

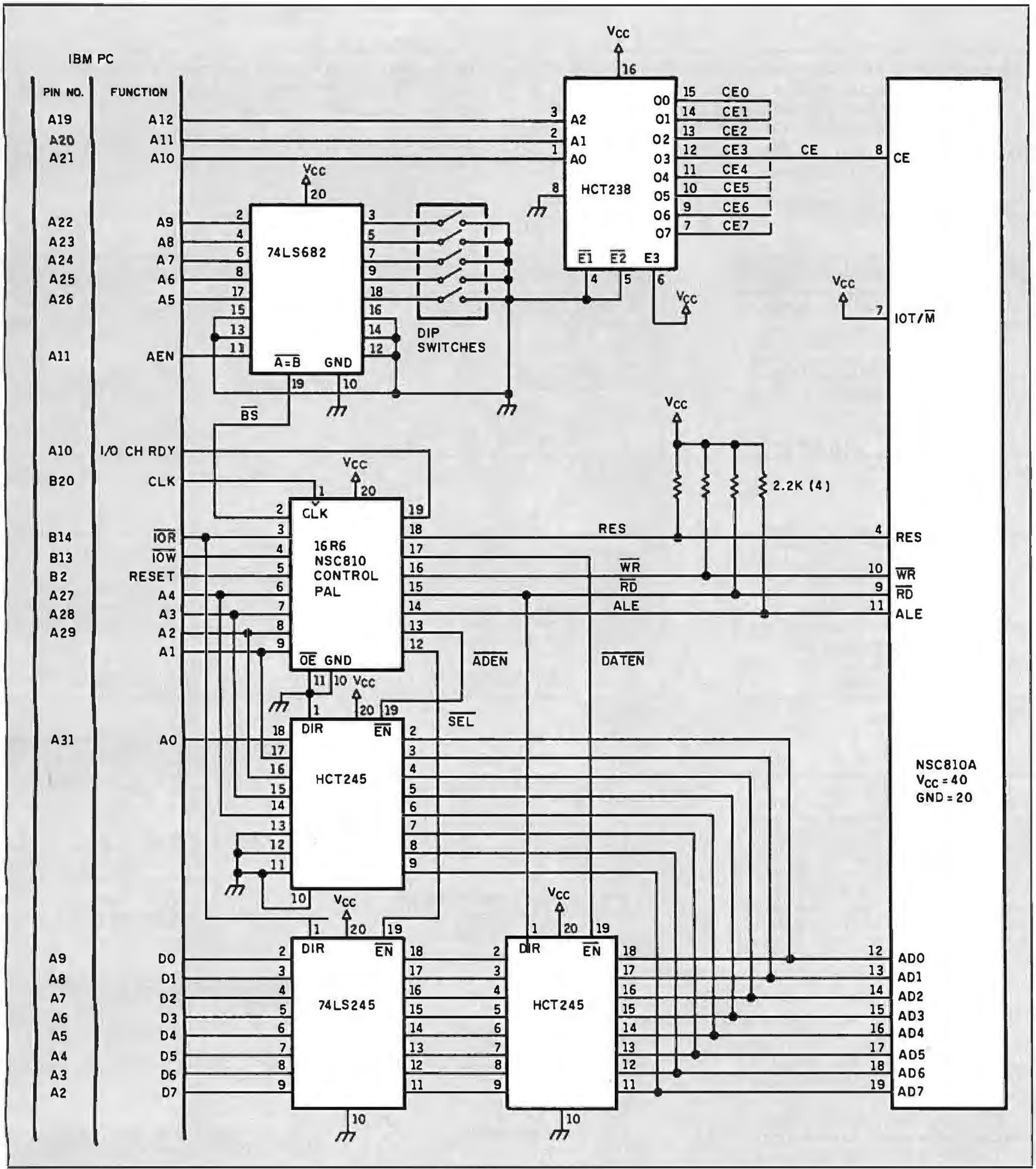


Figure 13: *The major differences in this interface are the use of HCT parts to get the proper interface voltages and the necessity of multiplexing the address and data lines. The PAL outputs are pulled up by resistors to ensure that the logic high output voltage is adequate for the NSC810A.*

you don't need and don't want to pay for.

With the information presented here, you now have the freedom to build your own peripheral card if a vendor's card won't do the job. You will also be able to produce your own interface cards if you are trying to build specialized computer-controlled equipment.

If you don't have access to a PAL programmer, don't despair. Certain companies will custom-program PALs for a small fee if you provide the listings. They will often give you a price reduction if the equations are on floppy disk, so it makes sense to make use of the BYTE disk service first.

It's possible that this article will inspire you to build the PAL programmer project in the January issue of BYTE ("A PAL Programmer" by Robert A. Freedman). ■

THE BUYER'S MART
A Directory of Products and Services

THE BUYER'S MART is a monthly advertising section which enables readers to easily locate suppliers by product category. As a unique feature, each BUYER'S MART ad includes a Reader Service number to assist interested readers in requesting information from participating advertisers.

RATES: 1x—$475 3x—$450 6x—$425 12x—$375
Prepayment must accompany each insertion. VISA/MC Accepted.

AD FORMAT: Each ad will be designed and typeset by BYTE. Advertisers must furnish typewritten copy. Ads can include headline (23 characters maximum), descriptive text (250 characters maximum), plus company name, address and telephone number. Do not send logos or camera-ready artwork.

DEADLINE: Ad copy is due 2 months prior to issue date. For example: October issue closes on August 1. Send your copy and payment to THE BUYER'S MART, BYTE magazine, 1 Phoenix Mill Lane, Peterborough, NH 03458. For more information call Mark Stone at BYTE 603-924-3754.

ACCESSORIES

SOFTWARE PACKAGING, DISKS
Cloth binders & slips like IBMs. Vinyl binders, boxes, and folders—many sizes. Disk pages, envelopes & labels. Low qty. imprinting. Bulk & branded disks. Much More! Low prices. Fast service. Call or write for FREE CATALOG.
Anthropomorphic Systems Limited
376-B East St. Charles Road
Lombard, IL 60148
1-800-DEAL-NOW (312) 629-5160

Inquiry 551.

COMPUTER PROTECTION
• UPS • LINE CONDITIONERS • ISOLATORS • MODEM PROTECTORS
• AC POWER INTERRUPTERS
— HUNDREDS OF HINTS & PRODUCTS —
FREE CATALOG 1-800-225-4876
ELECTRONIC SPECIALISTS, INC.
171 So. Main St., POB 389, Natick, Mass 01760
1-800-225-4876

Inquiry 552.

FREE CATALOG
Diskettes & Acc • Dust Covers • Furniture • Data Cartridges • Plotter Pens • Clean'g Supl • Tapes & Acc • Ribbons • Printers • Computer Paper • Copier Supplies • Surge Prot • Cables & Acc • Print Wheels • MUCH MORE!
Call, write or circle inquiry card for a FREE Catalog
GAAN COMPUTER SUPPLIES
186 B East Sunnyoaks, Campbell, CA 95008
(800) 523-1238, In Calif. (408) 370-6747

Inquiry 553.

COMPUTER VOLUME CONTROL
At last. . .no more annoying BEEPS! Quiets games tool Adjust volume anytime from a faint whisper to maximum. Installs easily, directly to speaker wires with solderless terminals.
Only $29.95 ppd or 2 for $49.95.
VISA/MC/Check or M.O. AZ add 6.7% tax.
INNOVATIVE COMPUTER SOLUTIONS, INC.
2929 N 44th St., #400, Phoenix, AZ 85018
(602) 946-4743

Inquiry 554.

DIABLO 630 SHEET FEEDER
AUTOMATIC 2-TRAY
MINT — ONLY $400!!
Prints all first pages on letterhead and all other pages on blanks automatically. Feeds thousands of consecutive pages without a misfeed. Printers also available.
Quality Discount Computers
135 Artistview Drive, Wellington, NV 89444
(702) 465-2473 Ext. 120

Inquiry 555.

BAR CODE

BAR CODE MADE EASY
PERCON® E-Z-READER™ keyboard interfaces and multiuser RS-232 models make it easy to add bar code to virtually any computer/terminal WITHOUT SOFTWARE MODIFICATION. Immediate shipping. Two year warranty. Bar code printing software available. Call for details on fast, accurate, easy data entry. Substantial reseller discounts.
PERCON®
2190 W. 11th St., Eugene, OR 97402
(503) 344-1189

Inquiry 556.

READ & PRINT BAR CODES
Internal unit (short-card for IBM PC/XT/AT/compatibles) or RS-232 Dual Port unit: **both work with all keyboards and software.** Includes stainless steel wand or LASER gun. User-friendly Bar Code and Text printing software with format/size flexibility.
Seagull Scientific Systems
601 University Ave., Suite 150, Sacramento, CA 95825
(916) 386-1776

Inquiry 557.

BAR CODE READERS
Among the industries best and most widely used barcode reader, reads all major barcode formats (code 39, 2 of 5, UPC/EAN, codabar), connects between keyboard & system, advanced CMOS uses keyboard power supply, connects to all IBM compatibles and DIN terminals, completely OS independent, software independent. Same day ship, 1 year warranty, 30 day satisfaction guarantee. CALL for prices too low to advertise.
Solutions Engineering Sales
8653 Georgia Ave., Silver Spring, MD 20910
800-635-6533

Inquiry 558.

DATA INPUT DEVICES
Bar Code & Magnetic Stripe Readers for microcomputers & terminals, including IBM PS/2 & others, DEC, AT&T, CT, Wyse, Wang. All readers connect on the keyboard cable & are transparent to all software. Low cost bar code print programs & magnetic encoders are also available. GSA approved.
TPS Electronics
4047 Transport, Palo Alto, CA 94303
415-856-6833 Telex 371-9097 TPS PLA

Inquiry 559.

PRINT BAR CODES & BIG TEXT
Print Labels with Large Text and/or bar codes on PC with Epson/Oki/IBM LaserJet printers. Text to 1"x1" reads at 50 ft. Bar Codes: Code 39, 2 of 5, UPC/EAN, MIL-STD, AIAG. Flexible format/size, Reversals. File Input, Menu-driven $279, Other menu-driven programs for bar codes $49-$129.
Worthington Data Solutions
417-A Ingalls St., Santa Cruz, CA 95060
(408) 458-9938

BAR CODE

BAR CODE READERS
PC/XT/AT Bar Code Reader attaches as 2nd keyboard, reads bar codes as keyed data. One model converts to external mounting via velcro or slot mounting in PC — $385. RS-232 Model—$399. Rugged metal wand, reads UPC, EAN, 2 of 5, Code 39, etc. Same day ship. 1 year warranty. 30 day $$ back guarantee.
Worthington Data Solutions
417-A Ingalls St., Santa Cruz, CA 95060
(408) 458-9938

BAR CODES/BIG TEXT FROM YOUR PROGRAM
Add bar codes and/or big graphics text to your program. Print from inside of dBASE, BASIC, C, any MS-DOS language or even a word processor. Bar codes: UPC-A, UPC-E, 2 of 5, and Code 39. Big text in wide and narrow sizes .5", .3" and .2" tall. Prints on Epson/Oki/IBM LaserJet printers $179-$239.
Worthington Data Solutions
417-A Ingalls St., Santa Cruz, CA 95060
(408) 458-9938

BOOKS/DISKS/VIDEOS

A Print-It-Yourself Book
You may read on screen as easily as you print out the 333 pages. Educational, entertaining with a big message as far out as the doomsday scenario & the sci-fi means of survivor education. $18 for the 3 disk set for PCs, packaged with free PC-Write 2.71 complete with spelling checker.
DIA Inc.
334 Forepeak Avenue, Beachwood, NJ 08722
(201) 341-8113

Inquiry 560.

CAD/CAM

CadON™
Add on to AutoCAD for schematic entry and PC layout. Features extensive user expandable libraries, netlist extract, bill of materials, ratsnesting. Does 100% check of layout against schematic. Unlimited trace widths, pad sizes, etc.
RUBOW SYSTEMS
19102 Bridwell St., Glendora, CA 91740
(818) 914-3963

Inquiry 561.

COMMUNICATIONS

BBS ALTERNATIVE
The Business Connection—closely-coupled commware. Host reacts only to matching remote nodes. Secure, private mail/messaging. Off-line msg creation on remotes. User-friendly, auto call/logon, auto msg interchange, help, more. For IBM & compats, DOS 3.X, 256K, Hayes-type modem. Shareware rel—$25. Complete multi-function ver including printed manual—$99.
SYNERGETIC INNOVATIONS
P.O. Box 79, Arroyo Grande, CA 93420

Inquiry 562.

THE BUYER'S MART

COMPUTER INSURANCE

COMPUTERS + INSURANCE = SAFEWARE
Call toll free for information
- LEASED COMPUTERS
- HOME COMPUTERS
- OVERSEAS COMPUTERS
- BUSINESS COMPUTERS
- COMPUTERS LEASED TO OTHERS
- COMPUTERS TAKEN TO SHOWS
- COMPUTERS HELD FOR SALE
- OTHERS' COMPUTERS IN YOUR CARE

SAFEWARE
The Insurance Agency Inc.
1-800-848-3469 IN OHIO (614) 262-0559

Inquiry 563.

CROSS ASSEMBLERS

Disassemble MPU OBJECT CODE on your PC
Relocatable, symbolic disassemblers are now available for the Motorola, Intel, RCA, TI, Rockwell, & Zilog micros! Automatic label generation, assembly capability and much more. Call and ask for what you need.

RELMS™
P.O. Box 6719
San Jose, California 95150
(408) 265-5411
TWX 910-379-0014 (800) 448-4880

Inquiry 564.

DATA CABLES & PARTS

Custom Cables for LAN's
Ethernet—Token Ring, RG62, 59, 58, Twinax, Dual Wang, Telco, Mod adapters, RJ11, 12, 45, octopus cables, Harmonicas, V.35, M50, IEEE-488, RS232, 422, 449, parallel, ribbon cable, switches, Baluns, surge sentry, connector parts, bulk cable, tools

Communication Cable Company
P.O. Box 600, Wayne, PA 19087 • 215-644-1900
Send for Catalog

Inquiry 565.

DATA CONVERSION

DATA TRANSLATION
Tape/Disk • Disk/Disk • OCR
Transfer data to and from hundreds of formats, data types, and word processors. Quick turnaround! Call today for a quotation to fill your special need.
- Reformat databases to your specifications
- Translate and print mailing lists
- Quality word processor to word processor conversions
- Authorized KEYWORD Service Bureau

DataCopy Service of Texas
4010 Cornell, Garland, TX 75042 (214) 272-7751

Inquiry 566.

DATA/DISK CONVERSION

Disk/Disk • Tape/Disk • OCR
Over 1,000 formats! 3½, 5¼, or 8 inch disks; 9 track mag tape; 10 MB Bernoulli cartridge. Data base and word processor translation. Specialists in Government Sensitive Data. Call for free consultation..

Computer Conversions, Inc.
We take the hassle out of data conversion ...
9580 Black Mtn. Rd., Ste J, San Diego, CA 92126
(619) 693-1697

Inquiry 567.

DISK/DISK or TAPE/DISK CONVERSIONS — AUTOMATICALLY
Bought a new computer? Convert your data in just 1-2 days.
SAVE TIME - SAVE MONEY!
DISK/DISK or TAPE/DISK Conversions start as low as $15.00 for IBM, CP/M, MS-DOS, UNIXV, XENIX & most other systems. (Apple, Macintosh, Commodore conversions start at $25.00). Call or write today for a cost saving quotation to fill your disk-to-disk or tape-to-disk data conversion needs.

CREATIVE DATA SERVICES
1210 W. Latimer Ave., Campbell, CA 95008
(408) 866-6080

Inquiry 568.

DATA/DISK CONVERSION

DISK AND TAPE CONVERSIONS
High quality conversion services & OCR scanning for Dedicated Word Processors, Mini and Micro computers. Over 1000 3½", 5¼", and 8" formats, 800-1600BPI tape. Conversion between Wang, NBI, CPT, DEC, Vydec, Lanier, OS/6, Xerox, Linolex, Lexitron, MemoryWriter, Editwriter, CompEdit, Exxon 500, Exxon Qyx, IBM Sys/34/36/38/5520, MAC, Victor, TRS, Apple II & III, NSTAR, IBM PC/AT/3½, HP, and most other Micros. Conversion directly into word processing software such as DW3, WP, MS/WRD, WS, Samna, MM, PFS, & many others.
DATA FORMATS, INC. (408) 629-1088

Inquiry 569.

CONVERSION SERVICES
Convert any 9 track magnetic tape to or from over 800 formats including 3½", 5¼", 8" disk formats & word processors. Disk to disk conversions also available. Call for more info. Introducing OCR Scanning Services.

Pivar Computing Services, Inc.
165 Arlington Hgts. Rd., Dept. #B
Buffalo Grove, IL 60089 (312) 459-6010

Inquiry 570.

dBASE III COMPILERS

FREE dBASE III +
COMPILER DEMO DISK
Don't buy Clipper™ until you see our FREE dBASE Compiler Evaluation Kit. Includes DEMO DISK with 8 PRG's and results of compiling with Clipper, Quicksilver & FoxBASE +. Also 15 benchmark tests, complete magazine reviews, and detailed brochures about all 3 compilers. FREE. No obligation. Call 24 hours, 7 days.

dataBASE Specialties **(415) 652-2790**
P.O. Box 2975, Oakland, CA 94618

Inquiry 571.

DEMOS/TUTORIALS

INSTANT REPLAY II
Build Demos, Tutorials, Prototypes, Presentations, Music, Timed Keyboard Macros, and Menu Systems. Includes Screen Maker, Keystroke/Time Editor, Program Memorizer, and Animator. Rec'd *Great* Reviews! Simply the BEST. Not copy protected. No royalties. 60 day satisfaction money back guar. IBM and Compatb. $149.95 U.S.Chk/Cr. Crd. Demo Diskette $5.00

NOSTRADAMUS, INC.
3191 South Valley Street (ste 252)
Salt Lake City, Utah 84109 (801) 487-9662

Inquiry 572.

DESKTOP PUBLISHING

Professional Type Composition
MP-XL—Hard-core typography and Typefonts for the HP LaserJet II, Okidata LaserLine 6, or Ricoh PC 6000 Laser Printers. H&J, Indents, Tabs, Graphics, ¼ pt. leading, random mix of size & face, 6–72 pt. fonts. MP-XL complete with 22 fonts, Pi + Bkmn 6–24 pt. .... $195.
Many additional type faces available 6–30 pt.

Micro Print-X, Inc.
P.O. Box 581, Ballinger, TX 76821
(915) 365-2343

Inquiry 573.

DOCUMENTATION

COMPUTER ASSEMBLY MANUALS
BIG BLUE SEED for IBM™ BUILDERS: Parts list, placement diagrams, instructions for assembling 75 bare cards. Now includes 640K, Turbo & AT MthBds. $17.95.
APPLE™ BUILDERS NEED APPLE SEED: Instructions for assembling 85 Apple-compatible bare cards including the popular 6502 II+ and IIe MthBds. $14.95.
Both for $30. Bare cards available. Check/MO/VISA/MC:

NuScope Associates*, Dept B
P.O. Box 790 • Lewiston New York • 14092
*A Division of Kosmic MicroTech Inc.

Inquiry 574.

ENTERTAINMENT

NFL CHALLENGE UPDATE
PROROSTR ™ increases the realism of your NFL Challenge game by providing you with an easy way to update player data files. Prints a formatted printout of the player names and data. Includes PRODATA™ the complete 1987 player roster and statistics. Requires IBM type NFL CHALLENGE (copyright XOR Corp.). PROROSTR $32 ppd. PRODATA may be purchased separately $22 ppd. Check, MO, MC, VISA

HD&S Software
Box 924183, Houston TX 77292
(713) 686-9102

Inquiry 575.

FLOW CHARTS

FLOW CHARTING II+ HELPS YOU!
Precise flowcharting is fast and simple with **Flow Charting II+**. Draw, edit and print perfect charts: bold and normal fonts, 24 shapes—95 sizes; fast entry of arrows, bypasses & connectors; Fast Insert Line; shrink screen displays 200-column chart; 40 column edit screen for detail work, much more!

PATTON & PATTON
81 Great Oaks Blvd., San Jose, CA 95119
1-800/672-3470, ext. 897 (CA residents)
1-800/538-8157, ext. 897 (Outside CA)
408/629-5044 (Outside the U.S.A.)

Inquiry 576.

HARDWARE

ANDY'S COMPUTERS
P.O. Box 6096, Austin, TX 78762
1 (800) 521-2283 for ordering
1 (512) 478-9803 for info
Prices include Amber Monitor (TTL), 640K M/B, 256K Installed Case, Power Supply, Keyboard, Printer, Port, Disk
Turbo XT 10 MZ. . . . . $500. AT 286 Mini 6-12 . . . . $964
AT 286 6-10MZ . . . . . $958 386 16MZ. $2775
Hard Disks add: 20 MEG $330 Portable XT 10 MZ . . . $759
Portable AT. . . . . $1190 Portable 386 16 MZ. $3200
Other equipment and configurations available at similar savings.

Inquiry 577.

DISCOUNT CLONES
- *XT Turbo — **$450.***
- *AT 8/10 MHz — **$1200.***
- *Seagate ST225 20MB hard drive with controller — **$300.***
- *Seagate ST251 40MB 38ms — **$495.***
- *Hayes compatible 1200 baud modem — **$99.***
- *NEC multisync — **$550.***
- *Panasonic KX/P 10091 — **$225.***

Automated Business Solutions
5 Broadway, 2nd Fl., Freeport, NY 11520—516-379-3995
30 day money back guarantee • 1 yr parts & labor warrantee
Dealer Inquiries Invited.

Inquiry 578.

CHIP CHECKER
- 74/54 TTL + CMOS
- 8000 Nat. + Signetics
- 14/4000 CMOS
- 9000 TTL
- 14-24 Pin Chips
- .3" + .6" IC widths

Tests/Identifies over 650 digital chips with ANY type of output in seconds. Also tests popular RAM chips. IBM compatible version $259. C128 + C64 version $159.

DUNE SYSTEMS
2603 Willa Dr., St. Joseph, MI 49085
(616) 983-2352

Inquiry 579.

HIGH PERFORMANCE LOW COST
SINGLE BOARD COMPUTER has optimum features for Monitor + Control Applications: 16 Chan A/D with Sample + Hold • 2 RS232/422 Ports • Buffered Digital I/O • Timer • 5.25 x 8.0. OPTIONS: Resident **FORTH 79 OS** with Target Compiler, Editor, Assembler, RamDisk + Auto Load/Start; 5 MHz 8085 • 10 bit A/D • Bat. Backed Clock/RAM • Networking Expansion Boards • IBM/CPM Support.
The NEW E-PAC 1000+ $249.00

EMAC INC.
PO BOX 2 42 CARBONDALE IL. 62902 618-529-4525

Inquiry 580.

THE BUYER'S MART

HARDWARE

Save on brownout protection!

Line Conditioner keeps power constant whenever AC input power varies up or down! Prevents damage and downtime! Maintains constant output of 120V. This is a stepped transformer system that has higher efficiency than CVT's and gives lower waveform distortion. Built-in spike protection!

INDUS-TOOL
730 W. Lake St., Chicago, IL 60606
Phone 312-648-2191

Inquiry 581.

HARDWARE/ADD-ONS

640K MotherBoard UPGRADE

640K memory *without using an expansion slot!* Our upgrades allow easy insertion of 256K chips on the system board. 100% compatible. 30 day $ back guarantee. 2 yr wrnty. *Compaq* Portable, *Zenith* 150, 151 & 161: $29 ($94 with memory). *IBM-XT*, 3270 & Portable: $39 ($104 with memory). *IBM-PC*: $49 ($114 with memory). $4 s/h. Site discount. Dealers welcome.

Aristo
16811 El Camino #213-D, Houston, TX 77058
713/480-6288, 800/3ARISTO.

Inquiry 582.

Motherboard Clock $59

SideClock does not waste an expansion slot! The Inventions' Clock/Calendar, installs in seconds: Just open the computer cover and snap it into place. Never enter the time and date again, SideClock does it for you. Software and user replaceable battery included. 30 day $$ back gtd. 2 year limited warranty. $59 + $3 s/h. Site discounts. Dealers welcome.

Aristo
16811 El Camino, #213-D, Houston, TX 77058
713/480-6288, 800/3ARISTO.

Inquiry 583.

HARDWARE/COPROCESSOR

PC MINI-SUPERCOMPUTER
Up to 40 MIPS In Your PC!

Fill your PC/XT/AT with 1 to 6 PC4000 boards for a high speed PC-RISC system. The PC4000 uses the NC4016 RISC Engine which executes high level Forth in silicon. Each PC4000 is a general purpose parallel coprocessor that delivers speeds in the 5 to 7 MIPS range. . . over a 100 times faster than a PC, K & R standard C and Forth available. From $1295.

SILICON COMPOSERS (415) 322-8763
210 California Ave., Suite I, Palo Alto, CA 94306

Inquiry 584.

HARDWARE/MONITORS

COLOR MONITORS
SURPLUS SALE

NEW, Hitachi 13" RGB,
640 x 240, .31 mask pitch, IBM Compatible, 50 & 60 Hz
Their Price: $450 Our Price: $150

International Materials Exchange Inc.

IME
33 Agassiz St.,
Cambridge, MA 02140
617-497-2290

Inquiry 585.

HARDWARE/PERIPHERALS

GM CAR COMPUTER READ-out

- Read engine computer data
  - Is sensor data reasonable?
  - Are status switches set?
  - Are any trouble codes set?
- Make disk records of car packets
- Let your PC be automotive.

EXECUTIVE ELECTRONICS, INC.
938 Main St., Dept. B., Yarmouthport, MA 02675
(617) 362-3694

Inquiry 586.

HARDW./POWER PROTECTION

POWERLINE GREMLINS??
POWER FAILURES??

The MEIRICK STANDBY POWER SYSTEM
is the TOTAL SOLUTION to your
powerline problems.
400 watt system - $449;
800 watt system - $795

MEIRICK Inc., POWER SYSTEMS DIV.
Box 298, Frisco, CO 80443 **303-668-3251**
1-800-323-5911

Inquiry 587.

INVESTMENT

***CAPTOOL!*™ $59**

Multiple portfolio manager. Stocks, bonds, funds, dpr. assets. True ROI before/after taxes. DJ/Compuserve pricing. Batch & client features. Stock & bond evaluators. Beats all others! IBM PC or PS/2. $3 S&H. Chk/VISA/MC.

TECHSERVE, inc.
P.O. Box 70056, Bellevue, WA 98007
800-826-8082 or 206-747-5598

Inquiry 588.

LAPTOP COMPUTERS

Toshiba 1100+ 20 meg. Hard Drive

A full featured External Hard Drive that attaches easily to your Toshiba. This COMPACT Drive has a Modem slot of its own—So you can have your Hard Drive and Modem too. . . Dos 2.1 or 3.2 compatible. Drive comes with 12 Volt A.C. Adapter. Optional Battery pack and Cigarette Lighter Cord available. Retails for $1095.00 Dealer Inquiries Welcome. NEC Drive Available soon. IN STOCK

801-364-8238
Computer Expansion Products, Inc.
3596 So. 300 W. #7, Salt Lake City, Utah 84115

Inquiry 589.

LAPTOPS & ACCESSORIES
FOR THE SMART BUYERS

• Zenith 181 & 183 • Toshiba • NEC Multispeed • DATAVUE • EXTERNAL 5¼" or 3½" Drive units (in ½ ht. case w/power & cable) for PC/XT/AT & compatibles • Brooklyn Bridge • Modems • 3½" int. drives

Call for low pricing

Computer Options Unlimited
201-469-7678 (7 Days, 9AM-9PM Eastern time)

Inquiry 590.

MANUFACTURING

THE FOREMAN

- Inventory • B of M • Purchasing
- Invoicing/Production/Job Cost/MRP
- Interface to Solomon Accounting
- One time machine builds/Produc ion
- Extensive cost control: Standard vs Actual
- Release type orders + PO's accepted
- Single user or Multi user versions available

DCM COMPUTER PRODUCTS, INC.
29226 Orchard Lake Rd., Suite 290, Farmington Hills, MI 48018
313-737-0808

Inquiry 591.

OKIDATA—ACCESSORIES

An Uki for Your Oki!

Compatibility of IBM Plug-n-Play & versatility of Okidata Instruction Set at flick of switch. Easy to install. Complete—no other ROM needed. Utility software included. Specify printer model. Add $3 ship. NY res add sales tax. Oki 83, 92, 93—$49.95. Oki 192,193—$79.95.

Uki Switch Corp.
Dept. B, 326 Linden Place, West Hempstead, NY 11552
212-685-8199

Inquiry 592.

PROGRAMMER'S TOOLS

Modula-2

REPERTOIRE is an integrated DBMS, window/forms/menu generator, multi-window text editor, and expression evaluator designed specifically for M2. Only $89 with full (600K) source code and 320p printed manual. Complete manual for this and many other M2 tools available on free demo disk. MC/VISA/AMEX/PO/COD.

PMI 4536 SE 50th, Portland, OR 97206
(503) 777-8844; BIX: pmi

Inquiry 593.

★ ★ NETBIOS Routines ★ ★

Network Master is a library of 80 routines which access NETBIOS functions. The routines are invoked by compiled programs to control the network. Rou ines support network names, resources and sessions with Wait and No-Wait options. Create your own transaction processing and program communication systems. Many compilers are supported. No royalties. Visa/MC $199.00

Starlight Software
2861 Central St., Evanston, IL 60201
★ (312) 864-9370 ★

Inquiry 594.

PUBLIC DOMAIN

$3 SOFTWARE FOR IBM PC

Public Domain & User Supported Software. Hundreds to choose from, wordprocessors, data bases, spreadsheets, assorted games for all ages, communications, business, music, art, programming language and useful utilitites for making your computer easier to learn. Most programs have documentation on the d sk. Write for your FREE catalog today!

BEST BITS & BYTES
P.O. Box 5332, Dept-B, No th Hollywood, CA 91616
(818) 893-6304

Inquiry 595.

QUALITY I.B.M. SHAREWARE
$3.95/Disk — 10 for $30.00

P.C Write 2.7, DOS utilities, commercial unlock, 8088 assembler DOS tutor, BASIC tutor, print utilities, games (7 disks), many more. Free list! MC/Visa.

C. D. I.
1-800-537-8000 ext. 25
Box 238, 3925 Ravenswood Rd., Ft. Lauderdale, FL 33312

Inquiry 596.

RENT SOFTWARE $1/DISK

Rent Public Domain and User Supported Software for $1 per diskfull or we'll copy. IBM, Apple, C-64, Sanyo 550 and Mac. Sampler $5. VISA/MC. 24 hr. info/order line. (619) 941-3244 or send #10 SASE (specify computer) Money Back Guarantee!

FutureSystems
Box 3040 (T), Vista, CA 92083
office: 10-6 PST Mon.-Sat. (619) 941-9761

Inquiry 597.

★ ★ FREE Catalog ★ ★

Collection of over 1700 programs collected on 180 diskettes, each diskette costs $3.75. Wide variety of programming, including:

• Games	• Word Processing	*And*
• Personal Finance	• Lotus Clones	*Much*
• CAD Systems	• Database Systems	*More!!!*

Call 1-800-843-5084
People's Choice Software
P.O. Box 3061, Warner Robins, GA 31099

Inquiry 598.

THE BUYER'S MART

SALES/MARKETING TOOLS

SALES MAGIC
Software solutions for better selling from
MARKET POWER INC.
computer Innovations

We are proud to have helped these companies: Navistar, Westinghouse, Kodak, Chemical Bank, Chrysler, Canadian National Hotels, and more...

Whether you are a company or an individual, we can help you. Call or write. **(916) 432-1200**

101 Providence Mine Rd., Suite 106 A
Nevada City • California 95959

FREE DEMO DISK

Inquiry 599.

SOFTWARE/1-2-3 COMPILER

THE BALER ***$495***

Convert worksheets into stand-alone applications. No Royalties. No Programming Required. 1-2-3 Work-Alike. 1A & 2.01 Commands, Functions & Macros. Formulas & Structure Are Tamperproof. Generates Customizable BASIC Source Code. 30 Day Money Back Guarantee.

Brubaker Software
Supporting 1-2-3 since 1983
8825 N. County Ln. Rd. E., Lafayette, IN 47905
800-327-6108 **317-564-2584**

Inquiry 600.

SOFTWARE/ACCOUNTING

ACCOUNTING KIT $79.95

Peanuts & Caviar is a kit designed to introduce the principals of accounting and bookkeeping to non-accountants. It includes a resetable general ledger software package ree. The kit consists of a 300 page book, worksheets, homework, homework analysis and GL software. The software generates detailed reports, t al balance, income statement and balance sheet. IBM or compatible. Used in curriculums by Platt College, Denver Public Schools, etc. $79.95.

Learn-Ed Software
8400 E. ILIFF, Suite 11, Denver CO 80231
1-800-621-8385 ext. 736

Inquiry 601.

SOFTWARE/A.I.

Neural Net Model

SIMNET s/w shows how a neural net model may be instructed to learn specific binary pattern associations. Once sys. has been taught the user may test its capability for generating proper output patterns in response to random input sequences. Also allows adjustments of learning rate. $59.95 U.S. money order, cash or certified check. Needs IBM PC 256K, one floppy.

Blair House Innovations
P.O. Box 7, Belcarra Park, Port Moody, BC V3H3EI Ca ada
604-939-5998

Inquiry 602.

SOFTWARE/BUSINESS

MILP88-MIXED INTEGER LP

A general-purpose system for solving mixed integer linear programs with up to 65 integers. 255 constraints 1255 variables. Solves problems by applying the branch and bound method. Menu-driven with many useful features similar to LP88. Req. IBM PC, 192K, $149 with 8087 support, user's guide. VISA/MC.

EASTERN SOFTWARE PRODUCTS, INC.
POB 15328, Alexandria, VA 22309
(703) 549-5469

Inquiry 603.

dFELLER Inventory

Business inventory programs written in modifiable dBASE source code.

dFELLER Inventory $150.00
Requires dBASE II or III, PC-DOS/CPM
dFELLER Plus $200.00
with History and Purchase Orders
Requires dBASE III or dBASE III Plus (For Stockrooms)

Feller Associates
550 CR PPA, Route 3, Ishpeming, MI 49849
(906) 486-6024

Inquiry 604.

SOFTWARE/CHURCH

SCRIPTURAL ROSARY PROGRAM

Computerized version of an ancient prayer provides convenient, interesting means of praying the rosary with accompanying scriptural meditations for each rosary bead. *Excellent for introducing children to the rosary.* For IBM PC XT, AT and compatibles. $11.95. Parochial school discounts available.

Catholic Software Systems
2207 Birch St. SE, Decatur, Alabama 35601
(205-350-0789)

Inquiry 605.

PowerChurch Plus®

Fast, friendly, reliable church administration system. Full fund accounting, mailing lists, membership, contributions, attendance, word processing, accts. payable, payroll, multi-user support, and much more - all for $495 complete. FREE Demo version.

F1 SOFTWARE
P.O. Box 3096, Beverly Hills, CA 90212
(213) 854-0865

Inquiry 606.

Church Package

Parishioner Time, Talent and Treasure System program is written in modifiable dBASE source code.

- Contributions • Disbursements • Ledger
- Names with mailing labels
- **Personal Information database.**

Requires dBASE II or III. PC-DOS/CPM-80 $200.

Feller Associates
550 CR PPA, Route 3, Ishpeming, MI 49849
(906) 486-6024

Inquiry 607.

ROMAR CHURCH SYSTEMS™

Membership-61 fields plus alternate addresses; labels, letters, reports any field(s). Offering-256 funds; optional pledge, statements; post to 255x/yr. Finance-gen. ledger w/budget, up to 500 subtotals & 99 depts., month & YTD reports anytime for any month. Attendance—8 service times, 250 events per service, 60 consecutive weeks. Available for floppy, 3½ & hard disk. Ad too short! Write for free 48-page guide.

Romar Church Systems, Attn: BJB
P.O. Box 4211, Elkhart, IN 46514
(219) 262-2188

Inquiry 608.

SOFTWARE/CULINARY

The Professional Recipe Manager

For IBM PCs 256K MS or PC/DOS. Combines recipe sizing, costing & nutritional computation $249.95 + $3 s/h check/AMEX/MC/VISA. (Mass add 5%). RECIPE MANAGER also available. $49.95.

GEM ISLAND SOFTWARE
P.O. Box 393B, Reading, MA 01867
617/944-1382

Inquiry 609.

SOFTWARE/EDUCATION

THE GMAT COURSE (PC/XT/AT)

Covers every type of exam question with instant feedback, solutions, timed tests, personalized score analyses, math and grammar reviews. All material is on the screen. Designed by MBA's, the program has improved students' scores by as much as 250 pts. Includes 5 disks + manual. $59.95

COMWELL SYSTEMS, INC.
P.O. Box 41852, Phoenix, AZ 85080
(info & orders) **(602) 869-0412**
(orders only) **(800) 255-2789**

Inquiry 610.

SOFTWARE/ENGINEERING

Engineer's Aide

Join the Desktop Engineering Revolution!

- Pipeline/Ductwork Sizing
- Orifice/Control Valve Sizing
- Pump/Fan/Compr. Sizing
- Project Financial Analysis
- Heat Exchanger Sizing
- Conversion Calculator
- Fluid Properties Library
- Specification Writer

Above programs in one stand alone integrated package for $695. For IBM PC & Macintosh.

Engineering Programming Concepts
1-800-367-3585 (24 hr)1-805-484-5381 (Ca)

Inquiry 611.

SIMULATION WITH GPSS/PC™

GPSS/PC™ is an IBM personal computer implementation of the popular mainframe simulation language GPSS. Graphics, animation and an extremely interactive environment allow a totally new view of your simulations. Simulate complex real-world systems with the most interactive and visual yet economical simulation software.

MINUTEMAN Software
P.O. Box 171/Y, Stow, Massachusetts, U.S.A.
(617) 897-5662 ext. 540 (800) 223-1430 ext. 540

Inquiry 612.

ENGINEERING SOFTWARE

You will save hours of development time with our library of integrated programs.

- Graph Plotting
- Equation Processing
- Simultaneous Equations
- Complex Arithmetic
- Pop Up Calculator
- Fourier Transforms
- Top Down Designer
- and more . . .

For a FREE CATALOG: write

Pulse Research
Dept Eng 5, P.O. Box 696, Shelburne, VT 05482
(802) 985-2928

Inquiry 613.

SOFTWARE/FORTRAN

FORTRAN 77 Extension

EXTEND subroutines for MS, RM, IBM Pro compilers to control keyboard, monitor, text & graphics, DOS file & directory operations, parallel & serial I/O. Plus user graphics for CGA, EGA, VGA, HP7475A, TEK 4010, AutoCAD DXF & db save files. Both 8087 & non 8087 libraries pkgd for $149. Royalty free.

Design Decisions, Inc.
P.O. Box 12884, Pittsburgh, PA 15241
(412) 941-4525

Inquiry 614.

SOFTWARE/GAME

SMART STUD POKER

Play Five- or Seven-card Stud Poker against two computer simulated players whose characters can be picked from five different character sets. You choose the ante, bankroll etc. Runs on IBM DOS 2.0, color monitor. $15 if ordered before Dec. 31; regular $25. Send money order or check to:

Kar Yung Software
P.O. Box 14074, Torrance, CA 90503

Inquiry 615.

PC HYPER-CALCULATOR

Pop-up MS-DOS emulatiom of HP-11C scientific and HP-12C financial calculators. Two-way transfer, program & data files, alpha prompts, hex/oct/bin arithmetic, 100 registers, 1000 program steps. Free 8087 version, utility programs. **$49.95 + $3 s&h.**

SUNDERLAND SOFTWARE ASSOCIATES
P.O. BOX 7000-64
REDONDO BEACH, CA 90277
CALL TOLL FREE **800-628-2828 ext. 502**

Inquiry 616.

THE BUYER'S MART

THE BUYER'S MART

SOFTWARE/REAL ESTATE

**** Real Estate Mortgage Program ****

For the professional. Qualify clients for home purchases and mortgages. Print detailed report for your client. Set default values from menu. Store client's data to disk file. Load data from disk for updates. From income program calculates, max debt, taxes, insurance, purchase price, monthly payment, mortgage amount, and more. All inputs can be edited or re-calculated. To order send $49.95 + $2 s/h. Other programs available. HOME MORTGAGE PROGRAM For the infrequent user $24.95 + $2 s/h. MASTER DIRECTORY Generates list of programs on disk $15.95 + $2 s/h.

THE SOFTWARE HUT
P.O. Box 613, Bantam, CT 06750 (203) 567-4992

Inquiry 635.

SOFTWARE/SCANNERS

Optical Character Recognition

Stop retyping: PC-OCR™ software will convert typed or printed pages into editable text files for your word processor. Works with HP ScanJet, Canon and most other scanners. Supplied with 15 popular fonts. User trainable: you can teach PC-OCR™ to read virtually any typestyle, incl. foreign fonts. Proportional text, matrix printer output, Xerox copies OK. $385. Check/VISA/MC/AmExp/COD

Essex Publishing Co.
P.O. Box 391, Cedar Grove, NJ 07009
(201) 783-6940

Inquiry 636.

SOFTWARE/SCIENTIFIC

Data Acquisition & Analysis on PC's

- FOURIER PERSPECTIVE II Advanced Digital Signal Analysis.
- PRIME FACTOR FFT subroutine library. Call from Turbo Pascal®, C, Fortran, Basic. Up to 65,520 data points. 2D interface available. Rectangular FFT's now possible in a multitude of dimensions.
- 2 & 3D Scientific Graphic packages with plotter support from Golden Software.
- Data Acquisition & Control Boards form MetraByte, Analog Devices, Burr-Brown, Strawberry Tree, National Instruments, Contec, & Tecmar.
- Data Acquisition & Analysis Software—Labtech Notebook, Asyst, UnkelScope, Lotus Measure with 1-2-3/Symphony, & Quinn-Curtis.

Call for FREE Application Assistance & Technical Literature
LOW PRICES—Satisfaction GUARANTEED

Alligator Technologies — (714) 722-1842
P.O. Box 11386 Costa Mesa, CA 92627

Inquiry 637.

FAST FOURIER TRANSFORMS

- ★ Subroutine for integer data: 1024 points in 109 millisec on 8 MHz AT.
- ★ Subroutine for floating-point data: (with 80287) 1024 points in 700 mS.
- ★ Call from MS FORTRAN, C, QuickBASIC, assembler, Turbo Pascal. Great docs.
- ★ Check or PO: $175 ea/ $275 for both.

J. W. Hartwell & Associates
Rte. 4, Box 1540, Hillsborough, NC 27278
(919) 732-7951

Inquiry 638.

Do You Need to do This on Your PC?

614 1/8	cu in
/	
1.2	liter per sec
= 8.388421	sec

Lascaux1000, *"The Intelligent Calculator"*.
(For IBM PC's) Verify logic of calculations. Reduce steps by 50%. Extensive table of physical constants. Easily customized. Much more! **$59.95** Demo disk $1.

Lascaux Graphics 3220 Steuben Ave.
Bronx, NY 10467 (212) 654-7429

Inquiry 639.

ORDINARY/PARTIAL DIFFERENTIAL EQN SOLVER

FOR THE IBM PC & COMPATIBLES

MICROCOMPATIBLES INC.
301 Prelude Dr., Silver Spring, MD 20901
(301) 593-0683

Inquiry 640.

SOFTWARE/SCIENTIFIC

TURN WordPerfect™ INTO A SCIENTIFIC WORD PROCESSOR ONLY $75

Display & print complete math & Greek symbol sets. Includes display chip, printer downloading program, & free custom printer drivers & macros with proof of purchase of WordPerfect. Call for flyer, list of supported hardware, & ordering info.

SCIENTIFIC WORD PROCESSING ENHANCEMENTS
8320 N. Lockwood Ave., Skokie, IL 60077
(312) 677-4270

Inquiry 641.

SOFTWARE/SORT

OPT-TECH SORT/MERGE

Extremely fast Sort/Merge/Select utility. Run as an MS-DOS command or CALL as a subroutine. Supports most languages and filetypes including Btrieve and dBASE. Unlimited filesizes, multiple keys and much more! MS-DOS **$149.** XENIX **$249.**

(702) 588-3737

Opt-Tech Data Processing
P.O. Box 678 - Zephyr Cove, NV 89448

Inquiry 642.

SOFTWARE/TOOLS

What is Turbo GhostWriter?
An Application Generator for Turbo Pascal®
Complete Database in Minutes!

Draw & paint screens with our tools. Tell Turbo GhostWriter how to retrieve your data. Screen Editor/Painter, FABS PLUS B-Tree File Manager, Multiple fields/key, Multiple keys/index, Application Generator produces Context Sensitive Help & Documentation. All for only $200! 60-day money back guarantee. MC/VISA/PO/COD-Company Check. In NC (919) 782-7703.

ASCII-Automated Software Concepts International, Inc.
3239 Mill Run, Raleigh, NC 27612-4135
Info/orders call 800-227-7681

Inquiry 643.

LANbase TOOLBOX FOR TURBO PROGRAMMERS!

Share a database written in Turbo Pascal over a Local Area Network. Must have Turbo version 3.0, DOS version 3.1, and a Local Area Network that supports the file sharing. Some procedures require Microsoft Networks to be running. LANbase ToolBOX sells for $99.95. LANbase ToolBOX PLUS is Database Toolbox compatible. This allows previous programs written using the Database Toolbox to be shared over a network. LANbase ToolBOX PLUS sells for $249.95 and includes the Database ToolBOX. VISA and MasterCard accepted.

Night Owl Computer Services
Software & Consulting
P.O. Box 138037, Toledo, OH 43613 (419) 474-9248

Inquiry 644.

SCREEN MANAGER

SAVE TIME! Powerful Screen Designer and Data Entry Manager increases your productivity! Interfaces to most languages. BASIC, C, FORTRAN, COBOL, PASCAL, PLM86, ASSEMBLER. Not a Code Generator! No Royalties. **$79.** Visa/MC

FREE DEMO DISK

The West Chester Group
P.O. Box 1304, West Chester, PA 19380
(215) 644-4206

Inquiry 645.

STATISTICS

STATA

Statistics and graphics join to make STATA the most powerful package for the **PC.** No comparable program is as **fast,** friendly, and accurate. $20 Demo. Quantity discount available. Call toll-free for more information. **AX/VISA/MC.**

1-800-STATAPC
Computing Resource Center
10801 National Boulevard, Los Angeles, CA 90064
(213) 470-4341

Inquiry 646.

STATISTICS

THE SURVEY SYSTEM

An easy-to-use package designed specifically for **questionnaire data.** Produces banner format cross tabs & related tables, statistics (incl. regression) & bar charts. Codes and reports answers to open-end questions. All reports are camera-ready for professional presentations.. CRT interviewing option.

CREATIVE RESEARCH SYSTEMS
1649 Del Oro, Dept. B, Petaluma, CA 94952
707-765-1001

Inquiry 647.

STATISTICS CATALOG!

If you need statistics for IBM PC or Apple II, call us and let our technical advisors help you find the statistics programs you need.
Write or call now to get a FREE catalog of statistics and quality control software.

HUMAN SYSTEMS DYNAMICS
9010 Reseda Blvd., Ste. 222
Northridge, CA 91324
(800) 451-3030 (818) 993-8536 (CA)

Inquiry 648.

RATS! Version 2.1

Best selling econometrics program. Over 4000 copies sold. OLS, 2SLS, logit, probit and much more! Forecasting with ARIMA, VAR. Exponential smoothing. Model simulations Support for daily/weekly data. High-quality graphics to screen, plotter, printer. $200–$300. VISA/MC. Demo available.

VAR Econometrics, Inc.
P.O. Box 1818, Evanston, IL 60204-1818
(312) 864-8772; (800) 822-8038

Inquiry 649.

StatPac Gold™

Voted World's Best Statistical & Forecasting Package in 1987 by PC World Magazine readers. Six times more votes than the next closest competitor. More comprehensive & easier to use than all others. Get the facts. Call now for your FREE brochure.

1-800-328-4907
Walonick Associates, Inc.
6500 Nicollet Ave. S., Minneapolis., MN 55423
(612) 866-9022

Inquiry 650.

Spreadsheet Statistics

NEW! 23 comprehensive statistics and forecasting modules for analyzing LOTUS worksheets. Low cost $24.95 each. Easy to use. Includes tutorial. Basic and advanced statistics with graphics and many user-definable options. Compare. Free brochure.

Walonick Associates, Inc.
6500 Nicollet Ave. S., Minneapolis., MN 55423
(612) 866-9022

Inquiry 651.

TERMINAL EMULATORS

EMU—Terminal Emulator

Emulates DATA GENERAL Dasher terminal models D200, D210, D214, D215, D220, etc. on IBM PC, XT, AT, PCjr, PS/2, and true compatibles. 50 to 19,200 baud even during file transfers. Satisfaction Guaranteed. $95 cash price. Custom emulators also available.

Rhintek, Inc.
P.O. Box 220, Columbia, MD 21045
(301) 730-2575

Inquiry 652.

THE BUYER'S MART

TESTING/DOCUMENTATION

TESTING/DOCUMENTATION
Quality presentation is a key factor in successfully marketing your products. Our professional services include:
- Test & evaluation
- Camera-ready copy
- Technical writing
- Finished manuals

Special services/fees for Shareware developers. For FREE quotes or further information, contact:
DOCUGRAPHICS
P.O. Box 6066—Suite L108, Santa Maria, CA 93456

Inquiry 653.

UTILITIES

STILL RIVER SHELL
Visual shell for PC/MS-DOS. Point & Shoot Power. V1.82. Only $9 for shareware diskette. $39 manual & diskette. MC/Visa accepted.
Bob Howard
PO Box 57, Still River, MA 01467
617-456-3699

Inquiry 654.

RECORD/PLAYBACK KEYSTROKES
TOTAL RECALL records an unlimited number of keystrokes and plays them back after a failure or mistake. Also for demos, running interactive programs in batch, or repeating keystrokes. 3 playback speeds plus ACTUAL for timing sensitive programs. Built-in edit. PC/XT/AT Compat. $79 VISA/MC 30 day guarantee.
COMPUTER FOUNDATIONS
2604 Elmwood Ave., Suite 320, Rochester, NY 14618
(716) 586-9756

Inquiry 655.

Source Level PC Debugger
Maximum performance debugger has window oriented display, complete symbolic support, easy to use command interface and powerful function keys. We challenge you to find a better debugger! $185 Call to order:
Computer Dynamics Sales
• (803) 877-8700 •
107 South Main St., Greer, SC 29651

Inquiry 656.

Screen Maker
Create text screens with extended character set in a frac ion of the time using our full screen editor. Then save them as source code in one of three languages (BASIC, C, TURBO PASCAL) or a user defined language. It's not a binary screen image. Requires 256K RAM, IBM PC or compatible. Only $39.95 + S/H. Call or send to:
DESIGN SOFTWARE
P.O. Box 1017, Jackson, NJ 08527
201-928-9654

Inquiry 657.

DATA MANIPULATION Xchange
Powerful pattern matching data manipulation language quickly solves data interchange problems. Define input/output of desired data structures. Reads serial & proprietary files including Multimate, Wordperfect, dBase, DCA/RFT, MS-Word and hundreds more. Xchange $495.00 + options.
Emulation Technologies
1501 Euclid Ave., Cleveland, OH 44115
216-241-1140

Inquiry 658.

UTILITIES

PEP Your Data to MS-DOS!
PEP ("Printer Emulation Package") is a unique software product which makes your MS-DOS system appear to be an intelligent serial printer. Converting your data from another operating system is as easy as printing it! Diskette plus 90 page typeset manual, $64.95 U.S. (Not copy protected.) Specify 5¼ or 3½ inch diskette size; add $5 for both formats.
Intelpro
13 Saratoga Dr., Kirkland, Quebec, Canada H9H 3J9
(514) 694-6862

Inquiry 659.

MS-DOS System Utilities For PCs and Z-100's
WORD Wizzard........$19. *Full Screen Editor*
FILE Wizzard............$9. *Directory Utility*
COM Wizzard......$9. *Command Line Editor*
KEY Wizzard...........$9. *Keyboard Utility*
MICRO WIZZARD SOFTWARE
199 Buckwood Dr., Hyannis, MA 02601
(617) 8-1430

Inquiry 660.

COPY AT TO PC
The 1.2mB drive has long been known to READ but NOT reliably WRITE on 360kB floppies. With "CPYAT2PC"™ 1.2mB drives CAN reliably WRITE 360kB floppies saving a slot for a second hard disk or backup tape. "CPYAT2PC" (Not Copy Protected) offers 'the preferable SOFTWARE SOLUTION.' *ONLY $79+$4* S/H VISA/MC/COD UPS B/R
MICROBRIDGE COMPUTERS
655 Skyway, San Carlos, CA 94070
Order toll free 1-800-621-0851 x777
415-593-8777 (CA) 212-334-1858 (NY)
TELEX EZLNK 62873089 Dealer Inquiries invited

Inquiry 661.

$59.95!!
Buy the RED Utilities now! Programs include: Disk cache to speed hard disk. Printer spooler. Batch file compiler. Path command for data files. Wild card exceptions. Protect hard disk from accidental formatting. Sort directories. Over 10 more programs. IBM PC. Visa/MC.
The Wenham Software Company
5 Burley St., Wenham, MA. 01984 (617) 774-7036

Inquiry 662.

WORD PROCESSING

HEBREW / GREEK / ARABIC
Russian and European Languages. Full featured, multi-language word processor supports on-screen foreign characters with no hardware modifications. $350 (dot matrix) or $500 (laser) + $5 s/h. Or send $15 + $4 s/h for demo. Req. 512K/graphics
Gamma Productions, Inc.
710 Wilshire Blvd., Suite 609, Santa Monica CA 90401
(213) 394-8622

Inquiry 663.

DuangJan 1.3
Bilingual word processor for English and one of these: Armenian, Bengali, Euro/Latin, Greek, Hindi, Khmer, Lao, Russian, Tamil, Telugu, Thai, Viet, . . . , or create your own language with font editor. $69+$4 s/h (+$10 for.) LaserJet +$49. Demo $5. IBM compatibles.
MegaChomp Company
3524 Cottman Avenue, Philadelphia, PA 19149-1606
(215) 331-2748/8138

Inquiry 664.

WORD PROCESSING

PC-Write™ Shareware Ver. 2.71
Fast, full featured word processor/text editor for IBM PC. With spell check, screen clip, mailmerge, split screen, ASCII files, macros. Easy to use. Supports 400 printers - LaserJet+ and PostScript. Software, User Guide, and Tutorial on 2 disks for $16. Try it, then register with us for only $89 and get User Manual, 1 year tele-support, newsletter and 2 upgrades. 90-day guarantee. VISA/MC.
Quicksoft 1-800-888-8088 CALL TODAY!
219 First N., #224-BYTC, Seattle, WA 98109

Inquiry 665.

Advertise **YOUR** computer products *the smart way!*

★

For as little as **$375** in **THE BUYER'S MART**

For more information and a Free Brochure Call
Mark Stone at BYTE
603-924-3754

★

BYTE Advertising
One Phoenix Mill Lane
Peterborough, NH 03458

Circle 64 on Reader Service Card

Circle 27 on Reader Service Card

Circle 76 o n Reader Service Card

Circle 171 on Reader Service Card

Circle 155 on Reader Service Card

Circle 43 on Reader Service Card

Circle 250 on Reader Service Card

Circle 8 o n Reader Service Card

Circle 140 on Reader Service Card

Circle 109 on Reader Service Card

Worldwide • Since 1974

• QUALITY COMPONENTS • COMPETITIVE PRICING
• PROMPT DELIVERY

Mail Order Electronics · Worldwide
Jameco®
ELECTRONICS

IBM COMPATIBLE COMPUTER KITS

Now You Can Build an IBM PC/XT or AT Compatible!

Jameco's new IBM PC/XT/AT compatible kits include everything you need to assemble a working computer and come with complete step-by-step assembly instructions. Plus . . . *Free!* Quicksoft PC Write Word-Processing Software.

All computer accessories below may also be purchased individually.

Please Note: JE1004 and JE1005 shipped in 2 boxes (48 lbs. total) – JE1008 shipped in 1 box (31 lbs.)

IBM PC/XT Compatible Kit

Part No.	Description	Price
4164-150	128K RAM (18 Chips)	$22.50
JE1040	Floppy Controller Card	$29.95
JE1010	Flip-Top Case	$34.95
JE1015	XT/AT Style Keyboard	$59.95
JE1030	150 Watt Power Supply	$69.95
JE1050	Mono/Graphics Card w/Printer Port	$59.95
JE1020	5¼" DSDD Disk Drive	$89.95
GREEN	12" Monochrome Green Monitor	$99.95
JE1000	XT Motherboard (Zero-K RAM – Includes Award BIOS ROM)	$89.95

SAVE $77.15 ***Regular List*** $557.10

JE1004 **IBM PC/XT Compatible Kit $479.95**

IBM PC/XT Turbo Compatible Kit

Part No.	Description	Price
4164-150	128K RAM (18 Chips) } *Total*	$ 22.50
41256-150	512K RAM (18 Chips) } *640K RAM*	$ 58.50
JE1010	Flip-Top Case	$ 34.95
JE1015	XT/AT Style Keyboard	$ 59.95
JE1030	150 Watt Power Supply	$ 69.95
JE1020	5¼" DSDD Disk Drive	$ 89.95
AMBER	12" Monochrome Amber Monitor	$109.95
JE1001	4.77/8MHz Turbo Motherboard (Zero-K RAM – Includes Award BIOS ROM)	$104.95
JE1071	Multi I/O w/Controller and Graphics	$119.95

SAVE $70.70 ***Regular List*** $670.65

JE1005 **Turbo IBM PC/XT Compatible Kit . . $599.95**

IBM AT Compatible Kit

Baby AT! ***NEW!***

Mini-286 6/8/10/12MHz Kit

Part No.	Description	Price
JE1043	1.2M/360K Floppy Controller	$ 49.95
JE1015	XT/AT Style Keyboard	$ 59.95
41256-120	512K RAM (18 Chips)	$ 71.10
JE1012	Baby AT Flip-Top Case	$ 69.95
JE1032	200 Watt Power Supply	$ 89.95
JE1022	5¼" High Density Disk Drive	$109.95
JE1003	Baby AT Motherboard (Zero-K RAM– Includes Award BIOS ROM)	$399.95

SAVE $50.85 ***Regular List*** $850.80

JE1008 **IBM AT Compatible Kit . . . $799.95**

TE5154 EGA Monitor . $399.95
JE1055 EGA Card . . . $149.95
(not incl. – see opposite page)

Storage and Retrieval of Documents, Catalogs, Manuals on CD-ROM

With Online's Image Compression and Expansion Systems, Local Area Network, and Laser Printer Output, we provide complete digitizing, retrieval software, and hardware solutions.

CALL OR WRITE FOR FURTHER INFORMATION

ONLINE COMPUTER SYSTEMS, INC.
Department OPC-B1
20251 Century Boulevard
Germantown, Maryland 20874
(800) 922-9204 or (301) 428-3700
Telex: 3746439 FAX: (301) 428-2903

OEM and Dealer Inquiries are Invited

Circle 156 on Reader Service Card

2400 BAUD INT. MODEM W/PC-TALK SW $159

2400 EXT. MODEM W/SW	$179
1200 INT. MODEM W/SW	$74
1200 EXT. MODEM W/SW	$89
1200 INT. + SERIAL PORT	$89
NEC MULTISPEED	$1299
EPSON 180CPS PRINTER	$179
EPSON FX-286e PRINTER	$435
SEAGATE 20M HD W/CONTL	$299
SEAGATE 30M HD W/CONTL	$309
286 SPEEDCARD FOR XT	$279
MOUSE W/PAINT SW	$49
MOUSE W/DR. HALO II	$60
CLOCK CARD FOR XT	$24
CLOCK/SERIAL CD FOR XT	$41
S/P (2ND S OPT.) FOR AT	$46

OCEAN INTERFACE CO.
P.O. BOX 1666
MONTEREY PARK, CA 91754
(818) 282-0498

Circle 153 on Reader Service Card

Brand Name DISKETTES

29¢ EACH — 5¼" DS DD Min. 100, 39¢ Min. 50
49¢ EACH — Color 5¼" DS DD Min. 50
79¢ EACH — 5¼" IBM-AT HD Min. 50

2 for 1 Replacement Guarantee if found initially defective within six months
With sleeves, tabs, labels. For TYVEC sleeves, add 2¢

99¢ EACH — 3½" SS DD Min. 25 Subject to availability
$1.09 EACH — 3½" DS DD Min. 25 Subject to availability
$4.29 EACH — Box of 10 5¼" DS DD Min. 10 boxes
$5.99 EACH — Box of 10 Color 5¼" DS DD Min 5 boxes Free Plastic Case
$19.20 EACH — Min. 5 600' 120 MB Data Cartridge
$15.90 EACH — Min 5 205 40 MB Mini Cartridge 3M DEI Compatible

Immediate shipping — Guaranteed satisfaction **100% Lifetime Warranty**
Call for greater discounts on larger orders MC/VISA accepted
Shipping: $3.50 per min. order. $1.50 each addtl. lot

1-800-537-1600 OPERATOR NO. 227
MICRO ELECTRONIC PRODUCTS

Circle 254 on Reader Service Card

ICs PROMPT DELIVERY!!!

SAME DAY SHIPPING (USUALLY)
QUANTITY ONE PRICES SHOWN for SEPT. 20, 1987

OUTSIDE OKLAHOMA: NO SALES TAX

DYNAMIC RAM			
1Mbit	1000Kx1	100 ns	$27.50
1Mbit	256Kx4	120 ns	32.00
51258	*256Kx1	100 ns	6.50
4464	64Kx4	150 ns	3.60
41256	256Kx1	80 ns	5.50
41256	256Kx1	100 ns	4.85
41256	256Kx1	120 ns	3.95
41256	256Kx1	150 ns	3.50
41264	2-PORT	120 ns	5.25
EPROM			
27C512	64Kx8	200 ns	$15.50
27C256	32Kx8	250 ns	6.75
27256	32Kx8	250 ns	5.60
27128	16Kx8	250 ns	5.35
STATIC RAM			
43256L-12	32Kx8	120 ns	$11.75
5565PL-15	8Kx8	150 ns	3.25

640K MOTHERBD UPGRADE: Zenith 150, IBM PC XT, Compaq Portable & Plus; hp Vectra

* STATIC COLUMN DRAM FOR 386 COMPAQ

8087-2 $160.00 · 80287-8 $250.00 · 80387-16 $540.00

OPEN 6½ DAYS, 7:30 AM-10 PM SHIP VIA FED-EX ON SAT.

SUNDAYS & HOLIDAYS: SHIPMENT OR DELIVERY, VIA U.S. EXPRESS MAIL

SAT DELIVERY INCLUDED ON FED-EX ORDERS RECEIVED BY:

MasterCard VISA or UPS CASH COD
Factory New, Prime Parts μPoo
MICROPROCESSORS UNLIMITED, INC.
24,000 S. Peoria Ave.
BEGGS, OK 74421
(918) 267-4961

No minimum order. Please note that prices are subject to change. Shipping & insurance extra, & up to $1 for packing materials. Orders received by 9 PM CST can usually be delivered the next morning, via Federal Express Standard Air is $4.00, or guaranteed next day Priority One is $10.50! All parts guaranteed.

Circle 137 on Reader Service Card

NEW

FLOPPY CLOCK™

SPECIAL INTRODUCTORY OFFER

Precision quartz movement. Genuine 5¼" floppy diskette face. Clear floppy diskette storage case—opens for desktop display (lid becomes accessory tray), and closes for wall mount. 1-year warranty. Grid styles available in BLUE, GRAY, or BLACK.

Send: Cash/Check/MO to BEARE ENTERPRISE
1480 RT. 46 SUITE 53-B / PARSIPPANY, NJ 07054
(201) 334-7491 / NJ Res. add 6% sales tax.
Accessories & 'AA' bat. not incl.

Circle 32 on Reader Service Card

JKL's OCTOBER BARGAINS

JKL AT: 80386 (16 MHz) w/case 200W, P.S., Keyboard . . . **$2195.00**

JKL AT 40: Above plus 40 Meg. HD, 1.2 Floppy, Graphics Card, & Monitor . . . **$3295.00**

JKL AT 80: Above plus 80 Meg. HD, 1.2 Floppy, EGA Card & Monitor **$4495.00**

Cases: AT . . . . **$74.10** XT **$35.10**
Power Supply: AT 200W **$85.80** XT 150W **$58.50**
Cards:
Monochrome Graphics Card w/Printer Port . **$50.70**
AT Hard Disk and Floppy Controller Card **$161.20**
Hard Disks: AT 40 Meg, 40MS Hard Disk . **$492.70**
XT 30 Meg Hard Disk w/Controller **$349.70**

JKL

Jack Krochmal, Ltd.
Computers, Peripherals & Supplies
717 Ellsworth Drive
Silver Spring, MD 20910
(301) 565-2910/587-3232
Toll Free: 1-800-JK3-0386

Circle 111 on Reader Service Card

PC220

PC220 turns your PC* into a VT220 terminal!

Join thousands who have discovered GMS's 5 year record of quality emulation software and support.

General Micro Systems
PO Box 5330
Minneapolis MN 55343-1553 USA
(612) 937-1442

Send $1 for Demo Disk.
Call for Free Information Package.

* Also runs on XT, AT, Series II and compatibles.

Circle 88 on Reader Service Card

2400 baud

300 1200 2400 MODEM **$160**

XT/AT, Apple, Atari, Commodore & ....
*Fully Hayes compatible
*Auto dial, auto answer, w 8 LED, speaker, Aluminum case

(SAME AS ABOVE)
300 1200 .... **$99**

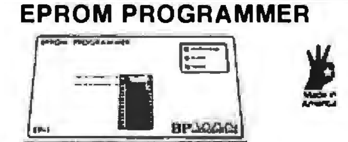
300 1200 for XT AT INTERNAL MODEM **$69**

*Add 7% shipping. **10 days money back guarantee** (less $15 ea for restocking & shipping charge)

FOSTER TECHNOLOGY:
*#4, 222 H St., Blaine, **WA** 98230
Phone: (206) 332-5081
*Phone: (604) 684-2368 **CANADA**
(Add 16% for Canadian Orders)

Circle 85 on Reader Service Card

NEW IMPROVED EPROM PROGRAMMER



STILL JUST $349

- Now supports over 250 specific EPROM's and EEPROM's from AMD, Intel, SEEQ, TI, Fujitsu, Hitachi, Toshiba, and 21 others.
- Programs 87C51, 8749, 8741A, 8742 and other microcontrollers with optional heads
- Splits 16- and 32-bit files
- 30 day money back guarantee.
- Free PCDOS software runs up to 38,400 baud
- Gold Textool ZIF socket
- Same day shipment.

CALL TODAY FOR MORE INFORMATION

BP MICROSYSTEMS
10681 Haddington, #190 / Houston, TX 77043
(713) 461-9430 (800) 225-2102
Telex: 1561477

Circle 41 on Reader Service Card

8 MHz Turbo-XT

$398

- 640K of RAM
- 4.77 or 8 MHz
- 150 Watt Power Supply
- Deluxe Keyboard
- 360K Disk Drive
- 8 Expansion Slots
- 8087 Socket
- One Year Warranty

OPTION A

- High Resolution Amber Flat Screer
- High Resolution Graphics Card
- Parallel Printer Port

ADD $148

- Hi-Res RGB Color Monitor
- High Resolution Graphics Card
- Parallel Printer Port

ADD $298

OPTION C

- 21.3 Megabyte Hard Disk Drive
- Dual Hard Disk Controller Card

ADD $298

Satisfaction Guaranteed or Your Money Back

Option A

- 1.2 MB Drive
- Hard/Floppy Controller
- 30/40 MB Hard Disk
- Hi-Res Amber Monitor
- Monochrome Graphics Card

30 MB add $790 — 40 MB add $930

Option B

- 1.2 MB Drive
- Hard/Floppy Controller
- 30/40 MB Hard Disk
- Hi-Res Color Monitor
- Color Graphics Card

Gel Printer

30 MB add $990 — 40 MB add $1140

Option C

- 1.2 MB
- Hard/Floppy Controller
- 30/40 MB Hard Disk
- EGA Card
- EGA Monitor

30 MB add $1250 — 40 MB add $1400

10 MHz Turbo-AT

$788

- 6 or 10 MHz 80286
- 640K (Expands to 1 MB)
- 80287 Socket
- 200 Watt Power Supply
- Enhanced Keyboard
- Real Time Clock/Calendar
- One Year Warranty

EPSON® PRINTER $148

LX-800, FX-86e, FX-286e, LQ-800, LQ-1000
LQ-850, LQ-1050, EX-800, EX-1000, LQ-2500, GQ-3500

Call for Our Best Price

PANASONIC

1080i Model II	$178
1091i	$198
1092	$328
1592	$428
1595	$458
1524	$588

CITIZEN

120D	$168
MSP-10	$268
MSP-15	$328
MSP-20	$298
MSP-25	$398
MSP-50	$368
MSP-55	$468

TOSHIBA

321 SL	$498
341 SL	$698
P351 II	$958
P351C II	$1088

NEC

P660	$458
P760	$648
P560XL	$888
P565XL	$968
P960XL	$1088
LC-860	$2168
LC-890	$3528

HARD DISK DRIVE

10 MB Full Ht Kit **$199**

Hard Disk, Controller (PC) and Cables

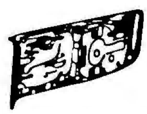
20 MB ½ Ht Kit **$278**

30 MB ½ Ht. Kit **$328**

30 MB for AT....$499 40 MB for AT....$559

21 Megabyte Card

21 MB Card **$358**



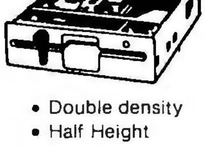
40 MB Card **$498**

- 21.3 MB Formatted
- Lowpower
- Head Park Zone
- Plated Media
- Light 2.4 Lbs.

360K DISK DRIVE

Built By Alpine Electronics to IBM Specifications, same as Qume 142

$65



- 360K
- 48 TPI
- Double-sided
- Double density
- Half Height

3½" DISK DRIVE

FOR YOUR PC/XT/AT

720K Disk Drive for Mass Storage or for Down Loading to your Lap-Top or New IBM PS/2. Requires DOS 3.20

$128

360K DISK DRIVE

Tandon TM100-2 Full Ht. Replacement Drive for Your IBM PC or XT

DELUXE KEYBOARD DRAWER

Ball-Bearing Slides with Ergonomic Palm Rest

DISK DRIVE FOR YOUR APPLE

AS LOW AS **$119**

20 MB Hard Disk	$799
IIc ½ Hard Disk Drive	$119
IIe ½ High Disk Drive	$119
IIe, II+ Disk Drive	$129
Grappler Printer Card	$89
II+ 16K RAM card	$39
1 MB RAM Card w/OK	$129

MicroSpeed Fast 88 7 MHz Turbo Board For Your Turbo

$99

WIN YOUR STATE LOTTO WITH YOUR COMPUTER!

"The home computer is the most powerful tool ever held by man" (or woman for that matter)!

Are you still wasting money with random guesswork? This amazing program will analyze the past winning lotto numbers and produce a powerful probability study on easy to read charts in just seconds. With single key presses from a menu you'll see trends, patterns, odd/even, sum totals, number frequency and more on either your screen or printer. Includes automatic number wheeling, instant updating and a built-in tutorial to get you started fast and easy!

CHECKS & CHARGE CARDS ACCEPTED WITH NO SURCHARGE.

All orders shipped same day (except personal checks)

APPLE & IBM Compatibles	$24.95
Macintosh (requires M/S Basic)	$29.95
Commodore, Atari & Radio Shack	$21.95

Back-Up Copies - $3.00

Please add $2.00 for shipping and handling.

Phone credit given with orders.

(513) 233-2200

SOFT-BYTE
P.O. Box 556, Forest Park
Dayton, Ohio 45405

MasterCard VISA

Circle 216 on Reader Service Card

DATA SWITCHBOXES

DB25 SERIAL/PARALLEL

2 Position	25.90
3 Position	31.90
4 Position	37.90
2 x 2 Matrix	39.90

CENTRONICS

2 Position	27.90
3 Position	33.90
4 Position	39.90
2 x 2 Matrix	41.90

CABLES • Molded w/thumbscrews

	6 FT.	10 FT.
Parallel Printer Cable	6.50	7.50
DB25 Serial Modem/Extender Cable M-M, M-F	6.50	7.50
Centronics Cable M-M	7.50	

Custom lengths and configurations available

MODEMS

	Internal	External
Mitsuba Super Modem 1200	89.50	139.50
Mitsuba Super Modem 2400	199.50	229.50

HARD DRIVES • Includes one-year warranty

20 MB Seagate ST-225 w/controller	299.00
30 MB Seagate ST-238 w/controller	359.00
40 MB Miniscribe 3650 w/controller	465.00

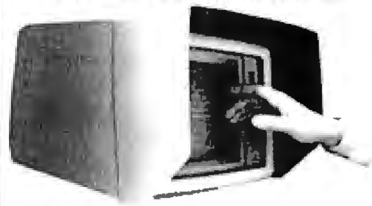
Expansion cards, print buffers and many other accessories also available.

DARE SYSTEMS INTERNATIONAL
1840 North State St., Provo, UT 84604
(801) 377-5591

M/C, VISA, COD Terms accepted upon approval
Call or write for our FREE catalog
(Dealer/Quantity prices available)

Circle 69 on Reader Service Card

The PC VISOR is here!!



Cut glare, increase screen security!

- PC VISOR *shades off* over 50 percent of the light normally reaching your screen. Turn down brightness save your tube and *eyes!*
- Security "Blinder" feature decreases angle of view to the user. No "over-the-shoulder" peekers
- Retail price **only $9.95** · $2 shpg Quantity discounts!

Available in PC beige for IBM, compatibles, Macintosh **for any monitor or terminal!** Proper fit is guranteed (Extra large sizes may require additional charge.) Specify monitor type when ordering

QUALITY ONE PRODUCTION SERVICES
PO BOX 2152, SAN JOSE, CA 95109
(408) 293-7226

Circle 189 on Reader Service Card

SAVE ON

9 TRACK TAPE SYSTEM

FOR IBM PC/XT/AT

- Mainframe to PC Data Transfer
- High Speed Backup
- All Software, Complete System
- Service and Support, easy Installation

call **(818) 343-6505** or write to:
Contech Computer Corp.
P.O. Box 153 Tarzana, Calif. 91356

CONTECH

Circle 56 on Reader Service Card

AMT

APPLIED MICRO TECHNOLOGY

IMPORTER · EXPORTER
P.O. BOX 24783, LOS ANGELES, CA 90024
800 458-0070 (NA)/213 474-0070 (CA)
TLX: 263220 AMT UR

IBM® COMPAQ®

ALL MODELS BEST PRICES FOR EXPORT

IOMEGA® BERNOULLI TECHNOLOGY

All electronic components and computer parts. International orders welcome.

Circle 15 on Reader Service Card

DYNAMIC RAMS

1Mbit	100ns	$26.00
41256	100ns	$ 4.25
41256	120ns	$ 3.35
✓41256	150ns	$ 2.95
4464	150ns	$ 3.35
✓4164	150ns	$ 1.35

■ PROCESSORS

8087-2	8mHz	$147.00
8087	5mHz	$104.00
80287-6	6mHz	$159.00
80287-8	8mHz	$245.00
V-20	8mHz	$ 12.75
V-30	8mHz	$ 12.75

■ STATIC RAMS

6264LP-15	150ns	$ 2.95

■ E PROMS

27512	250ns	$ 9.90
27C256	250ns	$ 5.80
27256	250ns	$ 4.75
27128	250ns	$ 4.45
27C64	200ns	$ 4.95
2764	250ns	$ 3.50
2732A	250ns	$ 3.95

■ 8000's (Parts in stock)

I.C. EXPRESS

15358 Valley Blvd., City of Industry, CA 91746
Phone 818-369-2688 (Mon-Fri • 8-5)

ORDER TOLL FREE

(800) 892-8889 • **(800) 882-8181**
Outside California / Inside California

CALL FOR CURRENT PRICES & VOLUME DISCOUNTS

Price Shown for Cash • MasterCard/VISA add 3% more
Prices are subject to change. Minimum order $10.00
California residents must add 6.5% sales tax
Shipping & Handling: UPS Ground $3.00, UPS Air $7.00 (under 1 lb.)
ALL MERCHANDISE IS 100% GUARANTEED

Circle 107 on Reader Service Card

LIONHEART

STATISTICAL & MANAGEMENT SOFTWARE FOR BUSINESS, TECHNOLOGY, & SCIENCE

PROFESSIONAL SERIES:

Experimental Statistics	$145
Business Statistics	145
Forecasting and Time-Series	145
Sales and Market Forecasting	145
Marketing Statistics	145
Quality Control and Industrial Experiments	145
Exploratory Data Analysis	85
Project Planning (PERT & CPM)	145
Decision Analysis Techniques	145
Optimization	145
Econometrics	145
Biometrics	145

TECHNICAL SERIES:

Multivariate Analysis	125
Cluster Analysis	125
ANOVA	95
Regression	95
Inference	95
Linear & Non-Linear Programming	95
Matrix Routines	95
Decision Trees & Tables	95
ARIMA Methods	95
Inventories & Queues	95

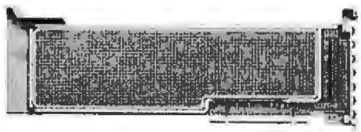
MS-DOS, MACINTOSH, AMIGA, ATARI ST (also ProDOS, C-128, CP/M)

VISA, MC, AMEX, Check

LIONHEART PRESS, INC.
P.O. BOX 379, ALBURG, VT 05440
TEL: (514) 933-4918

Circle 123 on Reader Service Card

IBM PS/2 PROTOTYPE BOARD



- FOR IBM PS/2 — MODELS 50 & 60 **$39.95**
- OVER 3000 1/10" SPACED HOLES WITH SOLDER PADS
- HAS 16-BIT AND VIDEO BUS INTERFACE EXTENSIONS
- HOLES AND PADS FOR MOUNTING UP TO A 39 PIN D-TYPE CONNECTOR AT REAR OF BOARD
- POWER AND GROUND BUSES AROUND EDGE OF BOARD ON BOTH SIDES OF BOARD
- IN STOCK — COMES WITH PLASTIC RETAINERS

NVS CO.

2124 ARBOR DRIVE NEW PORT RICHEY, FL 34655
(813) 376-8191

TERMS: MONEY ORDERS, CHECKS, C.O.D. OR P.O. TO ESTABLISHED COMPANIES
IBM AND PS/2 ARE REGISTERED TRADEMARKS OF INTERNATIONAL BUSINESS MACHINES CORPORATION

Circle 151 on Reader Service Card

DISKETTE

BOXED 39¢

Boxed Diskettes are pkgd 10 to a plain white box and include sleeves, labels, & Hub ring. 100% error free. Lifetime warranty. Major US Mfrs.

Your cost **$3.90/box**

Add $3 per 100 diskettes for shipping & handling

BULK 35¢

5¼ DSDD Includes sleeves, Labels & Tabs

5.25HD	.99
3.50SS	.99
3.50DS	.99

MONTHLY SPECIALS
Seagate 20mb $275 30mb $315
Includes controller, cables, and software

(213) 479-0345 Data Bureau Inc.
1731 So. LaCienga Blvd., Los Angeles, CA 90035

Please call as prices change. We will meet any advertised price on diskettes.

Circle 70 on Reader Service Card

Circle 110 on Reader Service Card

8087 5MHz $99.00 — 80287 6 MHz $179.95

INTERFACE CARDS
FROM MODULAR CIRCUIT TECHNOLOGY

MULTIFUNCTION CARDS

MCT-MF MULTIFUNCTION $79.95
ALL THE FEATURES OF 6 PACK+ AT HALF THE PRICE
- 0-384K DYNAMIC RAM USING 4164s
- SERIAL, PARALLEL, GAME PORTS, CLOCK/CALENDAR

MCT-MGMIO MONOGRAPHICS I/O $119.95
TOTAL SYSTEM CONTROL FROM A SINGLE SLOT
- 2 FLOPPY CONT, SERIAL, PARALLEL, GAME PORT, CLOCK/CAL
- RUN COLOR GRAPHICS SOFTWARE ON A MONOCHROME MONITOR

MCT-MIO MULTI I/O FLOPPY $79.95
A PERFECT COMPANION FOR OUR MOTHERBOARDS
- SERIAL, PARALLEL, GAME PORT, CLOCK/CALENDAR
- SUPPORTS UP TO 2 360K FLOPPIES, 720K WITH DOS 3.2

MIO-SERIAL 2nd SERIAL PORT $15.95

MCT-IO MULTI I/O CARD $59.95
USE WITH MCT-FH FOR A MINIMUM OF SLOTS USED
- SERIAL PORT, CLOCK/CALENDAR WITH A BATTERY BACK-UP
- PARALLEL PRINTER PORT ADDRESSABLE AS LPT1 OR LPT2

IO-SERIAL 2nd SERIAL PORT $15.95

MCT-ATMF AT MULTIFUNCTION $139.95
ADDS UP TO 3 MEGABYTES OF RAM TO THE AT
- USER EXPANDABLE TO 1.5 MB OF MEMORY (ZERO K INSTALLED)
- INCLUDES SERIAL PORT AND PARALLEL PORT

ATMF-SERIAL 2nd SERIAL PORT $24.95
MCT-ATMF-MC PIGGYBACK BOARD (NO MEMORY) $29.95

MCT-ATIO AT MULTI I/O $59.95
USE WITH MCT-ATFH FOR A MINIMUM OF SLOTS USED
- SERIAL, PARALLEL AND GAME PORTS
- USES 16450 SERIAL SUPPORT CHIPS FOR HIGH SPEED OPERATION

ATIO-SERIAL 2nd SERIAL PORT $24.95

MEMORY CARDS

MCT-RAM 576K RAM CARD $59.95
A CONTIGUOUS MEMORY SOLUTION IN A SHORT SLOT
- USER SELECTABLE CONFIGURATION AMOUNTS UP TO 576K USING 64K & 256K RAM CHIPS (ZERO K INSTALLED)

MCT-EMS EXPANDED MEMORY CARD $129.95
2 MB OF LOTUS/INTEL/MICROSOFT COMPATIBLE MEMORY FOR XT
- CONFORMS TO LOTUS/INTEL EMS
- USER EXPANDABLE TO 2 MB
- EXPANDED/CONVENTIONAL MEMORY, RAMDISK AND SPOOLER

MCT-ATRAM AT 4 MB RAM CARD $149.95
A POWER USER'S DREAM: 4 MB OF MEMORY FOR AT
- BACKFILL TO 640K, FLEXIBLE STARTING ADDRESS
- UP TO 2 MB ON CARD, 4 MB WITH OPTIONAL PIGGYBACK BOARD

MCT-ATRAM-MC PIGGYBACK BOARD (NO MEMORY) $39.95

MCT-ATEMS AT COMPATIBLE EMS $139.95
CAN BE USED FOR EXPANDED, EXTENDED OR CONVENTIONAL MEMORY
- CONFORMS TO LOTUS/INTEL EMS
- USER EXPANDABLE TO 2 MB
- RAMDISK, PRINT SPOOLER AND LIM/EMS SOFTWARE INCLUDED

DRIVE CONTROLLERS

MCT-FDC FLOPPY DISK CONTROLLER $29.95
QUALITY DESIGN OFFERS 4 FLOPPY CONTROL IN A SINGLE SLOT
- INTERFACES UP TO 4 FDDs TO AN IBM PC OR COMPATIBLE
- SUPPORTS BOTH DS/DD AND DS/QD WITH DOS 3.2

MCT-HDC HARD DISK CONTROLLER $79.95
HD CONTROL FOR WHAT OTHERS CHARGE FOR FLOPPY CONTROL
- SUPPORTS 16 DRIVE SIZES INCLUDING 5, 10, 20, 30 & 40 MB
- DIVIDE 1 LARGE DRIVE INTO 2 SMALLER, LOGICAL DRIVES

MCT-FDC-1.2 1.2MB FLOPPY CONTROLLER $69.95
ADD VERSATILITY & CAPACITY TO YOUR XT
- SUPPORTS 2 DRIVES, BOTH MAY BE 360K OR 1.2 MEG
- ALLOWS DATA TO FLOW FREELY FROM XTs TO ATs

MCT-FH FLOPPY/HARD CONTROLLER $139.95
SYSTEM STARVED FOR SLOTS? SATISFY IT WITH THIS TIMELY DESIGN
- INTERFACES UP TO 2 FDDs & 2 HDDs, CABLING FOR 2 FDDs & 1 HDD
- SUPPORTS BOTH DS/DD & DS/QD WITH DOS 3.2

MCT-ATFH AT FLOPPY/HARD CONTROLLER $149.95
FLOPPY/HARD DISK CONTROL IN A TRUE AT DESIGN
- SUPPORTS UP TO 2 360K/720K/1.2MB FDDs AS WELL AS 2 HDDs USING STANDARD CONTROL TABLES

DISPLAY CARDS

MCT-EGA ENHANCED GRAPHICS ADAPTOR $149.95
100% IBM COMPATIBLE, PASSES IBM EGA DIAGNOSTICS
- 256K OF VIDEO RAM ALLOWS 640 x 350 IN 16 OF 64 COLORS
- COMPATIBLE WITH COLOR AND MONOCHROME ADAPTORS

MCT-CG COLOR GRAPHICS ADAPTOR $49.95
COMPATIBLE WITH IBM COLOR GRAPHICS STANDARDS
- SHORT SLOT SUPPORTS RGB, COLOR & COMPOSITE MONOCHROME
- 640/320 x 200 RESOLUTION, LIGHT PEN INTERFACE

MCT-MGP MONOCHROME GRAPHICS $59.95
TRUE HERCULES COMPATIBILITY, SUPPORTS LOTUS 123
- SOFTWARE DRIVER ALLOWS COLOR GRAPHICS PROGRAMS TO RUN ON A MONOCHROME MONITOR
- PARALLEL PRINTER PORT

HALF HEIGHT HARD DISK DRIVES

40 MB $469
Seagate model ST-251 5¼" half height
FAST 40ms access time

60 MB $649
Seagate model ST-277 5¼" half height
FAST 40ms access time (RLL)

HALF HEIGHT HARD DISK SYSTEMS

20 MB $289 **30 MB $329**

Systems include Seagate ½ height hard drive, hard drive controller, cables and instructions.
All drives are pre-tested and warranted for one year.

IBM COMPATIBLE MOTHERBOARDS
FROM MODULAR CIRCUIT TECHNOLOGY

MCT-TURBO TURBO 4.77/8 MHz $109.95
- 4.77 OR 8 MHz OPERATION WITH 8088-2 & OPTIONAL 8087-2 CO-PROCESSOR
- DYNAMICALLY ADJUSTS SPEED FOR MAXIMUM THROUGHPUT & RELIABILITY DURING DISK I/O
- CHOICE OF NORMAL/TURBO MODE OR SOFTWARE SELECT PROCESSOR SPEED

MCT-XTMB STANDARD MOTHERBOARD $97.95

MCT-ATMB 80286 6/8 MHz $379.95
- 8 SLOT (2 EIGHT BIT, 6 SIXTEEN BIT) AT MOTHERBOARD
- HARDWARE SELECTION OF 6 OR 8 MHz
- 1 WAIT STATE
- KEYLOCKSUPPORTED, RESET SWITCH, FRONT PANEL LED INDICATOR
- SOCKETS FOR 1 MB OF RAM AND 80287
- BATTERY BACKED CLOCK

MCT-BATMB MINI 80286 $429.95
- REPLACEMENT BOARD FOR XT STYLE CHASSIS
- OPERATE AT 6/10 MHz WITH UP TO 1MB ON-BOARD MEMORY (ZERO K INSTALLED)
- SOCKET FOR 80287 MATH CO-PORCESSOR
- BATTERY BACKED CLOCK
- 8 SLOTS: 2 EIGHT BIT, 6 SIXTEEN BIT
- USES CHIPS & TECHNOLOGY CHIP SET FOR RELIABILITY AND SMALL SIZE

IBM COMPATIBLE KEYBOARDS

MCT-5339 $79.95
- IBM ENHANCED STYLE LAYOUT
- SOFTWARE AUTOSENSE FOR XT OR AT COMPATIBLES
- LED INDICATORS
- AUTO REPEAT FEATURE
- SEPARATE CURSOR PAD

MCT-5060 $59.95
- IBM AT STYLE LAYOUT
- SOFTWARE AUTOSENSE FOR XT OR AT COMPATIBLES
- LED INDICATORS
- AUTO REPEAT FEATURE

MCT-5150 XT STYLE LAYOUT $49.95

MCT-5151 KB5151 EQUIV. $69.95

WHY BUY A SYSTEM FROM JDR?
- BUILD IT YOURSELF AND SAVE
- QUALITY COMPONENTS AT COMPETITIVE PRICES
- TOLL FREE TECHNICAL SUPPORT IN THE U.S. AND CANADA
- MONEY BACK GUARANTEE (ASK FOR DETAILS)
- MORE THAN 10,000 SOLD!

"I am very pleased with the computer system I purchased from you and all the outstanding support and courteous people on your staff. Your attitude & commitment to customer satisfaction is exceptional. YOUR ATTITUDE SHOULD BE INDUSTRY STANDARD!"

Daryl Hammond

BUILD YOUR OWN 256K XT COMPATIBLE

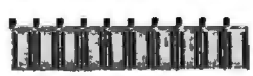
- MOTHERBOARD
- 256K OF MEMORY
- 135 WATT POWER SUPPLY
- FLIP-TOP CASE
- 5060 STYLE KEYBOARD
- 360K FLOPPY DRIVE
- DRIVE CONTROLLER
- MONOCHROME MONITOR
- GRAPHICS ADAPTOR

FOR ONLY $499.15

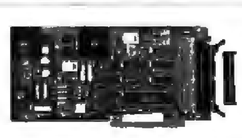
ANYONE CAN BUILD A SYSTEM IN ABOUT 2 HOURS USING A SCREWDRIVER AND OUR EASY-TO-FOLLOW INSTRUCTIONS!

DEVELOPMENT TOOLS
FROM MODULAR CIRCUIT TECHNOLOGY

MCT-EPROM PROGRAMMER $129.95
PROGRAMS 27xx & 27xxx EPROMS UP TO 27512
- SUPPORTS VARIOUS PROGRAMMING FORMATS AND VOLTAGES
- SPLIT OR COMBINE CONTENTS OF SEVERAL EPROMS OF DIFFERENT SIZES
- READ, WRITE, COPY, ERASE CHECK AND VERIFY
- SOFTWARE FOR HEX AND INTEL HEX FORMATS



MCT-EPROM-4 4 GANG PROGRAMMER $189.95
MCT-EPROM-10 10 GANG PROGRAMMER $299.95



MCT-PAL PAL PROGRAMMER $269.95

MCT-MP PROCESSOR PROG. $199.95

CALL FOR VOLUME QUOTES

BOMB

YOU CHOOSE THE BEST ARTICLE EACH MONTH

BYTE's ongoing monitor box (BOMB) lets you rate each article you've read in BYTE as excellent, good, fair, or poor. We tally your votes, total the points, tell you who won, and award the two top-rated nonstaff authors $100 and $50, respectively. An additional $50 award for quality goes to the non-staff author with the best average score (total points divided by the number of voters). If you prefer, you can use BIX as your method of voting. We welcome your participation.

ARTICLE#	PAGE	ARTICLE	AUTHOR(S)
1	11	**What's New**	staff
2	33	**The Technical Implications of the PS/2**	Vose
3	49	**TSRs Past and Future: MS-DOS and OS/2**	Duncan
4	59	**The 32-bit Micro Channel**	Shiell
5	67	**PS/2 Video Programming**	Wilton
6	83	**Comparing IBM's Micro Channel and Apple's NuBus**	Cornejo, Lee
7	97	**Spying on Windows**	Geary
8	115	**The State of Numerics**	Fried
9	125	**286/386 Protected-Mode Programming**	Barnum
10	133	**The IBM RT Gets Connected**	Levitt
11	143	**Application Input Drivers**	Sagan
12	159	**Better Batch Files Through Assembly Language**	Claff
13	173	**IBM PC Family BIOS Comparison**	Shiell
14	185	**Comparing Disk-Allocation Methods**	Weissman
15	193	**Rating the IBM Compatibles**	Brookshire
16	201	**Windows for BASIC**	Ross
17	215	**Pipes and Filters**	Baker
18	219	**A Timing-Independent BIOS**	Cohen, Hanel
19	225	**Three Bus Interface Designs for the PC**	Drummond

EDITORIAL INDEX BY COMPANY

Index of companies covered in articles, columns, or news stories in this issue.
Each reference is to the first page of the article or section in which the company name appears.

READER SERVICE

To get further information on the products advertised in BYTE, fill out the reader service card by circling the numbers on the card that correspond to the inquiry number listed with the advertiser. This index is provided as an additional service by the publisher, who assumes no liability for errors or omissions.

Also included in this issue:
Special Advertising Supplement to U.S. Subscribers
from Priority One Electronics

Alphabetical Index to Advertisers

** Correspond directly with company.*

READER SERVICE

To get further information on the products advertised in BYTE, fill out the reader service card by circling the numbers on the card that correspond to the inquiry number listed with the advertiser. This index is provided as an additional service by the publisher, who assumes no liability for errors or omissions.

* Correspond directly with company

Index to Advertisers by Product Category

Continued

READER SERVICE

Also included in this issue:
Special Advertising Supplement to U.S. Subscribers
from Priority One Electronics

** Correspond directly with company.*

TIPS

SUBSCRIBERS ONLY!*

Use BYTE's Telephone Inquiry Processing System

Usi g TIPS ca bri g product information as much as 10 days earlier.

SEND FOR YOUR SUBSCRIBER I.D. CARD

1) If you are a new subscriber or have lost your I.D. card, circle #1 on the Reader Service Card; **attach mailer label**. We will immediately send your personal TIPS subscriber card.

GET PREPARED

2) Write your Subscriber Number, as printed on your Subscriber I.D. Card, in boxes in Step 5 below. (Do not add 0's to fill in blank boxes)

3) Write numbers for information desired in boxes in Step 7b below. (Do not add 0's to fill in blank boxes.)

CALL TIPS

4) Now, on a Touch-Tone telephone dial: **(413) 442-2668** and wait for voice commands.

ENTER YOUR SUBSCRIBER AND ISSUE NUMBERS

5) When TIPS says: "Enter Subscriber Number"
(Enter by pushing the numbers and symbols [# or * enclosed in the boxes] on telephone pad ignoring blank boxes)
Enter ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ [#] [#]

6) When TIPS says "Enter magazine code & issue code"
Enter [1] [#] [1] [4] [7] [#] [#]

ENTER YOUR INQUIRIES

7a) When TIPS says "Enter (next) Inquiry Number"
Enter one inquiry selection from below (ignore blank boxes)

b) Repeat 7a as needed (maximum 17 inquiry numbers)

1. ☐ ☐ ☐ [#] [#] 6. ☐ ☐ ☐ [#] [#] 10. ☐ ☐ ☐ [#] [#] 14. ☐ ☐ ☐ [#] [#]
2. ☐ ☐ ☐ [#] [#] 7. ☐ ☐ ☐ [#] [#] 11. ☐ ☐ ☐ [#] [#] 15. ☐ ☐ ☐ [#] [#]
3. ☐ ☐ ☐ [#] [#] 8. ☐ ☐ ☐ [#] [#] 12. ☐ ☐ ☐ [#] [#] 16. ☐ ☐ ☐ [#] [#]
4. ☐ ☐ ☐ [#] [#] 9. ☐ ☐ ☐ [#] [#] 13. ☐ ☐ ☐ [#] [#] 17. ☐ ☐ ☐ [#] [#]
5. ☐ ☐ ☐ [#] [#]

END SESSION

8) End session by entering [*] [*] [9] [1] [#] [#]

9) Hang up after hearing final message
If you are a subscriber and need assistance, call (603) 924-9281.

Available 24 Hours, 7 Days a Week

If you are not a subscriber fill out the subscription card found in this issue or call BYTE Circulation 800-423-8912.
*Domestic and Canadian Subscribers Only!

FREE Information Retrieval Service

To assist you in making your evaluations, purchasing decisions, or recommendations, you can request further information directly from the manufacturer or service company on products and services advertised in this issue. There is no charge, no obligation. Just complete and mail the attached post-paid, self-addressed reply card, and we'll do the rest.

1. Circle numbers on reply card which correspond to numbers assigned to items of interest to you.
2. Check all the appropriate answers to questions "A" through "F".
3. Print your name and address and mail.

Fill out this coupon carefully. PLEASE PRINT. Requests cannot be honored unless the zip code is included. This card is valid for 6 months from cover date.

IBM
47IRNA

A. What is your primary job function? (Check one only)
1 ☐ Business Owner, General Management, Administrative
2 ☐ MIS/DP, Programming
3 ☐ Engineering/Scientific, R&D
4 ☐ Professional (law, medicine, accounting)
5 ☐ Other

B. How many people does your company employ?
1 ☐ 25 or fewer
2 ☐ 26-99
3 ☐ 100-499
4 ☐ 500-999
5 ☐ 1000 or more

C. Reason for request: (Check all that apply).
1 ☐ Business use for yourself
2 ☐ Business use for your company
3 ☐ Personal use

D. Your next step after information is received:
1 ☐ Purchase order
2 ☐ Evaluation
3 ☐ Specification/Recommendation

E. Please indicate the product categories for which you influence the selection or purchase at your (or your client's) company or organization. (Check all that apply).
1 ☐ Microcomputers
2 ☐ Peripherals
3 ☐ Software
4 ☐ Accessories and supplies

F. For how many microcomputers do you influence the purchase of products at your (or your client's) company or organization?
1 ☐ 1
2 ☐ 2-4
3 ☐ 5-9
4 ☐ 10 or more

Name ______________________
Title ______________________
Company ______________________
Address ______________________
City ______________ State ______________
Zip ______________ Telephone ______________

CIRCLE FOR FREE INFORMATION

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54
55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81
82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100	101	102	103	104	105	106	107	108
109	110	111	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127	128	129	130	131	132	133	134	135
136	137	138	139	140	141	142	143	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159	160	161	162
163	164	165	166	167	168	169	170	171	172	173	174	175	176	177	178	179	180	181	182	183	184	185	186	187	188	189
190	191	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207	208	209	210	211	212	213	214	215	216
217	218	219	220	221	222	223	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239	240	241	242	243
244	245	246	247	248	249	250	251	252	253	254	255	256	257	258	259	260	261	262	263	264	265	266	267	268	269	270
271	272	273	274	275	276	277	278	279	280	281	282	283	284	285	286	287	288	289	290	291	292	293	294	295	296	297
298	299	300	301	302	303	304	305	306	307	308	309	310	311	312	313	314	315	316	317	318	319	320	321	322	323	324
325	326	327	328	329	330	331	332	333	334	335	336	337	338	339	340	341	342	343	344	345	346	347	348	349	350	351
352	353	354	355	356	357	358	359	360	361	362	363	364	365	366	367	368	369	370	371	372	373	374	375	376	377	378
379	380	381	382	383	384	385	386	387	388	389	390	391	392	393	394	395	396	397	398	399	400	401	402	403	404	405
406	407	408	409	410	411	412	413	414	415	416	417	418	419	420	421	422	423	424	425	426	427	428	429	430	431	432
433	434	435	436	437	438	439	440	441	442	443	444	445	446	447	448	449	450	451	452	453	454	455	456	457	458	459
460	461	462	463	464	465	466	467	468	469	470	471	472	473	474	475	476	477	478	479	480	481	482	483	484	485	486
487	488	489	490	491	492	493	494	495	496	497	498	499	500	501	502	503	504	505	506	507	508	509	510	511	512	513
514	515	516	517	518	519	520	521	522	523	524	525	526	527	528	529	530	531	532	533	534	535	536	537	538	539	540
541	542	543	544	545	546	547	548	549	550	551	552	553	554	555	556	557	558	559	560	561	562	563	564	565	566	567
568	569	570	571	572	573	574	575	576	577	578	579	580	581	582	583	584	585	586	587	588	589	590	591	592	593	594
595	596	597	598	599	600	601	602	603	604	605	606	607	608	609	610	611	612	613	614	615	616	617	618	619	620	621
622	623	624	625	626	627	628	629	630	631	632	633	634	635	636	637	638	639	640	641	642	643	644	645	646	647	648
649	650	651	652	653	654	655	656	657	658	659	660	661	662	663	664	665	666	667	668	669	670	671	672	673	674	675
676	677	678	679	680	681	682	683	684	685	686	687	688	689	690	691	692	693	694	695	696	697	698	699	700	701	702
703	704	705	706	707	708	709	710	711	712	713	714	715	716	717	718	719	720	721	722	723	724	725	726	727	728	729
730	731	732	733	734	735	736	737	738	739	740	741	742	743	744	745	746	747	748	749	750	751	752	753	754	755	756
757	758	759	760	761	762	763	764	765	766	767	768	769	770	771	772	773	774	775	776	777	778	779	780	781	782	783
784	785	786	787	788	789	790	791	792	793	794	795	796	797	798	799	800	801	802	803	804	805	806	807	808	809	810
811	812	813	814	815	816	817	818	819	820	821	822	823	824	825	826	827	828	829	830	831	832	833	834	835	836	837
838	839	840	841	842	843	844	845	846	847	848	849	850	851	852	853	854	855	856	857	858	859	860	861	862	863	864
865	866	867	868	869	870	871	872	873	874	875	876	877	878	879	880	881	882	883	884	885	886	887	888	889	890	891
892	893	894	895	896	897	898	899	900	901	902	903	904	905	906	907	908	909	910	911	912	913	914	915	916	917	918
919	920	921	922	923	924	925	926	927	928	929	930	931	932	933	934	935	936	937	938	939	940	941	942	943	944	945
946	947	948	949	950	951	952	953	954	955	956	957	958	959	960	961	962	963	964	965	966	967	968	969	970	971	972
973	974	975	976	977	978	979	980	981	982	983	984	985	986	987	988	989	990	991	992	993	994	995	996	997	998	999

Fill out this coupon carefully. PLEASE PRINT. Requests cannot be honored unless the zip code is included. This card is valid for 6 months from cover date.

IBM
47IRNA

A. What is your primary job function? (Check one only)
1 ☐ Business Owner, General Management, Administrative
2 ☐ MIS/DP, Programming
3 ☐ Engineering/Scientific, R&D
4 ☐ Professional (law, medicine, accounting)
5 ☐ Other

B. How many people does your company employ?
1 ☐ 25 or fewer
2 ☐ 26-99
3 ☐ 100-499
4 ☐ 500-999
5 ☐ 1000 or more

C. Reason for request: (Check all that apply).
1 ☐ Business use for yourself
2 ☐ Business use for your company
3 ☐ Personal use

D. Your next step after information is received:
1 ☐ Purchase order
2 ☐ Evaluation
3 ☐ Specification/Recommendation

E. Please indicate the product categories for which you influence the selection or purchase at your (or your client's) company or organization. (Check all that apply).
1 ☐ Microcomputers
2 ☐ Peripherals
3 ☐ Software
4 ☐ Accessories and supplies

F. For how many microcomputers do you influence the purchase of products at your (or your client's) company or organization?
1 ☐ 1
2 ☐ 2-4
3 ☐ 5-9
4 ☐ 10 or more

Name ______________________
Title ______________________
Company ______________________
Address ______________________
City ______________ State ______________
Zip ______________ Telephone ______________

CIRCLE FOR FREE INFORMATION

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54
55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81
82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100	101	102	103	104	105	106	107	108
109	110	111	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127	128	129	130	131	132	133	134	135
136	137	138	139	140	141	142	143	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159	160	161	162
163	164	165	166	167	168	169	170	171	172	173	174	175	176	177	178	179	180	181	182	183	184	185	186	187	188	189
190	191	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207	208	209	210	211	212	213	214	215	216
217	218	219	220	221	222	223	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239	240	241	242	243
244	245	246	247	248	249	250	251	252	253	254	255	256	257	258	259	260	261	262	263	264	265	266	267	268	269	270
271	272	273	274	275	276	277	278	279	280	281	282	283	284	285	286	287	288	289	290	291	292	293	294	295	296	297
298	299	300	301	302	303	304	305	306	307	308	309	310	311	312	313	314	315	316	317	318	319	320	321	322	323	324
325	326	327	328	329	330	331	332	333	334	335	336	337	338	339	340	341	342	343	344	345	346	347	348	349	350	351
352	353	354	355	356	357	358	359	360	361	362	363	364	365	366	367	368	369	370	371	372	373	374	375	376	377	378
379	380	381	382	383	384	385	386	387	388	389	390	391	392	393	394	395	396	397	398	399	400	401	402	403	404	405
406	407	408	409	410	411	412	413	414	415	416	417	418	419	420	421	422	423	424	425	426	427	428	429	430	431	432
433	434	435	436	437	438	439	440	441	442	443	444	445	446	447	448	449	450	451	452	453	454	455	456	457	458	459
460	461	462	463	464	465	466	467	468	469	470	471	472	473	474	475	476	477	478	479	480	481	482	483	484	485	486
487	488	489	490	491	492	493	494	495	496	497	498	499	500	501	502	503	504	505	506	507	508	509	510	511	512	513
514	515	516	517	518	519	520	521	522	523	524	525	526	527	528	529	530	531	532	533	534	535	536	537	538	539	540
541	542	543	544	545	546	547	548	549	550	551	552	553	554	555	556	557	558	559	560	561	562	563	564	565	566	567
568	569	570	571	572	573	574	575	576	577	578	579	580	581	582	583	584	585	586	587	588	589	590	591	592	593	594
595	596	597	598	599	600	601	602	603	604	605	606	607	608	609	610	611	612	613	614	615	616	617	618	619	620	621
622	623	624	625	626	627	628	629	630	631	632	633	634	635	636	637	638	639	640	641	642	643	644	645	646	647	648
649	650	651	652	653	654	655	656	657	658	659	660	661	662	663	664	665	666	667	668	669	670	671	672	673	674	675
676	677	678	679	680	681	682	683	684	685	686	687	688	689	690	691	692	693	694	695	696	697	698	699	700	701	702
703	704	705	706	707	708	709	710	711	712	713	714	715	716	717	718	719	720	721	722	723	724	725	726	727	728	729
730	731	732	733	734	735	736	737	738	739	740	741	742	743	744	745	746	747	748	749	750	751	752	753	754	755	756
757	758	759	760	761	762	763	764	765	766	767	768	769	770	771	772	773	774	775	776	777	778	779	780	781	782	783
784	785	786	787	788	789	790	791	792	793	794	795	796	797	798	799	800	801	802	803	804	805	806	807	808	809	810
811	812	813	814	815	816	817	818	819	820	821	822	823	824	825	826	827	828	829	830	831	832	833	834	835	836	837
838	839	840	841	842	843	844	845	846	847	848	849	850	851	852	853	854	855	856	857	858	859	860	861	862	863	864
865	866	867	868	869	870	871	872	873	874	875	876	877	878	879	880	881	882	883	884	885	886	887	888	889	890	891
892	893	894	895	896	897	898	899	900	901	902	903	904	905	906	907	908	909	910	911	912	913	914	915	916	917	918
919	920	921	922	923	924	925	926	927	928	929	930	931	932	933	934	935	936	937	938	939	940	941	942	943	944	945
946	947	948	949	950	951	952	953	954	955	956	957	958	959	960	961	962	963	964	965	966	967	968	969	970	971	972
973	974	975	976	977	978	979	980	981	982	983	984	985	986	987	988	989	990	991	992	993	994	995	996	997	998	999

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS MAIL PERMIT NO. 176 DALTON, MA

POSTAGE WILL BE PAID BY ADDRESSEE

BYTE
READER SERVICE
PO Box 298
Dalton, MA 01227-0298
USA

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS MAIL PERMIT NO. 176 DALTON, MA

POSTAGE WILL BE PAID BY ADDRESSEE

BYTE
READER SERVICE
PO Box 298
Dalton, MA 01227-0298
USA

For your own subscription to BYTE, complete this card and mail.

47IRNU-0

Name ____________

Address ____________

City ____________

State ________ Zip ________ Country ________

☐ Bill me (U.S.A., Canada, Mexico only)

☐ Check enclosed (All checks must be payable in U.S. funds drawn on a U.S. bank.)

Charge to my ☐ MasterCard ☐ VISA

Card # ____________ Expiration Date ________

Signature ____________

Subscription Rates

	USA	Canada /Mexico
ONE YEAR:	☐$22	☐$25
TWO YEARS:	☐$40	☐$45
THREE YEARS:	☐$58	☐$65

☐ $69 Europe (air delivery)

☐ $37 Europe and worldwide (surface mail)

Air mail rates outside Europe available upon request.

Please allow 6–8 weeks for processing. Thank you.

For your own subscription to BYTE, complete this card and mail.

47IRNU-0

Name ______

Address ______

City ______

State ______ Zip ______ Country ______

☐ Bill me (U.S.A., Canada, Mexico only)

☐ Check enclosed (All checks must be payable in U.S. funds drawn on a U.S. bank.)

Charge to my ☐ MasterCard ☐ VISA

Card # ______ Expiration Date ______

Signature ______

Subscription Rates

	USA	Canada /Mexico
ONE YEAR:	☐$22	☐$25
TWO YEARS:	☐$40	☐$45
THREE YEARS:	☐$58	☐$65

☐ $69 Europe (air delivery)

☐ $37 Europe and worldwide (surface mail)

Air mail rates outside Europe available upon request.

Please allow 6–8 weeks for processing. Thank you.

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS MAIL PERMIT NO. 201 PISCATAWAY, NJ

POSTAGE WILL BE PAID BY ADDRESSEE

BYTE
Subscription Dept.
P.O. Box 6807
Piscataway, NJ 08855-9940

Make a career move!

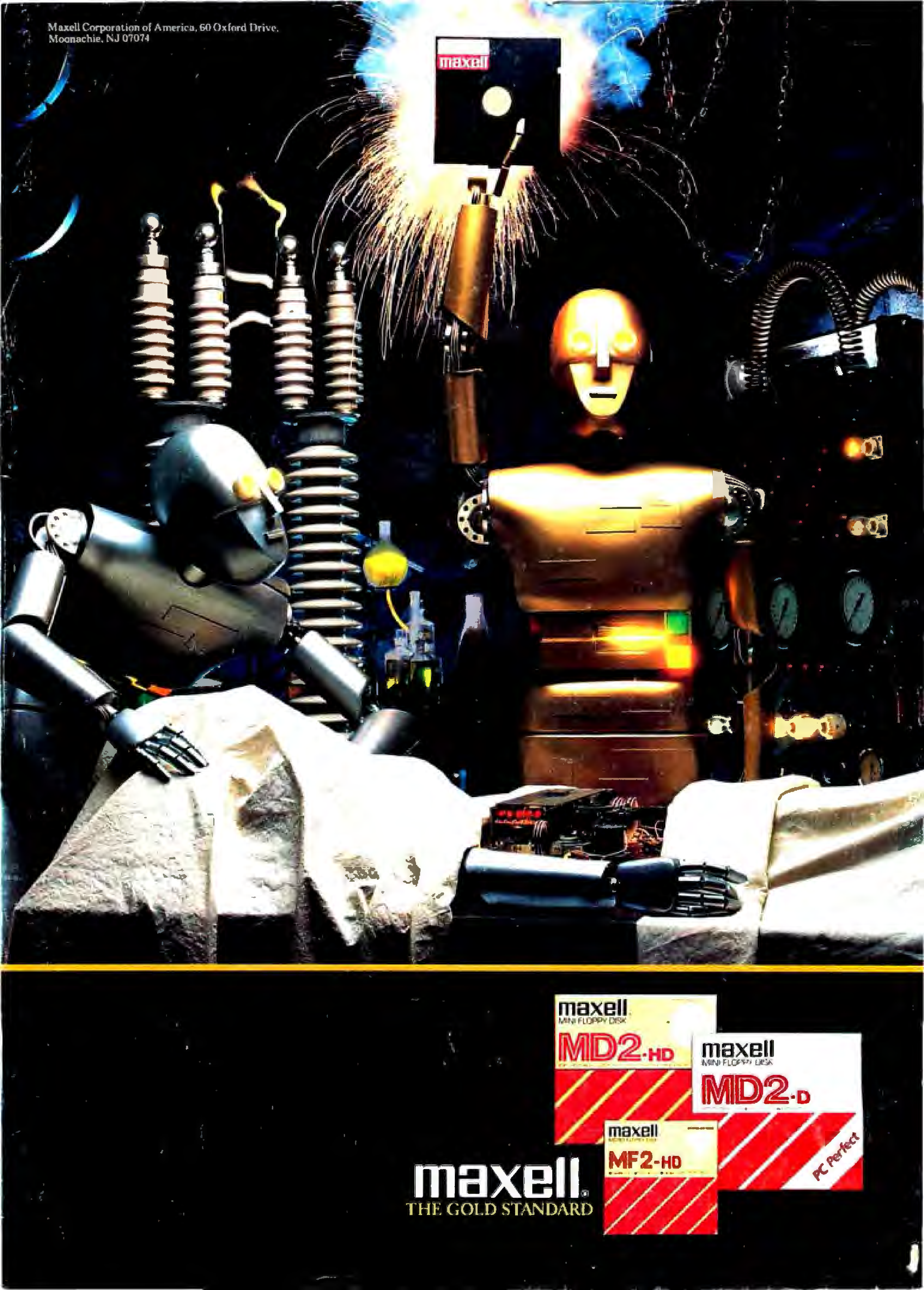
Get your *own* copy of BYTE!

Sharing is nice but not practical when you have to stay on the leading edge of microcomputer technology. From each BYTE issue theme to "MICROBYTES" and "Best of BIX", no other magazine—or group of magazines—can deliver you the vital technical information packaged in one issue of BYTE.

Don't rely on others for BYTE information—waiting on-line can be hazardous to your job. Make a career move! Have BYTE delivered to *your* door in *your* name. Use one of the adjacent cards to ensure your position in microcomputer technology.

For direct ordering call toll free weekdays 9:00am-5:00pm EST: 1 800 423-8272 (in New Jersey: 1 800 367-0218).

Welcome to BYTE Country!



Maxell Corporation of America, 60 Oxford Drive, Moonachie, NJ 07074
maxell
maxell MINI FLOPPY DISK MD2-HD
maxell MINI FLOPPY DISK MD2-D
PC Perfect
maxell MF2-HD
maxell.
THE GOLD STANDARD