

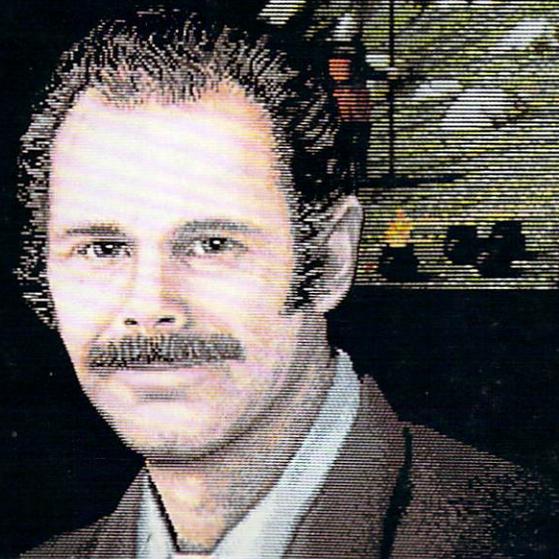
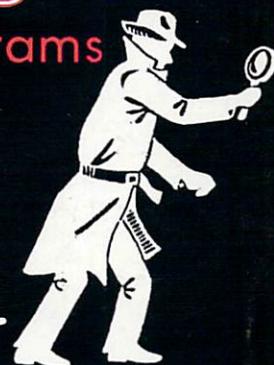
HOUSEHOLD INVENTORY SYSTEM in AmigaBASIC™

# Amazing Computing™

Volume 2 Number 4  
U.S.A. \$3.50  
Canada \$4.50

Commodore Amiga™ Information & Programs

Sleuthing Amiga Public Domain with CLI



*An Interview with  
Amazing Amiga Artist  
JIM SACHS*

# VIZAWRITE

## PERSONAL WORD PROCESSOR

### AMIGA

Announcing the first desktop publishing word processor for the exciting Commodore Amiga. VizaWrite Amiga is a brand new development of a product that has been a best seller for many years. Developed entirely in machine code, VizaWrite has the speed, compactness and style that makes the most of the Amiga.

■ VizaWrite brings desktop publishing to the AMIGA!! Combine pictures from your favorite "Paint" program into a document, reduce or enlarge pictures at any time and then print it!! For high quality presentation of both text and graphics.

■ VizaWrite supports all AMIGA proportional and fixed-width fonts. True proportional layout gives instant pixel accuracy to margins, tabulations and justification. Switch fonts at any time - underline, italic, bold, and superscript/subscript all show on-screen as they will print.

■ VizaWrite is extremely easy to use. We've made sure that our software is presented in a logical and natural way. Using pull-down menus, requestor boxes, and mouse selection of activities combine to make document preparation more effective and enjoyable.

■ VizaWrite is well behaved, allowing you to use the multi-tasking capabilities and run several programs simultaneously.

■ VizaWrite is broad minded, allowing you to include text from Textcraft, pictures from Graphicraft, Deluxe Paint® and most other products.

■ VizaWrite comes ready to run, no installation of the software is required and will run in 256K of RAM on one or more drives. Comes supplied with Workbench 1.2 and requires Kickstart 1.2 or greater.

■ Because VizaWrite is well behaved, it supports all peripherals, such as hard disk subsystems, as long as they are similarly well-behaved.

■ Automatically sets text into pages while editing. Text is always shown as "What you see is what you get."

■ Headers and footers show at the top and bottom of each page; they can be one or more lines and have their own font style and margins.

■ Ruler lines control page layout. Margins, tabulations, justification and line spacing are all adjustable using the icons on the ruler line. Rulers can be introduced anywhere in the document. Rulers can be hidden from view, if required.

■ Move between pages instantly: no waiting for disk accessing - select any page to work on.

■ Copy, cut and paste by highlighting text with the mouse.

■ Move around the text by pointing with the mouse or by using the cursor keys. Scroll through the document, forwards or backwards.

■ Edit and save any standard ASCII file. All character codes above the space character can be used in a document. Supports all international characters.

■ VizaWrite documents retain pertinent information when saved - such as author, creation date, notes, alteration count etc.

■ Open as many documents onto the screen as will fit into memory. VizaWrite uses memory extremely efficiently, and works with the AMIGA operating system in the standard way.

■ Mail merge from a standard ASCII file with configurable item delimiters. Alternatively, mail merge from a VizaWrite document, where each name and address is held in its own page.

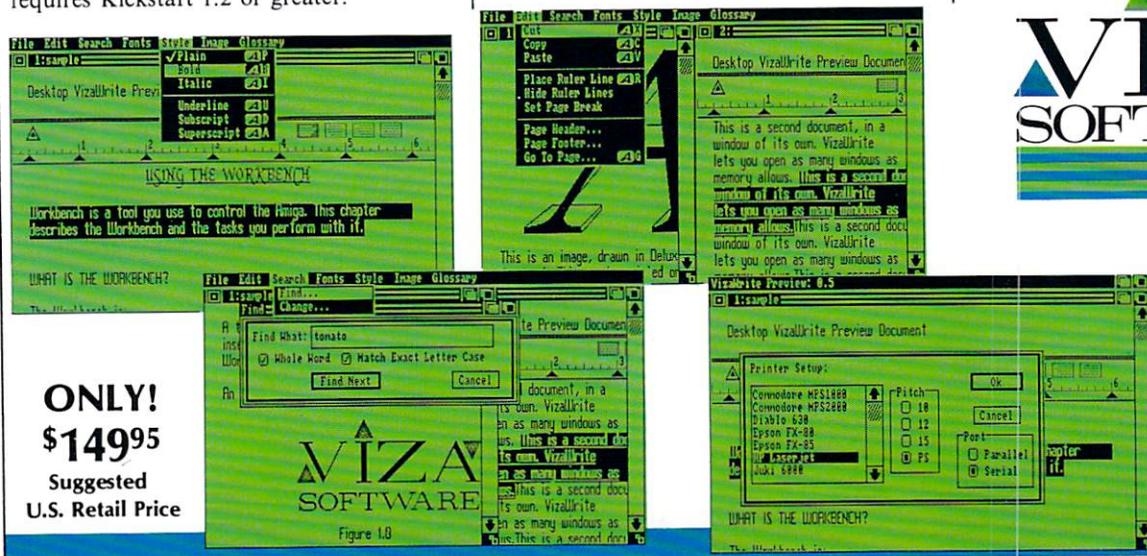
■ Optional configuration file permits the varying of many operation defaults, such as standard document size, screen colors, margins, tab stops, etc.

■ Document history window, used to log author's name, creation date, amended date, etc. Shows document statistics, such as word and sentence counts.

■ Glossary system permits single-keystrokes recall of frequently used phrases. This is inserted directly into the document at the current typing position, instantly.

■ Supports fixed-width font printing on any preferences-selected printer. Supports proportional printing on certain printer types only (this is a limitation of the printers). Recommended dot-matrix printer is NEC PINWRITER P6/P7, recommended daisywheel is JUKI 6100 or any DIABLO compatible. HP LaserJet is the recommended laser printer using the "F" font cartridge. The AMIGA proportional screen fonts are printed in high resolution on supported dot-matrix printers. Daisy-wheel users can use proportional print-wheels to print out documents laid out using the proportional screen fonts.

VizaWrite AMIGA now forms the nucleus of a complete desktop publishing system that Viza is developing for the Commodore AMIGA. Intuitive, simple, fast and powerful software - just what your AMIGA deserves.



**WHEN QUALITY COUNTS!**

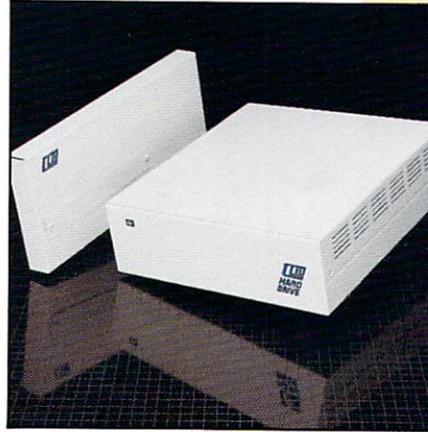
**PROGRESSIVE PERIPHERALS & SOFTWARE**

464 KALAMATH STREET  
DENVER, COLORADO 80204  
303-825-4144  
TELEX: 888837

**ONLY!**  
**\$149<sup>95</sup>**  
Suggested  
U.S. Retail Price

# 20-Meg SCSI Hard Drive \$999<sup>95</sup>

- Full AutoConfig
- Full Pass-Through out of Amiga expansion port
- Controller Supports 7 additional devices
- Internal Power Supply
- Faster than any comparably-priced drive



Create  
your own  
hard drive  
system

Buy Components  
Separately

Hard Drive only \$799<sup>95</sup>

SCSI Controller \$299<sup>95</sup>

## JetSet

Amiga Laser Printing  
Software \$69<sup>95</sup>

- Works with Hewlett Packard LaserJet™ or compatible laser printer
- Hundreds of Fonts available (starter typeface included)
- Works with Textcraft™ & Scribble™

## JetSet Fonts

\$49<sup>95</sup> to \$99<sup>95</sup>

Complete Typeface in each package (e.g. italic, bold italic, bold, demi-bold, regular in variety of sizes)

Selection Includes...Times ■ Triumvirate  
ITC Souvenir ■ Old English ■ Unical  
Commercial Script ■ Dom Casual  
ITC Benguiat Bold ■ Broadway  
Globe Gothic Outline ■ Borders  
Symbols ■ ITC Dingbats  
ITC Souvenir Greek/Math  
ITC Times Greek/Math. And many, many more.



aMEGA  
Board \$549<sup>95</sup>

Million Bytes  
of RAM

- Full AutoConfig Compatibility
- Works with all popular Amiga software

- Pass-Through for future expansion
- 6-Month parts & labor warranty

Available NOW at Amiga Dealers!

FEATURING  
over 1Meg of  
Rock, Sound & Graphics!



# ALIEN FIRES

2 1 9 9 A D

## TIME LORDS:

An immortal race of beings responsible for protecting the temporal balance of the universe from chaos and destruction.

Alien Fires uses every ounce of the Amiga's power to create a state-of-the-art visual and aural sensation, with colourful full-screen 3 dimensional computer graphics, digitized music and sound effects. It is an experience of incredible realism and sophistication that takes you through time and space to a new dimension in entertainment.

Play the part of a Time Lord in this imaginative new series of role-playing adventures. Alien Fires — Part I, 2199 AD, is now ready to take you into the distant future, in search of a great man and an awesome device capable of twisting the very fabric of time itself.

For the dealer nearest you,  
call toll-free:  
1-800-267-1904  
Dealer inquiries invited



Jaguar Inc.



## MetaScope: The Debugger

MetaScope gives you everything you've always wanted in an application program debugger:

- **Memory Windows**  
Move through memory, display data or disassembled code live, freeze to preserve display and allow restoration.
- **Other Windows**  
Status windows show register contents and program state with freeze and restore; symbol, hunk, and breakpoint windows list current definitions.
- **Execution Control**  
Breakpoints with repetition counts and conditional expressions; trace for all instructions or subroutine level, both single-step and continuous execution.
- **Full Symbolic Capability**  
Read symbols from files, define new ones, use anywhere.

- **Powerful Expression Evaluation**  
Use extended operator set including relationals, all assembler number formats.
- **Direct to Memory Assembler**  
Enter instruction statements for direct conversion to code in memory
- **and More!**  
Mouse support for value selection and command menus, log file for operations and displays, modify/search/fill memory, etc.

## MetaScribe: The Editor

MetaScribe has the features you need in a program editor:

- **Full Mouse Support**  
Use for text selection, command menus, scrolling — or use key equivalents when more convenient.
- **Multiple Undo**  
Undo all text alterations, one at a time, to level limited only by available memory.
- **Sophisticated Search/Replace**  
Regular expressions, forward/backward, full file or marked block.
- **Multiple Windows**  
Work with different files or different portions of the same file at one time.
- **Macro Programs**  
Lisp-like macro language lets you customize and extend the editor to meet your needs.
- **Virtual Memory**  
Set the amount of data memory to be used, transparently edit files larger than memory.
- **and More!**  
Keystroke macros for repetitive text, copy between files, block copy/paste/delete, set tabs and margins, etc.

Metadigm products are designed to fully utilize the capabilities of the Amiga™ in helping you develop your programs. If you're programming the Amiga, you can't afford to be without them.

## MetaTools I

A comprehensive set of tools to aid your programming (full C source included):

- **Make**  
Program maintenance utility.
- **Grep**  
Sophisticated pattern matcher.
- **Diff**  
Source file compare.
- **Filter**  
Text file filter.
- **Comp**  
Simple file compare.
- **Dump**  
File dump utility.
- **Whereis**  
File locator utility.

## DosDisk

A program that lets you access PC-DOS/MS-DOS™ diskettes on your Amiga. Use it to list file information and copy files between the PC-DOS/MS-DOS diskettes and Amiga diskettes or devices. Patterns can be used for file names, and you can even operate on all files in a directory at one time. A copy option converts source file line-end sequences as the copy is performed.

## Metadigm, Inc.

MetaScope  
\$95.00  
MetaScribe  
\$85.00  
MetaTools  
\$69.95  
DosDisk  
\$49.95

19762 MacArthur Blvd.  
Suite 300  
Irvine, CA 92715  
(714) 955-2555

(California residents  
add 6% sales tax).  
Visa/MasterCard  
accepted.

### Dealer Inquiries Welcome

Amiga is a trademark of Commodore-Amiga Inc.  
MS-DOS is a trademark of Microsoft, Incorporated

# Amazing Computing™

Your Resource to the Commodore Amiga™

Publisher: Joyce Hicks  
Circulation Manager: Doris Gamble  
Assistant to the Publisher:  
Robert James Hicks  
Traffic Manager: Robert Gamble

Managing Editor: Don Hicks  
Assistant Editor: Ernest P. Viveiros Jr.  
Hardware Editor: Ernest P. Viveiros  
Amicus & Technical Editor: John Foust  
Music Editor: Richard Rae  
Art Director: Keith Conforti  
Assistant Advertising Manager:  
John David Fastino  
Production Manager: Mark Thibault  
Assistant PM: Keven Desmarais  
Copy Editor: Michael Cabral

Advertising Sales  
&  
Editorial

1-617-678-4200

Amazing Computing™ (ISSN 0886-9480) is published by PiM Publications, Inc. P.O. Box 869, Fall River, MA. 02722. Subscriptions: in the U.S. 12 Issues for \$24.00; Canada and Mexico, \$30.00; Overseas, \$35.00. Printed in the U.S.A. Copyright© 1986 by PiM Publications, Inc. All rights reserved.

First Class or Air Mail rates available upon request.

PiM Publications, Inc. maintains the right to refuse any advertising.

PiM Publications Inc. is not obligated to return unsolicited materials. All materials requesting return must be received with a Self Addressed Stamped Mailer.

## MOVING?

Please advise PiM Publications Inc. at least four weeks before you move so your magazines will be delivered to your new address.

# Amazing Computing™

## Amazing Contents

Volume 2 Number 4

### *Amazing Features...*

<b>Amazing Interviews <i>Jim Sachs</i></b> by Steve Hull	9
The Amazing Artist who has become legendary in the Amiga community	
<b>The Mouse That Got RESTORED</b> by Jerry Hull and Bob Rhode	15
How to repair the micro-switches under the buttons in the Amiga mouse	
<b>Sleuthing Public Domain Disks with CLI</b> by John Foust	17
Take command of what exists below the Workbench	
<b>Highlights from the San Francisco Commodore Show</b> by Steve Hull	25
What's NEW -- in the world of Amiga™ computing	
<b>Speaker Sessions at the San Francisco Commodore Show</b> by Harriet Tolly	29
"...everything from highly technical talks to product overviews."	
<b>The Household Inventory System in AmigaBASIC™</b> by Bryan Catley	37
Maintain an inventory of your household possessions	
<b>Secrets of Screen Dumps</b> by Nathun Okun	59
Get those images 'In Print'	
<b>Using Function Keys with MicroEmacs</b> by Greg Douglas	67
Using the programmable keys is much faster than using the mouse...	
<b>AmigaTrix II</b> by Warren Block	71
More of those little shortcuts that make using the Amiga™ easier	
<b>BASIC Gadgets</b> by Bryan Catley	77
Create your own gadget functions!	

### *Amazing Reviews...*

<b>GRIDIRON!</b> by Keith M. Conforti	31
Real tackle football for the Amiga™	
<b>Star Fleet I Version 2.1</b> by John Tracey	35
"...advance in rank in the United States Galactic Alliance"	
<b>The TIC</b> by John Foust	57
A Tiny Battery-Backed Clock/Calendar for the Amiga™.	
<b>METASCOPE</b> by Harriet Tolly	81
An easy-to-use debugger by Metadigm	

### *Amazing Columns...*

<b>Roomers</b> by the Bandito	75
Pssst! Did you hear about....	
<b>The AMICUS Network</b> by John Foust	77
"The Commodore Show in San Francisco, GENie and MORE!!!"	

### *Amazing Departments...*

From the Editor	4
Amazing Mail	6
Public Domain Software Catalog	90
Index of Advertisers	96

# *From The Editor:*

Dear AC:

I purchased my Amiga a few months ago and (I) am a real beginner but (I) am working hard at reading and trying to learn. I also purchased several Public Domain software (disks) by Fred Fish and can not figure how to get into them. I have tried the little that I know of CLI but don't seem to get into them no matter what I use. I have some of the AMICUS that I can get into pretty well through the workbench. I would like to purchase more of them. Any advice for us beginners would be much appreciated. I enjoy the magazine, Amazing Computing™, and just took a subscription to it. Thank you for any help you can give me.

Very truly yours,

Gerry Hawkins  
Merritt Island, FL

The above letter is a sample of the mail we receive from Amiga users who are either confused about the Command Line Interface (CLI) or completely unaware of CLI. This is a severe problem, for unlike other machines, the Amiga has offered the user two interfaces to their machine.

Icons are convenient and fast, yet the Command Line Interface allows the user access to more tools on the Amiga. Through CLI, the user can see the "invisible" files, know the size of files, and work batch processes to fully customize their Amiga environment.

The problem with the CLI is Commodore almost completely ignored it in their standard release documentation. Users are propelled toward using third party documentation for a resident system which is available on every machine.

The only saving grace is the work some third party developers have done to make this documentation available. Bantam™ has not only printed the first and then the revised versions of the Commodore Business Machines' CLI documentation for developers, but they are also releasing an online help for CLI.

In that vein, please read John Foust's article "Sleuthing Public Domain Disks with CLI" on page 17. We hope this will clear away any problems not only with your experiences in the Public Domain library, but also add a new dimension to your Amiga understanding.

## **Jim Sachs**

We are very excited to have an interview with Jim Sachs, the Amazing Amiga Artist who was chiefly responsible for the artwork used in **Defender of the Crown**. Our biggest sorrow, with this issue, was our limited four color area did not allow us space to print many of Mr. Sachs brilliant images.

Jim Sachs is not only a great computer artist but a big fan of the Amiga. He does have one other contribution to the Amiga community, he wants users to delve into their machines and get the most from their computers.

Which brings me to the point of this editorial, this issue, and the magazine itself. Amazing Computing™ was developed to give EVERYONE a better understanding of the Amiga. It is a path to what is available for your machine. However, it is not the only path. Use your documentation, your friends, your user groups, and other third party documentation to expand your knowledge of this machine.

It is understandable that everyone cannot and does not want to be a programmer, however, there are a great many tools of which each user should at least be aware for those times we have a problem and can only rely on ourselves.

If you do have a problem, or you have a solution to a problem you have encountered, write us. We can all grow through the use of ability and communication.

Besides, it is fun.



Don Hicks  
Managing Editor

# Software designed for AMIGA®.

## **Lattice® C Compiler** \$225.00

New version 3.1 of the AMIGA DOS C Compiler replaces version 3.03. Major enhancements include the addition of: TMU, an assembler, a faster linker and version 3 MS-DOS.

With more than 30,000 users worldwide, Lattice C Compilers set the industry standard for MS-DOS software development. Lattice C gives you all you need for development of programs on the AMIGA. Lattice C is a full implementation of Kernighan and Ritchie with the ANSI C extensions and many additional features.

## **Professional Lattice® C Compiler** \$375.00

A new product called the Professional Lattice C Compiler is now available. It includes the C Compiler package (complete with TMU), plus LMK, LSE and the Metascope Debugger.

## **AMIGA® C Cross Compiler** \$500.00

Allows AMIGA development on your MS-DOS system. Price includes the Professional Lattice C Compiler described above.

## **Lattice Screen Editor (LSE™)** \$100.00

Designed as a programmer's editor, *Lattice Screen Editor (LSE)* is fast, flexible and easy to learn. *LSE's* multi-window environment provides all the editor functions you need including block moves, pattern searches and "cut and paste." In addition, *LSE* offers special features for programmers such as an error tracking mode and three Assembly Language input modes. You can also create macros or customize keystrokes, menus, and prompts to your style and preferences.

## **Lattice dBC III™ Library** \$150.00

The *dBC III library* lets you create, access and update files that are compatible with Ashton-Tate's dBASE system. *dBC III's* C functions let you extend existing dBASE applications or allow your users to process their data using *dBC III* or dBASE III.

## **Lattice Text Utilities (TMU™)** \$75.00

*Lattice Text Utilities* consists of eight software tools to help you manage your text files. GREP searches files for the specified pattern. DIFF compares two files and lists their differences. EXTRACT creates a list of file names to be extracted from the current directory. BUILD creates batch files from a previously generated file name list. WC displays the number of characters and optionally the checksum of a specified file. ED is a line editor which can utilize output from other *TMU* software in an automated batch mode. SPLAT searches files for a specified character string and replaces every occurrence with a specified string. And FILES lists, copies, erases or removes files or entire directory structures which meet the specified conditions.

## **Lattice Unicalc® Spreadsheet** \$79.95

*Unicalc* is a simple-to-operate program that turns your AMIGA computer into an electronic spreadsheet. Using *Unicalc* you can easily create sales reports, expense accounts, balance sheets, or any other reports you had to do manually.

*Unicalc* offers the versatility you've come to expect from business software, plus the speed and processing power of the AMIGA.

- 8192 row by 256 column processing area
- Comprehensive context-sensitive help screens
- Cells can contain numeric, algebraic formulas and titles
- Foreign language customization for all prompts and messages
- Complete library of algebraic and conditional functions
- Dual window capabilities
- Floating point and scientific notation available
- Complete load, save and print capabilities
- Unique customization capability for your every application
- Full compatibility with other leading spreadsheets
- Full menu and mouse support.

## **Lattice MacLibrary™** \$100.00

The *Lattice MacLibrary™* is a collection of more than sixty C functions which allow you to quickly and efficiently take advantage of the powerful capabilities of the AMIGA.

Even if your knowledge of the AMIGA is limited, *MacLibrary* can ease your job of implementing screens, windows and gadgets by utilizing the functions, examples and sample programs included with the package.

Other *MacLibrary* routines are functionally compatible with the most widely used Apple® Macintosh™ Quickdraw Routines™, Standard File Package and Toolbox Utility Routines enabling you to rapidly convert your Macintosh programs to run on the AMIGA.

## **Panel™** \$195.00

*Panel* will help you write your screen programs and layer your screen designs with up to ten overlapping images. *Panel's* screen layouts can be assigned to individual windows and may be dynamically loaded from files or compiled into a program. *Panel* will output C source for including in your applications. A monitor and keyboard utility is also included to allow you to customize your applications for other systems.

With Lattice products you get *Lattice Service* including telephone support, notice of new products and enhancements and a 30-day money-back guarantee. Corporate license agreements available.



# Lattice

Lattice, Incorporated  
Post Office Box 3072  
Glen Ellyn, Illinois 60138  
(800)533-3577 In Illinois (312) 858-7950  
TELEX 532253 FAX (312) 858-8473

INTERNATIONAL SALES OFFICES: Benelux: Ines Datacom (32)2-720-51-61  
Japan: Lifeboat, Inc. (03)293-4711 England: Roundhill (0672)54675  
France: Echosoft (1)4824.54.04 Germany: Pfothenaur (49)7841/5058  
Hong Kong: Prima 85258442525 A.I. Soft Korea, Inc. (02)7836372  
Australia: FMS (03) 699-9899 Italy: Lifeboat Associates Italia (02) 46.46.01

Amiga is a registered trademark of Commodore-Amiga, Inc.

# Lattice

# Amazing Mail:

Dear AC,

I wish to thank you for your recent articles on Modula-2, the Pascal style language. I am a recent user of Modula-2 and am excited about its power and ease of understandability. Writing applications software is coming easier to me, but writing systems software is another story. I'm still in a fog; but learning.

I would hope to see more articles on programming in Modula-2, with examples for someone like me to learn from and add to my library. Maybe, even a monthly column devoted to the popular languages used on the Amiga.

Thank you for your fine magazine. It doesn't just tell us about the Amiga, but shows us how to use it.

An Amiga user,  
Tom McGrath

P.S. Maybe you could publish a list of any Modula-2 users groups you know of.

*Thank you for the praise, however, AC is very fortunate to have a group of dedicated Amiga users as writers. We have had several people interested in Modula-2, but no permanent columns so far. Perhaps we will receive a response to this letter.*

Dear Amazing Computing:

I just picked up my Volume 2 Number 2 issue of Amazing Computing and was excited to see that you had a nation wide list of BBSs. Since I run an Amiga-oriented BBS and we have been up for well over a year, I was eager to see if we were listed.

Yes, indeed we were, however we were listed as "Stonehenge" and our hours were shown as "5P-8A", "24hrs Wkds".

Since both of these facts are slightly in error (I have helped maintain BBS lists before; I understand the difficulty) I thought I would write and give you the straight skinny.

When we went up over a year ago we were limited hours, but since January last year we have been 24x hours.

Our BBS is called XAKEPOBO MECTO (pronounced Hackerovo Miesto and Means "Hacker's Place" in another language) or XAK("zak") for short. It is a room-based BBS program called Stonehenge which runs on IBM PCs and compatibles (like the sidecar and A2000 with bridge) and we are one of 9 Beta test sites. Stonehenge is a networking BBS and all the Alpha, Beta and purchased sites participate in the net. Most of us are in the Seattle, Bellevue, Everett area however we have one site in southern California and one going in in Canada.

Each room in the BBS System is a list of messages pertaining to the subject of the room (such as Generic C, Amiga Info, etc.). Each room can be attached to a directory on the disk for up/down loading and each room can be attached to another room on another system either direct or through any system "between" them (in other words, each of the 64 possible rooms is much like most systems in their entirety). Within the system, the rooms can be grouped into hallways, (we have a General topic hallway, Amiga u/dload, Graphics, etc.).

The networking is accomplished via incoming and outgoing calls made automatically by the systems that are networking, on a regular interval. For local calls, we use a 2 hour interval to keep message latency down and conversation in continuum. For the farther sites, (like California) we usually keep it down to between a few times to once a day.

Files can be mailed across the net and there is internodal communications (like you can ask any BBS, from any other BBS, what version it is and the version will be returned in a message to you >from< the BBS without the need to bother the Sysop).

Our system is 24 hours 300/1200 baud. We are in Seattle (even though my address is Tacoma which is nearby) and our number was correct (206) 324-0830.

We are currently running on a PC clone with a 30 megabyte hard disk drive for upload/download which operates close to full. Many public domain disks are online and others are available upon request.

I hope this has been of help to both you and your readers (to include myself). You get great reviews on our system.

Sincerely,  
Michael Aichlmayr

*Good Luck!*

Dear Amazing Computing,

I would like to express my gratitude for the fine job you are doing on your magazine. I am not a subscriber, but I have every issue. You are my perfect excuse to visit the nearest Amiga store.

I have a request to make of your C programmers. I have been trying unsuccessfully to activate the cursor in a string gadget. I have spent several hours on CompuServe and have read every book I can get. Several books acknowledge that it can be done and also give the basic idea, but no one has given a demo. Somebody probably has a demo, but I have exhausted all of the avenues that I know to try. I would appreciate an article on string gadgets. Also I would welcome a discussion of an ISAM file in 'C' specific to the Amiga.

Any help you can give will be appreciated. Thanks again for all of the hours of enjoyable and informative reading.

Sincerely,  
Dennis Reese

*Read Harriet Tolly's article in AC issue V2.3 "Intuition Gadgets". I hope this will answer your questions.*

# AVAILABLE NOW! StarBoard2

If you've owned your Amiga® for a while now, you *know* you definitely need more than 512k of memory. You probably need *at least* double that amount...but you might need as much as an additional two megabytes.

We want to urge you to use **StarBoard2** as the solution to your memory expansion problem—and to some of your other Amiga-expansion needs as well!

## It's small, but it's BIG—

Since most of you want to expand your Amiga's memory without having to also expand your computer table, we designed **StarBoard2** and its two optional "daughterboards" to fit into a sleek, unobtrusive Amiga-styled case that snugly fastens to your computer with two precision-machined jackscrews.

The sculpted steel case of **StarBoard2** measures only 1.6" wide by 4.3" high by 10.2" long. You can access the inside of the case by removing just two small screws on the bottom and pulling it apart. We make **StarBoard2** easy to get into so that you or your dealer can expand it by installing up to one megabyte of RAM on the standard **StarBoard2** or up to two megabytes by adding in an Upper Deck.

## This card has decks!

The basic **StarBoard2** starts out as a one megabyte memory space with 0k, 512k, or one megabyte installed. If you add in an optional **Upper Deck** (which plugs onto the Main Board inside the case) you bring **StarBoard2** up to its full two megabyte potential. You can buy your **StarBoard2** with the Upper Deck (populated or unpopulated) or buy the Upper Deck later as your need for memory grows.

And you can add other functions to **StarBoard2** by plugging in its second optional deck—the Multifunction Module!

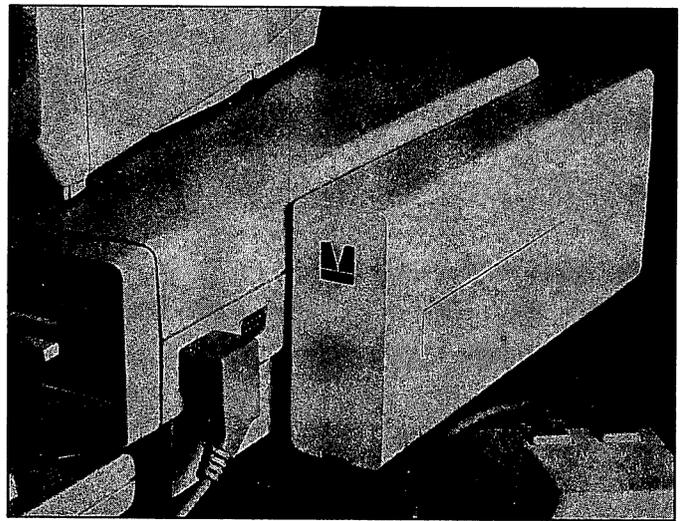
## StarBoard2: functions five!

If we count Fast Memory as one function, the addition of the **MultiFunction Module** brings the total up to five!

### THE CLOCK FUNCTION:

Whenever you boot your Amiga you have to tell it what time it is! Add a **MultiFunction Module** to your **StarBoard2** and you can hand that tedious task to the battery-backed,

**Auto-Configuring  
Fast RAM  
Zero Wait States  
User Expandable  
from 512k to  
2 Megabytes  
Bus Pass-Through  
MultiFunction  
Option: battery/  
clock, FPU,  
parity, Sticky-Disk**



real-time clock/calendar. A small piece of MicroBotics software in your WorkBench Startup-Sequence reads the clock and automatically sets the time and date in your Amiga. And the battery *is* included (we designed it to use an inexpensive, standard AAA battery which will last at least two years before needing replacement).

### THE FLOATING POINT FUNCTION:

If any one aspect most characterizes the Amiga it's *fast* graphics! Most graphic routines make heavy use of the Amiga Floating Point Library. Replacing this library with the one we give you with your **MultiFunction Module** and installing a separately purchased Motorola 68881 FPU chip in the socket provided by the Module will speed up these math operations from 5 to 40 times! And if you write your own software, you can directly address this chip for increased speed in integer arithmetic operations in addition to floating point math.

### THE PARITY CHECKING FUNCTION:

If you install an additional ninth RAM chip for every eight in your **StarBoard2**, then you can enable *parity checking*. Parity checking will alert you (with a bus-error message) in the event of any data corruption in **StarBoard2**'s memory space. So what good is it to know that your data's messed up if the hardware can't fix it for you? It will warn you against saving that data to disk and possibly destroying your database or your massive spreadsheet. The more memory you have in your system the more likely it is, statistically, that random errors will occur. Parity checking gives you some protection from this threat to your data residing in Fast RAM. Note that the Amiga's "chip" RAM cannot be parity checked.

### THE IMMORTAL MEMORY DISK FUNCTION (STICKY-DISK):

When you've got a lot of RAM, you can make nice big RAM-Disks and speed up your Amiga's operations a lot! But there's one bad thing about RAM-Disks: they go away when you re-boot your machine. Sticky-Disk solves that problem for you. It turns all of the memory space inside a single **StarBoard2**

into a Memory Disk that will survive a warm-reboot! When your Amiga attempts to grab a **StarBoard2** in Sticky-Disk mode, a hardware signal prevents the system from acquiring the **StarBoard2** as FastRAM (and thereby erasing your files)—instead it is re-recognized as a Memory Disk and its contents are preserved intact. If you want to work rapidly with large files of data that are being constantly updated (such as when developing software) you can appreciate the Sticky-Disk!

## Fast RAM —no waiting!

**StarBoard2** is a *totally* engineered product. It is a ZERO WAIT-STATE design, auto-configuring under AmigaDOS 1.2 as Fast RAM. Since AmigaDOS 1.1 doesn't support autoconfiguration, we also give you the software to configure memory in 1.1.

Any applications software which "looks" for Fast RAM will "find" **StarBoard2**. And you'll find that your applications run more efficiently due to **StarBoard2** on the bus.

## A passing bus? Indeed!

What good is an Expansion Bus if it hits a dead end, as with some memory cards? Not much, we think—that's why we carefully and compatibly passed through the bus so you could attach other devices onto your Amiga (including another **StarBoard2**, of course!).

## The sum of the parts...

A really nice feature of the **StarBoard2** system is that you can buy exactly what you need now without closing off your options for future expansion. You can even buy a 0k **StarBoard2** (with a one megabyte capacity) and populate it with your own RAM (commonly available 256k by 1 by 150ns memory chips). When you add **StarBoard2** to your Amiga you have a powerful hardware combination, superior to any single-user micro on the market. See your Authorized Amiga Dealer today and ask for **StarBoard2**

SUGGESTED RETAIL PRICING:	
StarBoard2, 0k (1 meg space):	\$349
StarBoard2, 0k (2 meg space):	\$395
StarBoard2, 512k (1 meg space):	\$495
StarBoard2, 1 meg (1 meg space):	\$595
StarBoard2, 2 megs installed:	\$879
StarBoard2, 2 megs & MultiFunction:	\$959
Upper Deck, 0k (1 meg space):	\$ 99
MultiFunction Module:	\$ 99
<i>also available:</i>	
Standard 256k memory card:	\$129
MAS-Drive20, 20 meg harddisk:	\$1495
MouseTime, mouseport clock:	\$ 50



**MicroBotics, Inc.**

811 Alpha Drive, Suite 335, Richardson, Texas 75081 / (214) 437-5330

AMIGA is a registered trademark of Commodore-Amiga

Dear AC,

Referring to my letter about the article "A Megabyte Without Megabucks". I would just like to thank you for releasing an article that really improves upon the state of Amiga user's willing to risk some hardware hacking on their beloved Amigas.

I wrote you a previous letter, stating a bit of disapproval with the article. Most of the letter was a frustrated release of my own immaturity in the area of hardware.

I took my machine to a friend who works with hardware as a professional and after a few hours of redoing my work, all of the memory was available to me.

With a few tips on soldering technique, I was able to get the Kickstart V1.2 hardware modification assembled and properly working.

At the time, I didn't have a modem, so I wrote my own AddMem routine. Trying to decrypt enough useful information from the Manuals by Addison Wessley was quite a chore. But, I knew I had succeeded when Avai and GfxMem showed that I had 512K of FastMem available to me.

Presently, I've got a 3 meg system (1meg internal, 2meg external on a MegaBoard 2). A few calls using a borrowed modem has left me with some powerful software (Ray Tracing Package, Terminal Programs, Useful Utilities), justifying the availability of my 3 meg.

Keep up the great work!

Thanks,  
David Sidlo

*I am glad you made the corrections through a friend. Several other readers have had problems, but were not able to find a handy person to straighten their work. TI hope this letter will give them the drive to complete this project.*

Dear AC,

In the March 1987 issue of *Sky & Telescope* magazine, the regular "Astronomical Computing" column, conducted by Roger Sinnott, was devoted to "Astronomical Software Benchmarks".

The Savage benchmark, described in *Byte* magazine in 1985 (Vol. 10, No. 11, page 67), was run on a range of computers, from a Z80A processor with a 4 megahertz clock speed to a Cray X-MP/24 supercomputer. The results reported by T.S. Kelso, I found to be of considerable interest. This benchmark was selected because it is similar to much of the computing done by amateur (and professional?) astronomers, for locating positions of stars, planets and the like. It consists of 2499 iterations of three pairs of complementary functions: tangent & arctangent, exponential & logarithm, and square & square root. The results should be 2500, and will differ from 2500 only because of roundoff errors in the computation of the functions. The extent of the accumulated error after the 2499 iterations gives a good idea of the accuracy of the computer's handling of the functions, and the time taken is also of some interest, especially if you plan to use the computer to control a telescope-pointing mechanism.

How does the Amiga computer do in these tests? I ran them (LISTings attached) on my Amiga, which is a standard off-the-shelf machine using Microsoft's interpreted AmigaBASIC version 1.2, and obtained these results:

Single-precision AmigaBASIC gave 2769.853 in 34 seconds, a poor result (it should be 2500), but no worse than the single-precision results obtained by Kelso in the tests on a Z80A processor.

Double-precision AmigaBASIC gave 2499.999999682243 in 73 seconds. This tiny error is equivalent to a clock running slow or fast by one second in 249 years. Kelso using a double-precision BASIC, obtained 2499.999999869946, equivalent to an error of one second in 595 years.

Kelso's result is about twice as accurate as mine, but Kelso's compiled BASIC took 38 minutes 22 seconds, thirty-one times longer than my interpreted BASIC! In the test, the Amiga looks pretty good.

Yours truly,  
Ivan C. Smith

' accuracy and speed test  
' see *Sky & Telescope*  
' March 1987, page 309  
' for Amiga computer  
' single-precision version  
' Ivan Smith, P O Box 226  
' Chester, Nova Scotia  
' Canada B0J 1J0

```
DEFINT i
DEFSNG a
PRINT "Start "TIME$
a=1!
FOR i=1 to 2499
a=TAN(ATN(EXP(LOG(SQR(a*a)))))+1!
NEXT
PRINT a
PRINT "Finish "TIME$
```

' accuracy and speed test  
' see *Sky & Telescope*  
' March 1987, page 309  
' for Amiga computer  
' double-precision version  
' Ivan Smith, P O Box 226  
' Chester, Nova Scotia  
' Canada B0J 1J0

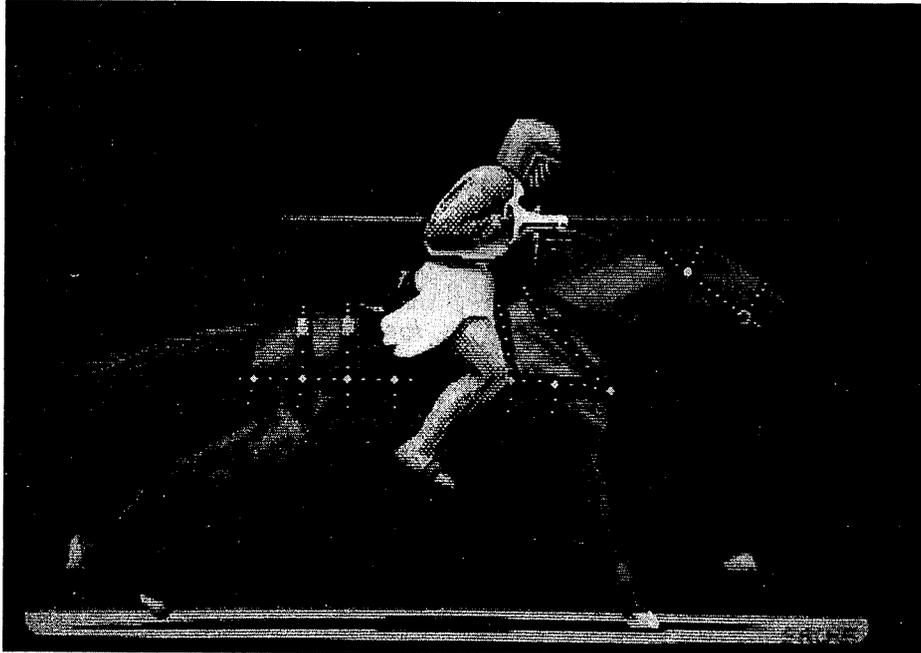
```
DEFLNG i
DEFDBL a
PRINT "Start "TIME$
a=1&
FOR i=1 to 2499
a=TAN(ATN(EXP(LOG(SQR(a*a)))))+1&
NEXT
PRINT a
PRINT "Finish "TIME$
```

*Great! But remember, some benchmarks can be more of a test of the programmer than the computer.*

•AC•

## Amazing Interviews...

# Jim Sachs



*"His graphics work on the best selling Amiga game to date, **Defender of the Crown**, has numbered Jim Sachs among that small but honored fraternity of "superstar" Amiga developers."*

*by Steve Hull*

I remember the first time I saw Jim Sachs' Amiga art. It was May 1986 and my Amiga had only been part of the family for two weeks. I had heard a lot about the Amiga's graphics capabilities, but it was really the sound that sold me. Graphically, the dealer hadn't shown me a single thing on the Workbench Demos disk that my old Atari 800 couldn't do with a few display-list interrupts. Even the bouncing ball demo was making the rounds on the 8-bit Atari bulletin boards. A little disappointing after all the hype, I thought.

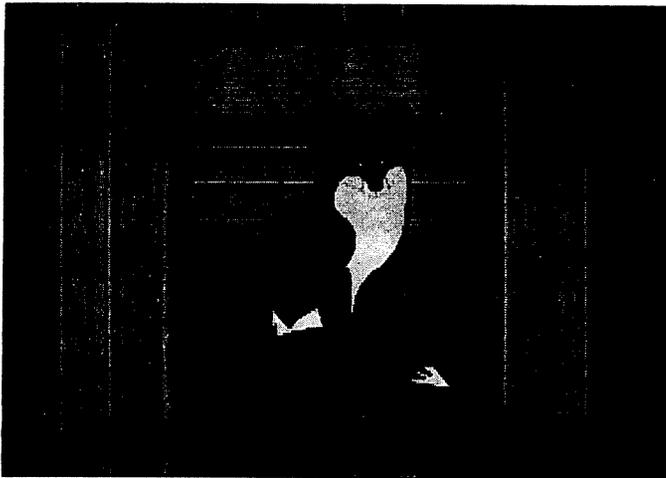
All that changed the night I downloaded an archived file of Jim Sachs' artwork from Virginia's Empire BBS. My jaw dropped as I beheld, for the first time, pictures that have since become legendary in the Amiga community: the exquisitely rendered **Sachs Castle 1990**; the finely-textured woodcut, **Foxpuppy**; and of course the **Porsches**. And all these were done in *low-res*??? In one download, I became a believer in Amiga graphics.

Jim Sachs is a former Air Force pilot who left the service when "career enhancement" demanded he trade his worldwide cargo routes for the ordinary life of an instructor pilot in Lubbock, Texas. In 1984, aware that the home computer revolution was quickly leaving him behind, he bought his first Commodore 64 for \$360 "I was sure the price would never go below that," he laughs.

Within eight months he had taught himself machine language and had produced his first commercial effort, **Saucer Attack**. A multi-staged shoot-em-up set in Washington D.C., Saucer Attack featured better graphics than some Amiga games. Unfortunately, software pirates nibbled away his profits; he describes the title as "the game every kid has and nobody bought."

Sachs spent another year working on what he describes as "the ultimate C-64 game", a time travel fantasy called **Time Crystal**. The rampant pirating, throughout the younger

*continued...*



C-64 user community, caused him to decide against marketing *Time Crystal*; nonetheless his public-domain demo continues to fuel rumors of the game's release wherever C-64 users congregate.

His graphics work on the best selling Amiga game to date, *Defender of the Crown*, has numbered Jim Sachs among that small but honored fraternity of "superstar" Amiga developers. His work has appeared on the covers of every Amiga magazine as well as many of those not Amiga-specific. His *Mazda* artwork is virtually a trademark for Aegis Images. And you would be hard-pressed to find an Amiga dealer that *doesn't* have a Sachs color screen dump proudly tacked to the wall.

Today, Jim Sachs lives in a mountain resort community in southern California, thousands of feet above the smog line, in a partially completed cabin immediately recognizable as *Sachs Castle 1990*. His wife Edi is a personable, understated lady who accepts her husband's computer addiction with good humor.

A tireless advocate for the Amiga, Sachs is as likely to turn up at a small users' group meeting, sharing his secrets, as he is to be seen at a major exhibition, promoting his newest projects. Though proud of such accomplishments as being named *The Guide to Commodore Computing's* Computer Artist of the Year, Sachs is likelier to bring up a more recent collaboration between he and Edi - a healthy new baby girl, Jill Marie.

Sitting in that portion of *Castle 1990* that is complete in 1987, I began by asking Jim the question everyone asks when they first see his work...

**AC:** How did you *ever* learn to draw like that?

**JS:** I have no formal training, but I've pretty much always been involved with art since I was a kid. It turns out that the kind of art that I'm good at - and the kind that I like to do - translates very well to the computer screen.

**AC:** How so?

**JS:** Detail, contrast - mostly contrast. That's where most of the people fall down in trying to draw on a computer screen. That's because it's the opposite of how you would do it on paper. On paper, you start with something that's white and put down darker colors. With the computer screen it's just the opposite; you start out with a black screen and paint with points of light.

It just turned out that my interpretations of light and shadow and detail happen to turn out really well on a computer screen. I don't know how to describe it. You should have talked to one of the guys I had working for me, Steve Quinn - he's working for Cinemaware now. He was doing some pretty good work on the Amiga - in fact, some really outstanding work - but he could never get the contrast.

Finally, he took a step back and did all of *Defender of the Crown* on the Macintosh. He did every screen that appears in the released version. They ported it over from my version, and then Steve had to completely redraw it - starting with *straight* black and *straight* white. Then he had to dither everything in between to make all the different shades and tones. He came up yesterday just *thanking* me for the education; it was a total revelation to him. And now his Amiga work is so much better, because he can think in terms of contrast now, and not just color.

**AC:** You virtually set the state-of-the art for Amiga graphics with *Defender of the Crown*.

**JS:** You know, I didn't do *all of every scene* in that whole game. My job was to create graphics, but also to ensure a certain continuity. Much of the artwork handed in to me needed to be cleaned up, in order that all the graphics would *look* like they were done by the same person. It's all a matter of style.

**AC:** And attention to detail.

**JS:** That's right. The details I include are things that most people will never notice, but if the details weren't there they would. Things like the shading on the title screen of *Defender*; if the light wasn't coming from a certain direction and casting the shadow of the brass flowing down across the rocks, it would really look flat.

**AC:** How do you feel about the game?

**JS:** We had a lot more planned for it, but a lot fell by the wayside in order to get the program on the market promptly. Cinemaware was getting an incredible amount of interest on the product, and Christmas was coming up.

**AC:** You can't fault them for wanting to make money back on their investment.

**JS:** No, you can't, and we were really breaking some new ground here. Nobody knew the kind of time this sort of project would require. It was, in many ways, just like producing a movie - there were the factors of coordinating script writers, graphics, music composing and so on. All told, I think we did pretty well, though a little more time would have allowed us to put more into the gameplay.

At this point, Sachs takes me on what could be called a guided tour of "the cutting room floor" - those portions of *Defender of the Crown* that, due to time or disk space, were omitted from the finished game. Called to the joust, a knight rides off to battle in a full-screen clear wash superimposed over the island, a red arrow tracing his path across the map.

Heralding the contest, the trumpeters' chests rise and fall as they play the fanfare, while pennants snap crisply in the breeze. The joust begins, and the view cuts from the wide overhead panorama to ground level intermediate views of each contestant and his steed, alternating between each other as they close in ("That's five weeks of work right there," Sachs observes.). A boulder, catapulted short of the top of the castle wall, impacts against the side, spreading a spiderweb of cracks.

The most breathtaking moment occurs when Sachs lobbs a firepot over the castle wall; there is a pause, and then flames appear, licking up the sides of the castle and consuming the massive oak drawbridge. It is an impressive commentary on *Defender* that even the rejected portions of the game are stronger than many released titles.

**AC:** I can understand how you feel about all that was cut out; even so, the game is still a striking effort.

**JS:** Well, it's hard to argue with success. We sold 20,000 copies of *Defender of the Crown* in the first six weeks, and for a machine with a user base of about 150,000, that's phenomenal.

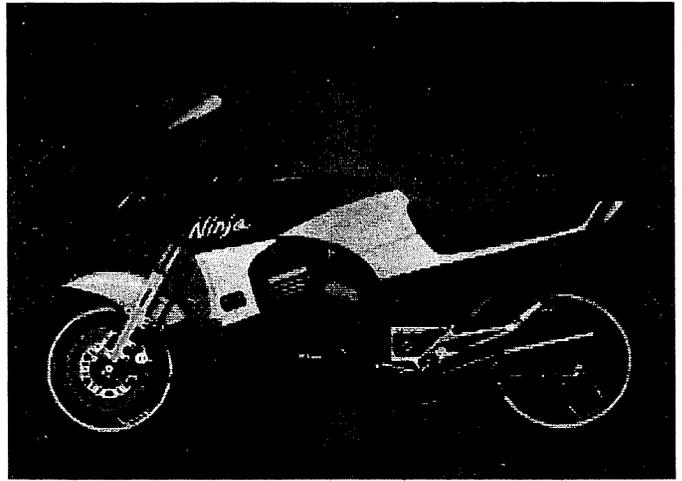
**AC:** One of your first pictures that made the rounds was an advertisement for an Amiga version of *Saucer Attack*. Are you still planning on releasing *Saucer Attack*?

**JS:** I was going to port it over - it was going to be my first project. In the '64 version you shot at the saucers against a flat background of Washington, D.C. The Amiga version would have featured a 3-D environment; you'd actually drive through the streets with the buildings coming at you. The saucers wouldn't just be flying over the buildings but *over your head* - you'd have to back up to get 'em...but right now I'm just too swamped with other work.

**AC:** So *Saucer Attack* for the Amiga isn't going to see the light of day?

**JS:** Probably not. Not unless I happen to have a whole lot of time left over. I would have to do it under my own label and it doesn't pay to have things under my own label anymore. Going independent, I'd have to pay for all the advertising. On the '64 version of *Saucer Attack* I made \$30,000 but I spent \$27,000 in advertising, so it wasn't worth it - except to gain a name.

I was able to fly to Commodore in Pennsylvania and grab Paul Goheen, who was head of Commodore software at that time, sit him down in front of a monitor and say, "This is what I can do, give me developer's status on the Amiga," and he did. So it was worth going through that, and eleven months



worth of work on *Time Crystal* - a product that's never going to come out - because it got me Developer's status on the Amiga in time to get a jump on other artists.

**AC:** How does programming on the Amiga compare with the C-64?

**JS:** There isn't any comparison. To design the background for *Saucer Attack* I had to draw the design on graph paper and then enter it into the computer as a series of hexadecimal numbers. That screen took about three months, putting it in a pixel at a time, not being able to see what I was doing. Designing graphics on the Amiga is just a dream. It's fantastic being able to see what I'm doing.

**AC:** How about machine-language programming?

**JS:** I'm on a crusade right now to get people to ignore Intuition and the operating system and *get right down into the heart of it*. That's why I was able to squeeze all of that performance out of the '64, because I was writing directly to the hardware registers. The *Time Crystal* demo uses the whole 64K - the place where BASIC used to live is all my own code, the place where the Kernal used to live, everything except for the last two bytes in the whole computer is all my own programming.

That's why the Atari people are able to make the ST *look* like the Amiga - they're all former '64 people that're used to really getting down and dirty with it, and they get in there and use the microprocessor to do the things that the Amiga does just loafing along, letting the Copper and the Blitter do.

I've seen some *fantastic* screen demonstrations done on the ST and nobody has bothered to do that kind of thing on the Amiga because it's so *easy* not to - you become complacent, you just access all its built-in routines for doing these things, and you *just can't do that* if you want to beat the competition!

**AC:** Speaking of the competition, why did you choose to go with the Amiga rather than the Mac, which was firmly established -

*continued...*



JS: It's black and white.

AC: - or the ST?

JS: The ST came out when I already knew about what the Amiga was going to be, and even though I hadn't seen an Amiga yet I had read so much about it that I knew it was what I needed. I knew from the beginning that the ST was designed as low-priced competition for the Amiga. Well, I wanted the *real* one. The '64 was state of the art when I bought the '64. The Amiga was state of the art when I bought the Amiga. I wasn't *about* to write for less than state-of-the-art.

The Amiga is the first step in a process of hardware evolution that will eventually - I would say within the next ten years - allow you to sit down at your desk, and do a movie like *Star Wars*, or anything that George Lucas or Steven Spielberg can do for 40 million dollars right now. You'll be able to sit down at your home computer and do *everything* - the sound, the music, the artwork - the story, that would rival a Hollywood production.

AC: Rumors have already started to leak out about your latest "Hollywood production" - tell us about it.

JS: Right now I'm producing *20,000 Leagues Under the Sea* for Aegis. We had hoped to get Disney interested in buying-in, but so far they haven't been very cooperative, so I'm taking most of it directly from the book.

AC: What will that mean for the game?

JS: There are a few things I can't use. The island of Volcania - Captain Nemo's home base - is a Disney invention. In the book, Captain Nemo *has* no home base. He destroyed the place where he built the Nautilus as soon as the Nautilus was done. The book is also more realistic; in the movie Nemo escapes from a slave camp by stealing one of their ships with a few of the other slaves. And with no other resources than that, he builds the entire submarine.

AC: Out of coconut leaves?

JS: In the book it's a lot more logical. Captain Nemo was really rich - he had a fleet of ships - and he went to different countries, contracting with major manufacturers all over the world for little parts of the ship, so nobody knew exactly what he was making. He just gathered all the parts, took them to his desert island and put it together.

AC: What kind of graphics are you working on?

JS: I'm working on routines that will let you come up to a ship and sail all the way around it, and see the whole process in detailed 3-D. I'm working on some ship models right now - I figure I'll need 19 different views to do what I'm trying to do.

AC: Let me see if I understand what you're saying. 20,000 Leagues will have the 3-D graphics of Arctic Fox with the detail of Defender of the Crown?

JS: Exactly. One of the things I have to do - in fact, one of the *hardest* things, it's never been done before, is to write a "shrink" routine that will take an object and shrink it in real-time to any particular size you want at that instant. The only view actually in the computer is the large view with the full detail. Then for any particular moment, you don't see *that* view, you see that view **compressed** for whatever depth you are from the object at the moment.

I need a routine that starts with the object and removes certain lines of code that I predefine - for instance, some of the detail in broad areas that you could afford to lose. A line here and a line there, and it shrinks a little bit at a time. I was originally going to write the routine to do the 3-D version of Saucer Attack, so that the entire city would appear off in the distance and as you advanced, the buildings would not only get bigger but actually gain detail, instead of losing detail - 'cause the program would be putting the detail *back in*. So now I'm finally getting around to having to do the routine.

AC: You've banked a lot of your future on the success of the Amiga - so how does it feel, witnessing a Commodore marketing plan that some have described as "Ready...fire!...aim..."?

JS: I think they're getting it together. Commodore has **finally** taken the hint from guys like me and everybody who's been on a crusade to change their marketing strategy. They were going after the IBM market and it's a *mistake*; they were getting *killed*. This is the **greatest creative tool that's ever been designed**, and they couldn't see it - they were marketing it to do spreadsheets and word processing, trying to replace IBMs in peoples' offices and they're never going to do that. IBM is too entrenched. Here's a machine that can do things no other machine ever has been able to do, and they were *blind* to it - they didn't use it in their advertising.

AC: What would you have used?

JS: Graphics and sound. *That's* what people buy this machine for. That's where the heart and soul of this thing is. Amiga always could see it - the people who designed it could

*continued...*

# Big Ram Expansion

## FOR YOUR AMIGA®



- 2 Megabytes - easily expandable to 8 megabytes
- Fully self contained, including 100 watt power supply.
- Full speed, no wait state operation.
- Pass through bus connector.
- Automatically "Auto-configs" under Kick 1.2

Our RAM-BOards™ are delivered to you with 2 to 8 megabytes of RAM (memory) installed. Each board is equipped with it's own power supply and is **EASILY EXPANDABLE** to the full **8 MEGABYTES**. Our RAM-BOards™ sit comfortably on top of your AMIGA® computer and cause no interference with other attached devices.

- **RAM-BOard-2™** 2 megabytes installed **\$599.95**  
Add \$380 per 2 megabytes of additional memory installed at the time of original purchase.
- **RAM-EXP-2™** 2 megabytes of additional expansion RAM for installation on RAM-BOard-2 **\$399.95**
- **T-UTIL™** **\$49.95**  
Disk and file repair utility software for your AMIGA™ computer system
- **T-MOVE™** **\$49.95**  
Software to move files to and from IBM-PC™ format disks (either 3.5" convertible format, or 5.25" standard format)



# TechniSoft



Post Office Box 7175 5505 Walden Meadows Drive  
Murray, Utah 84123 (801) 268-4961

All continental U.S. software orders add \$2.50 shipping/handling.  
RAM-BOard orders add \$7.50 shipping/handling.  
COD add an additional \$2.50 per order.  
5 year warranty covers RAM installed at the time of original purchase. Your RAM-BOard™ works for 5 years, or we fix it, free of charge.

RAM-BOard, T-MOVE, T-UTIL are trademarks of Techni Soft.  
IBM-PC is a trademark of International Business Machines, Inc.  
AMIGA is a registered trademark of COMMODORE-AMIGA, Inc.

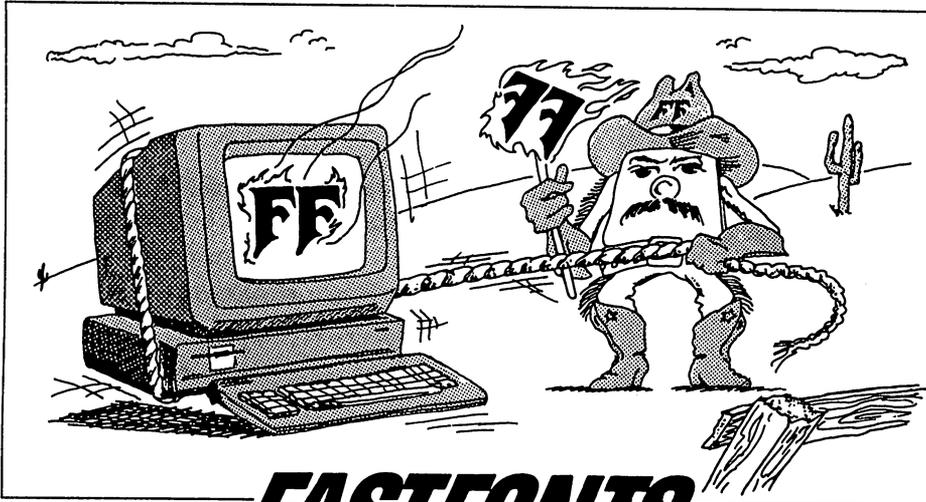
see it - then of course, they were all laid off. R.J. [Mical] knew that. Dale Luck and all the other people that designed the thing, but Commodore was absolutely blind to it. Now that they've started changing their marketing strategy they're selling really well. Defender of the Crown seems to have helped them, too.

**AC:** Do you have any advice for aspiring Amiga artists and developers?

**JS:** Bulletin boards are a great way to become known. There are also users' groups, the Fred Fish disks, and the AMICUS™ Network. If somebody is really good and has a unique style on the Amiga, it's no problem getting noticed. I'm sure the market for artists is going to *skyrocket* now that we've pretty much increased the stakes.

The manufacturers are not going to be able to continue to put out Commodore 64 port-overs like they have been. They won't sell up against games like Defender of the Crown and some of the projects I'm working on - and I'm sure other people are working on too. *Everybody's* going to be starving for artists really soon. I would say that getting your work on the bulletin boards and on public-domain disks - at users' groups and dealers - would be the way to get noticed.

•AC•

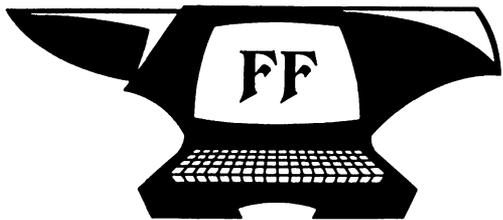


# **FASTFONTS**

## from the creators of TxEd.

- ★ FASTTEXT routines speed up text display — works with your existing programs!
- ★ Replace the System TOPAZ Fonts with one of four fonts we supply — or with one of your own.
- ★ FunKeys hotkey program lets you move windows, program macros, or create a CLI at any time with a single keystroke.
- ★ ScreenBlanker protects your display from damage.
- ★ TxEd V1.3 is still available for only \$39.95

Boulder Font: ABCDEFGHIJKLMNOPQRSTUVWXYZ  
 Gulch Font: ABCDEFGHIJKLMNOPQRSTUVWXYZ  
**WESTERN FONT:** ABCDEFGHIJKLMNOPQRSTUVWXYZ  
 Siesta Font: ABCDEFGHIJKLMNOPQRSTUVWXYZ



**MICROSMITHS, INC.**

P.O. Box 561, Cambridge, MA 02140 (617) 354-1224  
 BIX:cheath Compuserve: 74216,2117

**FASTFONTS**  
**\$49.95**

Mail orders,  
 add \$3 P & H.  
 Mass. residents  
 add 5%.



# THE MOUSE THAT GOT RESTORED

*The Amiga Mouse is a very well-engineered device. It is built to provide thousands of miles of rollerball and clicks for a reasonable price. It is not so economical, however, to be a throwaway item.*

by Jerry Hull and Bob Rohde

*This article tells you how to repair or replace the micro-switches under the buttons in the Amiga mouse. This may well be the simplest hardware project imaginable. However, by replacing a part that costs literally pennies, you can save yourself the \$60 to \$70 expense of a new mouse.*

## **Jerry: The Switch that Failed**

I am not what you would call "a hardware person." My previous experience consists solely of assembling a Heath-Kit Smart Outlet Box. Fortunately, my friend Bob knows hardware very well. (He helped me with the outlet box.) I am sure of this, because he gets paid an outrageous salary. He is the principle architect of the repair described below.

I was one of the first on my block to get an Amiga, back in November 1985. I've used it a lot since, especially after becoming addicted to MEAN 18. A couple months ago my game went rapidly downhill (a little golf joke) when my left mouse button began misfiring. Sometimes it didn't make contact, and I would whack my putt out into the rough.

I abruptly realized that the good health of my wonderful computer was completely at the mercy of what was, till then, a somewhat "cutesy" input device. It was also around this time that I learned that a new Commodore mouse would cost me \$60 to \$70 (depending on who I bought it from). So I decided to take a look inside the little beast.

It soon became apparent what had gone wrong. There are two little switches that are depressed by plastic ridges on the underside of the mouse buttons. These ridges do not meet the switch "actuators" squarely -- they push on the bottom part first. (By "bottom" I mean in the direction of the ball, as opposed to the tail.) The bottom part of the left switch no longer made reliable electrical contact, apparently because of excessive use. However, the top part of the switch still connected fine.

So my first solution was simply to glue a little piece of plastic cut off from a guitar flatpick onto the top part of the left switch's actuator. This way, the top of the switch would make contact when the mouse button was pushed. This is what counts, for me, as hardware expertise. Unfortunately, the glue didn't work so well, and the piece of plastic fell off.

Hoping for a more professional (read "more permanent") solution, and realizing that my resources were exhausted, at

this point I drew my friend Bob into the project. Below, he describes the ultimate solution, in which we replaced the defective switch with a commercially available component.

However, you might also be interested in our interim fix. Using some occult meters, Bob determined that the four-pinned switch is functionally symmetrical. So we simply unsoldered the left switch, rotated 180 degrees, and soldered it back in! But who could tell how much life the switch had left? Complete replacement seemed a better answer.

## **Bob: Analyzing the Problem**

Having determined that the switch itself had failed and that it was a discrete component, I set about looking for a replacement. I found some that seemed to have identical electrical and mechanical specs in a catalog from

### **Digi-Key Corporation**

P.O. Box 677

Thief River Falls, Minnesota 56701

It is a Panasonic "super thin flat knob" switch, number EVQ-PXR04K. The Digi-Key part number is P9950. Their price is \$.36 for a single piece. However, they require a minimum order of \$10.00, so you have to find something else in their catalog, or really stock up on switches.

If you buy ten or more, the price goes down to \$.33 apiece. However, there is also a flat shipping charge of about \$5.00 with Digi-Key. These costs can be defrayed if a number of people get together on the purchase: a user's group, for instance.

When the switches arrived - in just two days - I realized that they were not, in fact, identical, and would need a little modification. The switches in the Amiga mouse have actuators which are about 1 millimeter tall. The Digi-Key switches are only half that high.

A more thorough search of catalogs would probably locate a better replacement switch. However, the Digi-Key switch proves to be quite usable in the Commodore mouse, with just a little additional effort. This is what we well-paid hardware types call "FTF": filing to fit.

*continued...*

### **Bob: Filling to Fit**

To make up the small difference in height, I simply applied a 0.5 millimeter layer of well-mixed two-part epoxy adhesive to the top of the actuator. This can be done quite easily with the help of a medium-sized nail.

Pick up a small drop of the mixed epoxy on the point of the nail and then put the point on the center of the actuator. The epoxy will flow quite nicely onto the plastic. By changing the height of the nail above the switch, you can control the rate of this flow. It might be a good idea to practise making dots of glue on something you don't care about ruining, first.

Move the epoxy around the top of the actuator until it is all covered, and about a half-a-millimeter high. This is a pretty thin layer. (A matchbook cover is about a third of a millimeter high.) Just add enough glue to make the DigiKey switch's actuator the same height as the one in the Amiga mouse.

It will flow naturally to a fairly even, slightly domed layer. Be very careful not to let any epoxy run down the sides of the actuator button. This could glue it permanently open. Of course, if you bought \$10.00 worth of switches, you'll have plenty on which to improve your technique.

When the dot of glue looks good, remove the nail while simultaneously rotating it. This will help to keep you from accidentally pulling the dot back off or dropping excess glue on the switch. Let the epoxy set for at least a day, or until it is completely dry. It will take longer to dry if you skimp on hardener when you mix up the epoxy.

### **Bob: Performing the Transplant**

In addition to the built-up switch, you will need some solder wick or "desoldering braid" (you could also use a desoldering bulb), a small phillips-head screwdriver, some resin-core radio-type solder, a wet sponge, and a good soldering iron. A 22 to 25 watt pencil-type works best, one with a 3/16" or 1/8" chisel tip. A pair of needle-nosed pliers will also prove helpful, as would a small vice to hold the small circuit board while you soldering on it.

You start by removing the mouse cover. On the underside of the mouse, where the cord ("tail") comes out, you will see two small phillips-type screws. Remove these and put them in a safe place. Turn the mouse back over, and pull up carefully at the tail end of the top cover. Try not to pull on the cord itself!

When you see a small gap between the top and bottom covers, you can pull the top cover tail-ward, until it disengages from two tabs at the other end. Put the top cover aside where the dog can't get it.

The mouse's innards are attached to the bottom mouse cover near the "tail" (at the top of Photo 1). In the center of the mouse you will see a black plastic assembly about 50 millimeters square with the mouse ball in its center. This is the pulse generator that provides information about the mouse's movements to your Amiga. It is a clever piece of manufacturing engineering, consisting essentially of two perpendicularly-oriented (X & Y) incremental digital encoders.

Attached to the top of this assembly - towards the tail cord - is a small circuit board with the two click switches mounted on it. The switch with the black actuator, on the left, is the Digi-Key replacement. There is a single screw holding this board to the pulse generator housing. Remove the screw and lift straight up on the board, so that two small plastic pins on the other end don't bind.

When you flip the board over, you can see the pins for the switches soldered to their pads -- four pins for each. To desolder a pin, lay a section of the solder wick directly on the solder, and place the hot soldering iron tip on the wick. Once the solder melts, it will be drawn up from the pin and pad, into the wick. Move to a new section of wick when the current section is full. Clean off every trace of solder on the pin and pad.

Do this for all four pins, and then remove the switch from the board. A little gentle tugging with needle-nose pliers might be needed to loosen the pins from the circuit board. Replace with the switch you have doctored with epoxy, and solder its four pins into place. Reassemble the mouse, and the job is done.

### **Jerry: Soldering Tips and Postmortem**

Bob has skipped over the actual details of soldering. For the sake of those who know even less than I, let me offer a few tips garnered from my vast experience.

Make sure that the solder you use is "resin core" electrical-type solder, instead of "acid core" solder. Ask somebody if you're not sure what you have.

Keep a small, damp sponge close at hand, and frequently wipe the tip of your soldering iron clean on it. Then melt solder on the iron tip until it looks nice and shiny. This is what the pros call "tinning."

When you're ready to solder a pin, hold the iron tip against both the pin and the circuit board pad for 2 or 3 seconds, until both are hot. Then hold the solder against the other side of the pin. You want the heated pin and circuit pad to melt the solder, not the soldering iron itself.

Allow the melting solder to flow completely around the pin, then remove both solder and iron and let the connection cool. Now do the next one. Do not forget to tin!

"How did this repair job work out?", the more sensible of you are asking. As I remarked, it's been about two months since we collaborated on the repair. Given the extent of my MEAN 18 addiction, I suspect that 100,000 clicks is an underestimate of the abuse my left mouse button has since received.

During this workout it has identifiably misfired only three times: 99.997 percent reliability. Since the Digi-Key actuator is shorter than the Commodore original, my guess is that it very occasionally double-bounces. However, I find this rare malperformance quite tolerable. After all, I am consistently under par, even at dread Turnberry!

# Sleuthing Public Domain Disks with CLI

by John Foust



THE AMIGA is often so tantalizing to the novice user. The icons on the Workbench may give the impression of ease of use, but some aspects of icons are very frustrating to beginners. The power of the machine seems just out of reach.

There are seemingly artificial restrictions on file names and disk names. Magically, the novice is 'just supposed to know' that you cannot use the slash character in a NotePad file name.

On the other hand, there are arcane names for other system files and directories. The labels 'file', 'icon', 'volume' and 'disk' swim in user's heads. Which is which? Some files do not have icons, and never appear on the Workbench. Some icons do not have files. Some directories do not have drawer icons. Disks have two names, at least. There are so many incredible programs available, and not all work from the Workbench. The programs must be invoked from the CLI, or command line interface.

While the Workbench and most commercial program disks are well organized, many users have had troubles exploring public domain disks. Problems like this occur in other situations. Perhaps you have seen another user do something on their Amiga, and you would like to reproduce it on your own. Or that you have read about a technique in a magazine, and the instructions appear to be incorrect. What next?

Given a public domain disk, what is the best way to explore it? Exploring a public domain disk is like exploring a well by riding in the bucket. Beforehand, you might know something about the disk and the well. You may be told the disk contains a wonderful animated demo of a dancing bear, and a series of Impressionist pictures. Similarly, you know what to expect at the bottom of the well. In both cases, lowering yourself to the bottom can be done with style, with a plop, or not at all.

Because you might end up sinking with a 'plop', be sure to make a copy of the public domain disk before you start. Do not take chances until you know a disk is safe, and how to access all the programs on a disk.

## **Auto-boot disks**

All users jump in the bucket, and prepare to go down. With the public domain disk, the user might try to insert it as a Workbench disk, after turning on the machine and after inserting the Kickstart disk.

In the best case, the bear demo and the pictures will appear on their own, with no further intervention from the user. On this disk, someone wise prepared the disk beforehand and gave the computer instructions about running the bear demo. This disk is meant to be used as a boot disk. But few disks are organized this way. Most commercial software is a bootable Workbench disk, but the program does not start automatically after the Workbench loads.

Be careful in this situation. Remember, you have booted a foreign Workbench. Most users have a Workbench disk that has been customized to their needs. These custom changes include the date and time, the type of printer you own, your mouse pointer, and the colors you prefer. For example, this foreign, self-booting demonstration disk might not have the printer driver needed for your printer, so you would not be able to print the pictures on this disk. In order to print pictures from this disk, you will need to move the pictures to another disk. More on this later.

Jumping in the bucket might mean double-clicking on the disk icon that appears when the disk is inserted while the Workbench is running. In the worst case for the novice user, there will be no icons in the window that appears.

# AMIGA T-SHIRTS AND SWEATSHIRTS!

*Quality white shirts silk screened with the Amiga logo in beautiful color*



**ORDER YOURS TODAY!**

Sizes: Small, Medium, Large, X-Large  
Prices: T-shirts: \$12.00 Sweatshirts: \$18.50

Also available: Amiga Stickers \$1.50  
(Great for car bumper, window, notebook, etc. Black & White)  
All prices include shipping and handling. In CA, add 6% tax.

SEND CHECK OR MONEY ORDER TO:  
T's Me, P.O. Box 11746, Santa Ana, CA 92711

2525 Shadow Lake, Santa Ana, CA 92701  
Dealer Inquiries Invited: Call (714) 639-6545

Amiga is a registered trademark of Commodore-Amiga, Inc.

Allow 3-6 weeks for delivery.

What does this mean? A very, very common first impression is that the demo and pictures are not present on the disk. After all, if no icons appear, where could the demo be? In the Amiga, files of programs and pictures do not need icons. Icons are optional, and easy to use, but not necessary. There are other ways of running programs on the Amiga, methods other than double-clicking on icons.

In some ways, this example is not kind. Many user groups have gone to great lengths to insure that their public domain disks are easy to use, and have icons for those users who prefer to use icons. This example is more likely to happen: the disk will have some files with icons, and some without.

In this example, let us presume that no icons about bears appear, so our hopes for the dancing bear have been temporarily dashed. However, a few icons are present, and some look like picture frames. They all have names reminiscent of Impressionist art. These are no doubt the pictures we seek. Some other icons are present. Searching a public domain disk involves a certain amount of sleuthing, even for experts.

Double-clicking on the picture icons only produces a flash on the screen. The error message 'Error while opening DPaint: 205' appears on the Workbench menu bar. (If you have several windows open, they might be blocking the

Workbench menu bar. Resize and move them if necessary, if you see a flash of the screen, and nothing happens. Error messages often appear on the menu bar.)

What happens when you double-click on an icon? Two quick mouse button clicks on an icon are interpreted by the operating system as an intent to start a program. Some icons represent programs themselves, other icons represent files that are acted upon by programs. In other words, some icons represent data, and the icon knows which program to invoke to act on that data.

In this case, something went wrong when the operating system felt the double clicks. The error message is cryptic, it is true. The number '205' has little meaning to me, as well as most other Amiga users. But the message tells something about what the operating system expected to find.

Here is an easy way to find a few clues about an icon, and the file associated with it. An icon is somewhat separated from the program or data it represents. When you invoke the IconEd icon editor from the Workbench disk, and load an icon, you are editing a very small file on the disk. The program or data is stored in a separate file on the disk. The IconEd program is managing the details of this operation, but there is something important to learn here. An icon is only complete with a corresponding program or data file. Both files are stored on the disk, in drawers. If either one is not present, the program or data file will not work as planned. Later, we will see a method to check that both files are present.

A single click on an icon selects it. The icon will change color. In some cases, the icon will change shape, as well. Do not let this surprise you. Some people have Trashcan icons whose lid opens when clicked once. The second click on the selected icon will open a drawer or start a program. The first click only selects the icon.

After selecting the picture icon, use the right mouse button to choose the Info option from the first Workbench menu on the left. The Info command will bring up a screen with information about the icon. The icon file itself holds information about the program. The icon file holds the image of the icon itself, and this can be changed by the IconEd program. The rest of the information is available here, in Info.

At the top of the Info screen, we see that the icon we selected is named RainyDay. It is of type Project. The size in bytes is 12458. Below, a window tells us the Default Tool for this icon is DPaint. This explains the cryptic error message we saw on the Workbench menu bar moments ago. When this icon is double-clicked, it expects to invoke a program named DPaint. Meanwhile, you might guess that moving this icon to a Deluxe Paint disk will let us double-click on it, and have it invoke the Deluxe Paint program. This picture icon is a Project icon. Project icons are used for data files that know the name of the program that created them, or that can act on them.

## Volumes

You are free to change the name of this program in several ways. If you changed the file name of the Deluxe Paint program to 'Painter', you would need to select each picture icon you have, and use Info to change the name of its Default Tool. To do this, click on the text, and change it using the cursor, backspace and delete keys, and then entering 'DPaint:' at the beginning. In this example, change it to 'DPaint:DPaint'. Be sure to press Return after making these changes. Then click 'SAVE' in the lower left corner to save this change to disk.

Knowing something about Amiga disk names can enhance this Default Tool name. By changing the default tool name to 'DPaint:DPaint', double-clicking will bring a different result. A double-click on this icon will now bring up a system requester box in the upper left corner of the Workbench screen that says 'Please insert volume DPaint in any drive'. If you insert the Deluxe Paint disk at this point, the program will load, and the picture will be loading automatically. (If you have changed the volume name of your Deluxe Paint disk, all bets are off, and the system requester will remain.)

Adding 'DPaint:' to the name of the Default Tool told the operating system to expect to find the program 'DPaint' on a volume named 'DPaint'. What is a volume? It is the name for a disk. The standard disk for Deluxe Paint has a volume name of 'DPaint', this is the name that appears below the disk icon on the Workbench screen. There is no contradiction about having a program file named 'DPaint' on a disk named 'DPaint'. In 'DPaint:DPaint', the volume name comes first, followed by the colon, then the program name, 'DPaint'.

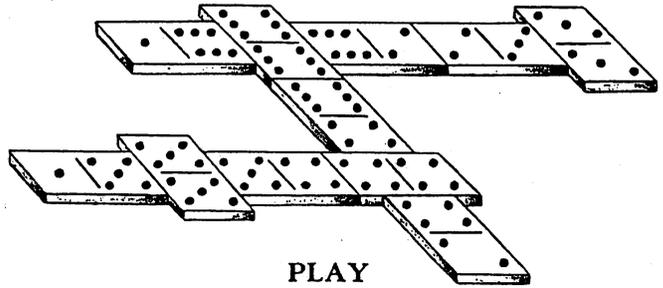
This use of the volume name serves several purposes. It means you do not have to copy the picture to a Deluxe Paint disk to view it. Double-clicking on this icon presents a requester, reminding you to insert the Deluxe Paint disk, so the DPaint program can be invoked. If you have two disk drives, and have the Deluxe Paint disk in the second drive, this system requester will not appear, because the operating system knows the Deluxe Paint disk with the volume name 'DPaint' is in the drive, and it will immediately go there to look for the program named 'DPaint'.

The purpose of the picture icons is clearer. They expected to be placed on a Deluxe Paint disk to be viewed. Very often, a program exists on public domain disks that will view IFF pictures without a program like Deluxe Paint. Besides the picture icons, imagine there is an icon labelled 'SeelLBM'. Without knowing its purpose, we can use the Info command to learn more about it.

Single-clicking the 'SeelLBM' icon, and selecting 'Info' from the Workbench menu brings up a window that tells us this is of type Tool. Traditionally, tools act on other things, and this analogy extends to the Workbench. This is a program that acts on other data files. There is little other useful information in the Info window, however.

The true purpose of the SeelLBM program is unknown, based on these clues. A smart user might be able to tell you that it is used for viewing ILBM files, which are a type of IFF files.

# DOMINOES



## SEVEN TOED PETE

on your 512k Amiga

SEVEN TOED PETE is a numerical strategy game designed especially to take advantage of the Amiga's graphic and speech capabilities. Learn with the DOMINO TUTOR then play against the expert. Challenging for adults yet easy for children to learn, and it provides excellent arithmetic practice.

Send check or money order for \$24.95 to

### POLYGLOT SOFTWARE

10431 Ardyce Court, Boise, Idaho 83704

More acronyms? 'ILBM' stands for 'interleaved bit map', which is a way of describing the way a color screen image is stored on disk. 'IFF' is a more common acronym in Amiga parlance. It stands for 'Interchange File Format', and was coined by a group who created a standard for data exchange on the Amiga, so all programs could work together more easily.

### Extended select

One might guess that all the picture icons on this disk should have had their Default Tool changed to 'SeelLBM'. This would have allowed us to view them without inserting any other disks. This situation is somewhat common on a public domain disk. The files might be arranged in strange ways. The disk might hold a mish-mash of files from several sources. Some people create new public domain disks from other public domain disks, and sometimes files are lost in translation.

Why not change the Default Tool to Deluxe Paint, as well? In fact, either is fine, and will dictate which program is invoked when the picture icon is double-clicked. Because of the IFF standard file format, many programs could operate on this picture data. If we wanted to invoke Deluxe Paint on this picture file, we could single-click the picture icon, then while holding down the shift key on the keyboard, double-click the Deluxe Paint icon. This technique is called 'extended select.' It can be used to override the Default Tool of an icon, and direct the computer to allow this other program to operate on this data.

## ONE MEG MEMORY & CLOCK CARD

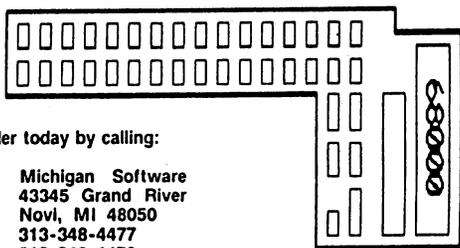
Introducing the "INSIDER", a One-Megabyte RAM upgrade that comes complete with a Real Time Clock and plugs "Inside" the Amiga CPU.

Simply unplug the 68000 Microprocessor and plug the INSIDER into the 68000 socket, then plug the 68000 back on top of the INSIDER. It's that simple to add an ADDITIONAL One-Megabyte.

Complete with ADDMEM software for Versions 1.1 and 1.2 of the AMIGA operating system. Completely transparent to the operating system if a program such as DPAINT can't run in a Computer with more than 512K!

- AUTO CONFIG's under 1.2
- No Forced Wait States
- Transparent Refresh
- Fully tested and burned in
- ADDMEM program included
- Exhaustive Memory test included
- Fast Memory that's TRULY FAST!
- Only 600ma Power Draw (typical)
- **WORKS WITH AMIGA SIDECAR!!**
- Real Time CLOCK with built in 10 year battery!
- One Year Warranty

All this for only \$349.95 plus shipping. No soldering or trace cutting. Dealer installation is recommended.



Order today by calling:

Michigan Software  
43345 Grand River  
Novi, MI 48050  
313-348-4477  
313-348-4478

VISA/MASTERCARD/AMEX

SPECIAL\*\* 256K RAM CARD FOR ONLY \$49.95 + SHIP, while supplies last, upgrade to 512k for less! Call or write for FREE CATALOG

A drawer icon named 'Pics' is present on the disk. Double-clicking here brings up a window with several more picture icons.

### The Amiga CLI

We must travel deeper into the well to learn more about this public domain disk. The commands and programs of the Amiga CLI will show everything on a disk. Minimal CLI skills are a must for people exploring public domain disks. The CLI is cumbersome to some, and "home, sweet home" to others. There are many Amiga users who never enter the CLI, and there are many who never use the Workbench, and only use the CLI.

The command line interface is text based. The user enters short commands, such as 'dir' or 'type', and the computer runs the program named 'dir' or 'type'. When that program finishes, the command line interface prompt appears, and the user repeats the process. For every program they run, they have entered a short command at the command line prompt.

On the Amiga, the command line prompt is a number, followed by a greater-than sign. The cursor rests on the same line, after the prompt:

1>

On many computers, such as the IBM PC, a command line interface is the only way people send commands and run programs.

Many Amiga users have never used the CLI. On a virgin Workbench disk, fresh from Commodore, the CLI icon is not present on the Workbench. Before it can be invoked, it must be enabled in a separate process.

To use the CLI on the Amiga, first double-click on the Preferences program icon. In the middle-left hand side of the Preferences screen, there is an the word 'CLI' with an 'on-off' box next to it.

If you are having trouble deciding which color is the highlight color, look at the 1-2-4 box on the right hand side of the screen. Only one of these boxes is highlighted. The other two are the background color, the same color as the edges of the screen.

To leave the Preferences program click once on the box marked 'Save' at the bottom left part of the screen. This will save your changes to your Workbench disk. This will not affect the color scheme you have chosen, it will only make the CLI icon visible the next time you boot with this Workbench disk.

To see the CLI icon, you must reboot. The CLI icon will appear in the System drawer, so double-click on the System drawer icon to see the CLI icon. Double-click the CLI icon, and a window will appear. It is titled 'New CLI Window', and a '1>' prompt should be present with a cursor in the upper left hand corner. Press return a few times, it will not hurt. All commands in the CLI are sent to the operating system with the Return key. You just told the CLI to do nothing, several times, and it responded in kind. You might want to resize the CLI window to the full height and width of the screen.

### What lies beneath the surface of the Workbench, in AmigaDOS?

There are several hidden aspects here. From the Workbench, all is disks, icons and drawers. These are direct analogies to what is actually happening in AmigaDOS, beneath the user-friendly veneer of the Workbench. The functions and appearance of the Workbench are a superset of what exists underneath in AmigaDOS and the CLI. The Workbench functions, such as icons, are not necessary for AmigaDOS. For this reason, many disks do not have icons for programs or files. CLI commands can reveal all files on a disk, whether or not they have icons.

The bounds and limitations of the Workbench are gone in the CLI. In their place, short, user-typed commands cause action, not mouse clicks. There is a close analogy between the ease of mouse clicks, and the commands in the CLI. Essentially, the computer gains the same information from each command, whether from clicks on the Workbench or commands in the CLI. In the case of the Workbench, as much information and dirty work as possible is kept from the user, in order to make the operating system interface more user friendly.

## Files and directories

All information on a disk is grouped in units called files. Files can contain any amount of data, from zero characters to the full size of a disk, which is more than 900,000 characters. All files have names. Usually, the names of files have little meaning in AmigaDOS. To the Workbench, a file name with a suffix of '.info' indicates an icon file.

The Workbench has drawers, separate filing areas for data. In AmigaDOS, these are called directories. Some encourage the analogy of directories being file folders, with the pleasant exception that all file folders can hold other file folders, and that file folders are never full until the disk is full. Presumably, you could go overboard and insist that disks are filing cabinets.

The 'dir' command is an absolute arbiter of files residing on a disk. 'Dir' is a program stored on the Workbench disk. If you type 'dir' at the CLI prompt, and press return, the CLI loads the 'dir' program from disk. The 'dir' program runs and prints the names of the files on the disk.

To make things easier, enter the following lines at the CLI prompt:

```
1>
1> copy c:dir ram:
1> copy c:list ram:
1> copy c:type ram:
1> copy c:cd ram:
1> path add ram:
1>
```

These CLI command sequences will allow you to remove the Workbench disk, and insert the public domain disk you are inspecting. These commands make copies of the 'dir', 'list', 'type' and 'cd' programs, and store them in the memory. From now on, if you enter 'dir', the program will be loaded from memory, and not the disk. This is handy if you do not have a second drive. Even if you have two drives, you might find this preferable to keeping the Workbench in a drive. Loading programs from memory is much faster than loading from disk.

With one simple CLI command, you can see all the files on a disk. Enter 'dir opt a' at a CLI prompt. You will see a list of all the files on the Workbench disk. This is an extensive list. The command 'dir opt a' is typical of CLI commands. It starts with the name of the program - 'dir' - followed by other space-separated words that modify the results of the 'dir' program.

If you have two disk drives, you can view the files on the external disk drive by entering 'dir df1: opt a'. The additional 'df1:' instructs the 'dir' program to look for files on the first additional drive.

When you double-click on a disk icon on the Workbench, you are telling the computer you wish to focus your attention on that disk. There is a similar CLI command that directly corresponds to opening a disk icon, or opening a drawer icon. This is the 'cd' command. Imagine that 'cd' stands for "change disk" or "change directory."

## Use Your Own Photos

...in programs such as Deluxe Paint or Images. Your pictures from flat art 2" x 3" to 8½" x 11" or color slides 35mm to 4" x 5" will be digitized by the Digi-View system to 32 color, 320 x 200 resolution pictures compatible with any IFF paint program. Minimum order is 8 images for \$24, disk included (California residents add state sales tax) plus \$2.50 shipping. Additional images \$2.00 each. Pictures may be cropped to fill the screen. For no cropping specify *full frame*.

## Photographic Clip Art!

Sample disk includes landscapes, clouds, trees, buildings, celestial object, etc. Use in your own IFF paint programs. Customize to suit your needs by flipping, stretching, stamping, changing colors. More realistic than drawings! Order Clip Art Sampler #1. Catalog of other Clip Art disks will be included. \$20 (California residents add state sales tax) plus \$2.00 shipping.

# DIGI-PIX

800 Heinz Street □ Berkeley, California 94710 □ (415) 644-0614

Deluxe Paint is a registered trademark of Electronic Arts. Images is a registered trademark of Aegis Software. Digi-View is a registered trademark of NewTek. Copyright 1986 DIGI-PIX.

To tell the operating system you wish to focus on the disk in a drive, enter a 'cd' command. To focus on the internal disk drive, enter 'cd df0:'. Because the 'cd' program is stored in memory because of the series of 'copy' commands recommended above, the operating system does not need to load the 'cd' program from the Workbench disk.

Here is what appears when the public domain disk is inserted in the drive, and attention is focused upon it, followed by a 'dir' command to view all the files on the disk.

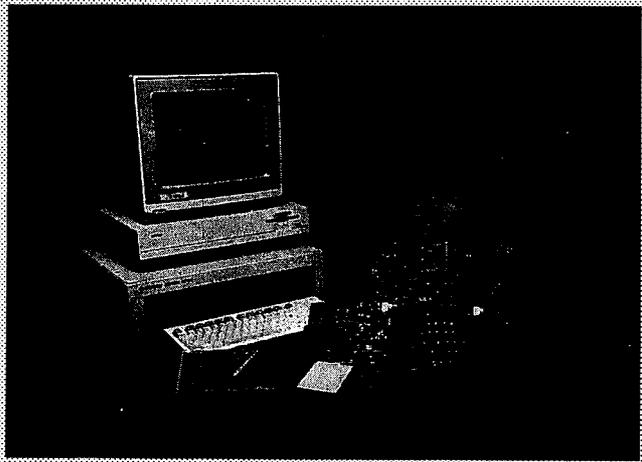
(Insert the disk in the internal disk drive.)

```
1> cd df0:
1> dir opt a
    bear (dir)
      background.pic      bear
      bear.data            readme
    pics (dir)
      Balloon.hires       Garden.med
      Rowboat.lores       Rowboat.lores.info
      LittleGirl.lores    LittleGirl.lores.info
    .info
    Hamburg.info         Hamburg
    RainyDay             Pics.info
    SeeILBM              RainyDay.info
    Walking              SeeILBM.info
    Zoo.BAS              Walking.info
                      Zoo.BAS.info
```

2>



## THE GOODS...



## THE GENERAL...

### **\*A 1000 Expansion Chassis \$699.00**

5 expansion slots  
135 watt power supply  
Zorro compatible  
Room for up to four drives  
Safe "flex" Amiga interconnect

### **\*Hard Disk Controller \$300.00**

Two ST506 hard disk drives  
and two floppy disk drives

### **\*RAM Board \$799.00**

Two meg of "zero wait states"  
VLSI controller  
Low noise multi-layer PCB

### **\*Prototyping Card \$49.00**

### **\*Extender Card \$49.00**

### **\*256K Memory Module \$109.00**

## THE NET...

### **\*ETHERNET Controller Card \$699.00**

High performance Standard and  
Cheapernet connectors  
Side mount version available

### **\*AMIGA NFS \$149.00**

Full implementation of  
industry standard network  
file systems protocol.

### **\*ARCNET Controller Card \$425.00**

Low cost  
General purpose  
Side mount version available

### **\*ANET**

Arcnet software package  
All Amiga network  
File sharing/Remote print

## THE DEAL...

\*First 500 orders received before July 1, 1987 in excess of \$1500 will receive our Hard Disk Controller Card absolutely **FREE** of charge.

\*This offer may be withdrawn at any time without notice. A copy of this ad must be included with your order.

**\*These products are available for the Amiga A2000.**

## THE SOURCE...

★ **AMERISTAR TECHNOLOGIES INC** ★

P.O. Box 415  
Hauppauge, N.Y. 11788  
(516) 7243344

N.Y. state residents add appropriate sales tax.  
NFS is a registered trademark of Sun Microsystem Corp.  
Amiga is a registered trademark of Commodore Amiga, Inc.  
Ethernet is a registered trademark of Xerox Corp.  
Arcnet is a registered trademark of Datapoint Corp.

```

1> type readme opt h
0000: 0A44616E 63696E67 20426561 72204465 .Dancing Bear De
0010: 6D5F0A2D 2D2D2D2D 2D2D2D2D 2D2D2D2D mo.-----
0020: 2D2D2D2D 0A0A546F 2072756E 20746869 ----.To run thi
0030: 732C2065 6E746572 20276265 61722720 s, enter 'bear'
0040: 61742061 20434C49 2070726F 6D70742E at a CLI prompt.
0050: 20205468 6520666F 6C6C6F77 696E6720 The following
0060: 66696C65 730A6D75 73742062 65207072 files must be pr
0070: 6573656E 7420696E 20746865 2073616D esent in the sam
0080: 65206469 72656374 6F72793A 0A0A2020 e directory:..
0090: 20626561 722E6461 74610A20 20206261 bear.data. ba
0A00: 636B6772 6F756E64 2E706963 0A0A0A ckground.pic...
1>

```

The display stopped short in the last line because the entire file had been displayed before control-C was hit.

Moving on to the 'Pics' directory with 'cd df0:pics', note that two of the pictures have icons, but two - 'Balloon.hires' and 'Garden.med' - do not. To make icons for these picture files, use the copy command to duplicate an existing '.info' file for each:

```

1> copy littlegirl.lores.info Balloon.hires.info
1> copy littlegirl.lores.info Garden.med.info

```

Capitalization is not important. When you go back to the Workbench and open the 'Pics' drawer, the three icons will be stacked on top of each other because the 'copy' command simply duplicated the information in the 'LittleGirl' icon, which includes the starting position of the icon within the window. Move them to a new position, and select the Snapshot option from the Workbench menu.

There is one last file in the 'dir opt a' display for this disk, the 'Zoo.BAS' file. Move back to the root directory of the disk by

entering 'cd df0:'. This introduces another clue for examining public domain disks. File names follow a regular pattern. Programs in the Basic language often carry a '.BAS' suffix. C source code has '.c' and '.h' extensions. Texts have '.TXT' or '.DOC' extensions. Executable programs rarely have suffixes. Data files might carry a '.DAT' extension. IFF pictures might have '.PIC', '.lores', '.med' or '.hires' extensions.

Note that Amiga Basic is not on this disk. If it was, it would appear in the 'dir opt a' display. It is not legal to include the Amiga Basic program on public domain disks, so this is a common situation. To run this program, you must copy the Amiga Basic program to this disk. From the Workbench, this means inserting the Extras disk that came with your machine, opening its disk icon, and clicking and dropping the Amiga Basic icon in the window of this public domain disk. Alternately, you could move the 'Zoo.BAS' icon to a disk that has Amiga Basic in a similar way.

To close the CLI window, replace the Workbench disk in the drive, and enter 'endcli'. The 'endcli' program will be loaded from disk, and the window should disappear.

With these simple tricks, the CLI is no longer a mystery, and you have command of what exists below the Workbench. There are many more CLI commands. They are found in the 'c' directory on the Workbench disk. If you are interested in learning more, there are many books available that describe each command in detail.

•AC•

## Multi-Forth...at a new low price... \$89

Simply the best programming environment for the Amiga

Multi-Forth is a new language designed to unleash the full power of the Amiga. Multi-Forth provides complete access to all Amiga libraries including Intuition. It compiles stand-alone applications in seconds (other languages typically take several minutes). There are no royalties, and no "levels." CSI provides the best support of any computer language vendor, including CSI technical hot line, our own CompuServe public forum at GO FORTH with hundreds of public domain/shareware programs, and a comprehensive 350 page manual. Programming the amazing Amiga is interactive and fun with Multi-Forth.

Contact us for a technical data sheet with the complete list of features.



Multi-Forth is a registered trademark of Creative Solutions, Inc.  
Amiga is a trademark of Commodore Corporation.



4701 Randolph Rd. Suite 12  
Rockville, MD 20852  
301-984-0262 in MD or  
**1-800-FORTH-OK (367-8465)**

# Highlights From The SAN FRANCISCO COMMODORE SHOW

by Steve Hull

## **Aegis Development**

The Aegis booth, as always, was a treat for the eyes and ears, and filled to capacity throughout the entire show. Looking at their new offerings, it's hard to believe that as recently as 18 months ago they were "Aegis who?" The small company's persistence in supporting the Amiga has turned it into a large company.

The most exciting new offering from Aegis was demonstrated in a two-minute videotape titled "Infinite Loop." The tape showed off the kind of three-dimensional animation possible with Aegis' soon to be released product, Videoscape 3-D. It's a difficult product to describe but I'll try.

The tape begins with the familiar bouncing Amiga ball. But this one's doing something you've never seen it do before; it looms large in the screen, and then the red "checkers" slowly pull back into the sphere, leaving the ball slowly rotating, a framework of white checkers. This cuts to an outer space scene, where a starship dodges through a meteor field and ultimately through tunnels of glowing energy. The view shifts to a bird's eye view flying low over a city; skyscrapers pass beneath in dizzying perspective.

From there we follow the progress of a red sports car down a road. The "camera" follows as the sports car pulls up to a house; the camera then leaves the car, flies around the house, zooms on a doorway - goes through the door and into the room, where an Amiga (of all things) sits on the desk. Without a break, the viewpoint zooms until the computer screen fills the picture, and then - the Amiga ball. Infinite loop. The first time I saw the tape they had to shake me to get my breathing started again.

Here's the hitch: The program takes between 1-2 seconds to render a screen done in 3-D solids, although it can execute wire-frame animation smoothly in real-time. To duplicate the videotape demo, you'll need a VCR capable of recording single frames, much in the same manner as an animation camera. But Aegis just may have been blessed by good timing - such VCRs have already begun creeping into the home market. They're not cheap - but up until now, video production houses have had to pay up to \$4,000 *per second* of finished product to rent this kind of animation capability.

Aegis is experimenting with a screen compression format that will allow the Amiga to flip through frames in real-time; in any case, Videoscape 3-D promises to blow the desktop video market wide open - its closest announced Amiga competitor retails for roughly 15 times its suggested list price of \$200!

The secret lies in some elegant algorithms written by Videoscape's programmer, Allen Hastings. Most 3-D animation packages are integer based, trading precision for increased speed. The 3-D rendering routines in Videoscape are based on floating-point calculations; this makes its 1-2 seconds per frame speed just this side of incredible. This speed combined with floating-point accuracy will allow Aegis to adapt the 3-D routines used in Videoscape 3-D to future Computer Aided Design products; previously, plotting 3-D solids accurately took prohibitively long.

Besides the demo video, an alpha copy of the program was demonstrated during one of the Amiga sessions. Videoscape 3-D runs on a standard 512K Amiga with one disk drive, and Aegis expects to release it by summer.

A prerelease version of Sonix was demonstrated on an Amiga connected to the MIDI ports of a Yamaha DX-7 keyboard, with head-turning results. Although it can be used to compose music scores, Aegis is promoting Sonix as primarily a music synthesis tool. With Sonix, musicians will be able to generate any type of analog or digital sound, and create or modify existing IFF instruments or other digitized sound files. It allows mixdown to 16 MIDI channels, and will print high-resolution sheet music on any graphics-capable printer listed under Preferences.

A side-benefit of Aegis' research on Sonix will be the eventual integration of music into Aegis Animator; the Sonix "player" routine comprises only 7K of code. Sonix is available now at a suggested retail price of \$79.95.

Aegis also announced but did not demonstrate a new telecommunications package. "Dugal" handles baud rates from 300 to all the way up to MIDI speeds, and numbers among its features executable scripts, macro keys, and XMODEM with automatic chopping of pad characters.

*continued...*

# JUMBO RAM

- Semi kit (no soldering)
- Add RAM chips for 1/2 or a full megabyte
- Software configured for 1.1 or 1.2
- For 1/2 megabyte add 18, 41256-15 chips, for 1 megabyte add 36 41256-15 chips
- Enhances VIP Professional, Draw, Digi-View, Animator, Lattice and many others
- Six month warranty

**JUMBO RAM Board \$199.95**

\$3.50 S & H

RAM Chips available at prevailing price.

*Amiga Schematics still available!*

Order Toll Free!

**800-762-5645**

**Cardinal Software**

14840 Build America Dr.

Woodbridge, VA 22191

Info: (703) 491-6494



Diga! includes several terminal emulations ranging from the plain vanilla VT-100 to the graphics oriented Tektronix 4014 (the Tektronix graphics may be saved to disk for later editing or viewing as an IFF file). Besides the emulations included, Diga! allows users to create any terminal emulation they desire using a custom editor.

Perhaps Diga!'s most intriguing feature is its "Doubletalk" transfer protocol. Under Doubletalk, two Diga!-configured terminals can simultaneously send and receive files - while the operators chat in real time! Aegis got the idea for the Doubletalk protocol when their road-bound salesmen complained about not being able to carry on business during lengthy uploads to the main office. Aegis expects to ship Diga! this month at a suggested retail price of \$79.

Aegis' graphics wizard Jim Sachs was on hand with some graphic demos of his latest project, an interactive computer version of Jules Verne's 20,000 Leagues Under the Sea. He promises the program will set new standards for graphics and sound when it is released this Christmas, but he says he's having problems trying to figure out how to end the game. "The problem with basing a game on 20,000 Leagues is the book ends on a sad note, with Captain Nemo dying and the Nautilus sinking. That's not a very upbeat ending for a game," he notes wryly, "especially if you win." If anyone can think of a better ending, Sachs is ready to award \$200 for the best idea. So far most of the plots submitted have involved aliens and time warps; not precisely what he had in mind. If you have a better idea, write to Jim at P.O. Box 1182, Lake Arrowhead, California 92352.

On the final day of the Show Aegis broke out a mock-up demo of their upcoming graphic adventure, Arazok's Tomb. There's not much to report about the program, as all Aegis showed were animated simulations of a game in progress. Suffice to say the graphics are dazzling, with animated sequences that range from the sublime to the startling, as when a giant hooded cobra suddenly rears up and bites you right in the -- but that would be giving it away, wouldn't it? Arazok's Tomb is available now at a suggested retail price of \$50.

### **JDK Images**

JDK Images did big business with their Pro Video CG I character generator in two major shows; besides the Commodore Show, they also exhibited their product at the Video Expo upstairs in the San Francisco Civic Center. The Pro Video CG I is one of the few products currently available that is written "below" the Intuition level, allowing silky-smooth animation and commercial-quality scrolling of entire hi-res multicolor screens. They wowed professional video producers at the Expo - especially with their list price of \$199.95 - and there wasn't much room to move around in their booth at the Commodore Show either. Certain software titles have become notorious for motivating hardware sales; the CG I is going to move a lot of Amigas into video operations.

### **WordPerfect Corporation**

A development version of the long awaited WordPerfect word processor was on display, with both software engineers and marketing representatives on hand to answer questions. WordPerfect Corporation has made a major commitment to the Amiga, which is good news - not just because their software is bound to raise the state of the art of Amiga word processing, but because of the credibility the company brings with it.

WordPerfect is famous for pairing power with ease of use; features include a fast 115,000 word spelling checker, thesaurus, superior file management, and multi-column formatting and macros. Best yet, Amiga WordPerfect is both document- and file-compatible with the IBM-PC version; if you know WordPerfect on the IBM, you know WordPerfect on the Amiga. In addition, the Amiga version incorporates the full Intuition interface including pull-down menus, mouse support and multitasking.

At first look, the Amiga version appears easier to use and a good deal faster than its IBM counterpart. Availability is expected in the second quarter of 1987, at a suggested retail price of \$395 - but that's sure to be heavily discounted in the marketplace.

### **Elias Engineering**

Elias Engineering of Santa Rosa, California was exhibiting their Digital Circuit-Maker software. The program allows circuit designers to lay out digital logic symbols using the mouse, wire them together on screen, and test the circuit.

Circuit-Maker can also generate its own input waveforms and display results on a simulated logic analyzer and oscilloscope. Circuit-Maker runs on a 512K Amiga and can print diagrams to a standard dot-matrix printer. Digital Circuit-Maker lists for \$125 and is available now. They hope to introduce a similar program for designing analog circuits within a year.

### **Access Associates**

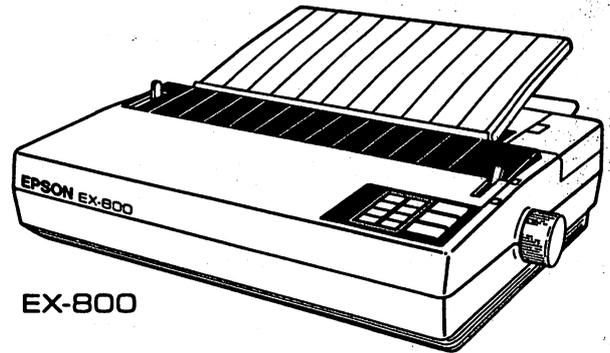
Access Associates fielded an eye-catching display accented by professional quality animated video graphics advertising their Alegra Memory Expansion Unit. The Alegra is a compact (4" x 9" x 3/4") unit that attaches to the side expansion bus of the Amiga 1000. The Alegra is autoconfigurable and has zero wait states. It does not allow for expansion bus pass through. Though the Alegra comes configured with 512K, Access says it's easy to upgrade up to two megabytes with one megabit DRAMs. The Alegra memory expansion is available now and retails for \$379.

### **Brown-Wagh Publishing**

Brown-Wagh Publishing was there in force, exhibiting new entries in both desktop publishing and desktop video. Publisher 1000 is described as an ideal tool for tasks ranging from newsletters to presentation graphics. Features include the ability to compose pages in either medium or high resolution in multiple columns, and easily import IFF graphics

# EPSON® EX-800

## FOR YOUR AMIGA!



EX-800

### **Presenting Epson® EX-800 Dot-Matrix Printer**

- Prints 300 characters per second printhead speed in draft mode (Elite 12 CPI).
- 60 characters per second printhead speed in Near Letter Quality mode.
- New push-button SelectType II front control panel lets you choose from a combination of eight different typestyles.
- Automatic Sheet Load easily and quickly inserts single sheets of paper.
- 8K internal buffer stores up to four pages of data at a time.
- User-installable color option kit adds color to text and graphics.
- Bidirectional printing provides maximum throughput performance for both text and graphics.
- Built-in Push Tractor Feed assures convenient loading.
- One year warranty.

Uses JX-80 printer driver.

EX-800

\$449.95

Plus S & H

ORDER TOLL FREE!

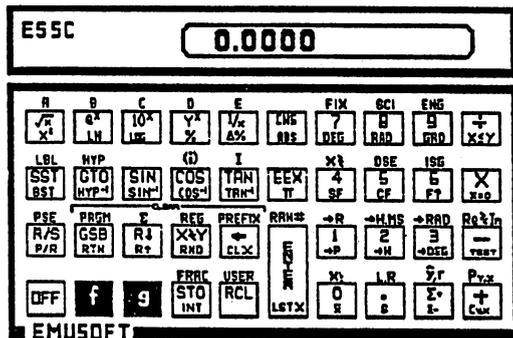
# 800-762-5645

## Cardinal Software



14840 Build America Dr.  
Woodbridge, VA 22191  
Info: (703) 491-6494





**ES-5C PROGRAMMABLE, SCIENTIFIC CALCULATOR**

- \* Operates similarly to the Hewlett-Packard HP-15C™
- \* Programs can be stored and recalled from disc
- \* Logarithmic, trigonometric and hyperbolic functions
- \* Statistics and probability
- \* Linear regressions
- \* Polar and rectangular coordinates
- \* Fixed point, scientific and engineering notations
- \* Complex math operations
- \* Virtually all functions programmable



Now available in full color for the Commodore Amiga

Price \$19.95

To order send check to:

Emusoft Corporation  
1400 Chicago Ave #303  
Evanston, Illinois 60201

Hewlett-Packard and HP-15C are trademarks of Hewlett-Packard Inc.

or word-processed text. A laser printer driver that supports Postscript is currently under development, and Brown-Wagh promises free updates when it becomes available. Publisher 1000 retails for \$199.95, and is available now.

Brown-Wagh was also showing off their entry into the desktop video market, TV\*Text. Billed as a text presentation program, TV\*Text brings home many features that until now would have cost many thousands of dollars on a dedicated character generator. Though it comes with its own custom fonts, it can use any font available for the Amiga. TV\*Text is Genlock compatible and lists for \$99.95.

**Byte by Byte**

Perhaps the show's most lusted-after peripherals were displayed by Byte By Byte, who did brisk business taking orders for their new Pal Jr. hard disk and memory system. The Pal Jr. features a 20 megabyte hard disk coupled with a DMA controller card capable of a mind-boggling 10 megabit per second data transfer. The Pal Jr. also includes a megabyte of RAM, 90 watt power supply and a Zorro-compatible expansion slot. I don't want to say the machine's fast, but a double-click on the hard disk symbol brought up program icons at a rate that resembled the "Dots" demo. At the time of the show all orders were filled through March, but \$1,495 will buy you a place in line!

Matthew Leeds, marketing director for Byte By Byte, revealed a major project using the talents of Eric Graham, the programmer behind the "Juggler" demo. Graham was signed to produce what will eventually comprise a modular HAM/Ray-tracing/Animation software package for the Amiga. The first module - so far unnamed - will be a ray-tracing paint program in which artists will be able to compose their works in three windows simultaneously, representing the X, Y and Z views.

Once the drawing is complete, the artist will define whether the surface is matte or shiny, any light sources (point or diffuse, and the Kelvin temperature), and point of view. From there the program takes over, rendering the finished work in 30 minutes to two hours, depending on complexity. Once rendered, pictures may be saved in a proprietary format, IFF, or a special compressed file format for use in animation.

Byte By Byte has not yet announced an estimated delivery date, and but they anticipate the price for the first module at \$99. A second module capable of producing realtime animation using page flipping (from memory or directly from a DMA hard drive) will follow at a later date.

**Delta Research**

Delta Research was on hand aggressively promoting JForth, a very fast implementation of the '83 Standard (79 and FIG loadable). To help neophytes get started quickly, the manual begins with a Forth tutorial. Representatives claim the product yields the speed of C while maintaining the advantages of an interactive programming environment, compiling as words are defined. The benchmarks supplied by Delta were indeed impressive: 10 iterations of the standard unmodified Sieve of Erastosthenes in 8.82 seconds. JForth is available now and retails for \$99.95

**In-Vitro Design**

This young but promising company made its debut appearance at the February Commodore Show, promoting their flagship product, the MaxiRAM Memory Expansion. \$300 buys 512K of no-wait-state RAM (with the ability to upgrade later to two megabytes) and a true pass-through of the Amiga 1000 expansion bus. Included in the package is autoconfiguration software, a recoverable RAM-disk, and a full one year warranty on parts and labor. The MaxiRAM Memory Expansion is available now.

•AC•

# Speaker Sessions at The Commodore Show in San Francisco

*"...everything from highly technical talks  
to product overviews."*

by Harriet Maybeck Tolly

A full range of speakers were lined up at The Commodore Show. Of the 25 sessions scheduled, seven were aimed specifically at the Amiga owner. The sessions were an hour and a half long, and included everything from highly technical talks to product overviews.

First up was Arthur Abrahams of A-Squared. Although he and Wendy Peterson of A-Squared are responsible for the real-time frame grabber Live!, he did a good job of covering much general material. At first he was swamped with questions about the Amiga 2000. He was gracious enough to answer people even though the official 'New Products' talk was going on concurrently in the other room. One other humorous note on these conflicting talks - while Commodore's Pete Baczor was telling his crowd that Live! was as yet unreleased because of changes Abraham was making, Abraham was telling his crowd that Commodore was responsible for the holdup.

Abraham began with a review of how the Genlock fits into the Amiga video picture. Genlock is the piece of equipment that allows an external video source to be synchronized with the Amiga video signal. Wherever the Amiga displays the background color (color 0), the external signal from the Genlock is substituted. The Genlock is transparent to the Amiga. It has its own clock, and the Amiga gets synchronized to this external source. The most common example of its use is scrolling text over video for credits or titles.

The signal out of the Genlock is not broadcast quality. A time base corrector is needed to correct the signal. Abraham felt that the video market would support a low cost time base corrector to be used with the Genlock.

Next, Abraham covered his product, Live!. Live! is a real-time frame grabber, as opposed to Digi-View, which is a non-real-time digitizer. Live!'s input is NTSC composite video, like a television signal or video camera. He mentioned that it would be quite possible to build a board that could take input from an RGB camera. Live! is 200 times faster than Digi-view. It can handle 12 frames per second in color and 15 frames per second in black and white. The output of Live! is a digital representation of the image. Any image from Live! may be saved in IFF format to be changed by any of the IFF paint programs available.

The video input is the slowest component of the system. Therefore, having a faster processor (ie. a 68020) will not improve the performance of Live!.

It is also possible to send one composite signal into both the Genlock and Live!. This produces a digital image as explained above. Then, wherever the digitization produces the background color, the signal from the Genlock is substituted. This produces one image that is part digital, part analog. The effect is very interesting.

Live! is also capable of producing HAM (hold and modify) images. Images produced in 'fast HAM' mode get a vertical streaking effect. Images can be done in a 'slow HAM' mode, where the subject needs to remain still longer. This produces images without streaking.

A-Squared will be releasing a product called F/X to be used in conjunction with images from Live!. By switching the actual color numbers in the palette (not just the contents), F/X will be able to simulate chroma key. This is the process used to put a video image of your favorite meteorologist on top of the weather map.

Arthur concluded with a highly technical discussion of the blitter. The blitter is part of Agnes, one of the specialized display chips that works in conjunction with the 68000 on the Amiga. It is a coprocessor that is used by the Amiga for fast data movement. The 68000 fetches operands on even cycles, leaving the odd cycles for the display chips. The blitter is the central figure in getting images out onto the Amiga display.

RJ Mical, of Intuition fame, gave two very different talks. His first session was titled 'Introducing the Amiga and the Sidecar'. This was intended to be a general introduction to the Amiga and its capabilities. Most of the crowd were already Amiga converts and wanted Sidecar information. Mical 'began at the beginning' showing Preferences, Clock, Lines demo, Robo City, etc. He mentioned two interesting facts in passing. First, the (nauseating) choice of colors on the Workbench was based on a NASA study which showed them to be easy to see. Second, the graphic demo called Demon, often maligned for depicting an overendowed female, was drawn by a woman.

# KILL A MOUSE GO TO JAIL!

## MOUSEWASH IS A SPECIALLY DESIGNED BALL WHICH:

- CLEANS THE INSIDES OF YOUR MOUSE!
- SAVES YOU TIME AND MONEY!
- CAN BE USED HUNDREDS OF TIMES!
- NEEDS NO CHEMICALS

## SO TREAT YOUR MOUSE TO MOUSE WASH TODAY

**T & L Gallery Vol. #1 \$19.95**  
OVER 85 IMAGES FOR DELUXE PRINT™

**LET SAYIT! READ YOUR TEXT FILES TO YOU!**  
IT READS MOST STANDARD TEXT FILES ALoud!

MOUSE WASH \$7.95--SAYIT \$14.95  
\$2.00 SHIPPING \$3.00 FOR C.O.D.

**T & L PRODUCTS**  
**2645 Wilson Street**  
**Carlsbad, CA 92008 (619) 729-4020**  
™ of Electronic Arts

He continued with an overview of the Sidecar, including the standard PC-Flight-Simulator-on-the-Amiga demo. It was amusing that this session was dedicated to the Amiga 1000, even though the Amiga 2000 had been announced in Byte and Amiga World. Mical was still under non-disclosure as he had not seen the articles himself, and therefore could not speak about the A2000.

Mical's second session, called Programmer's Workshop, was much more technical. He almost immediately went into a question and answer format. Questions ranged from 'Why was mutual exclude for gadgets omitted from V1.2?' (no time), to 'Why hasn't a C64 emulator been done yet?' (it can't be done cheaper than a C64 itself).

At the time this session took place, he had seen the Amiga 2000 articles and was able to answer questions about the new machine. He was asked how he felt about CBM yielding to demand for PC compatibility, and explained that much of the demand came from the European market, where Commodore is seen as a business machine company. Mical's development environment is currently an AT clone (although he says he'll switch to native development as soon as he gets a hard disk).

Desktop Publishing, the latest direction the Amiga has taken, was covered by Kailash Ambwani of Gold Disk. Gold Disk is the publisher of Page Setter, currently the only desktop publishing package available for the Amiga. He gave a complete demo showing how text and graphics can be laid out. Currently this package only handles black and white, although they hope to see full color desktop publishing programs in the future for use with four-color separation printing processes.

Also, they plan to release a Postscript module in early March that will allow PageSetter to be used with laser printers. The rise of desktop publishing programs has created a side business of companies who provide laser printer output for layouts you have created. This is currently widely available for the Mac, and with the release of Postscript for PageSetter, should soon be available for the Amiga.

Jim Goodnow of Manx Software Systems spoke on C programming and the new 3.4 version of the Manx compiler for the Amiga. The new compiler is loaded with improvements and new features that will assure it remains the compiler of choice for many Amiga developers. Overall, the new code generator creates smaller and faster code. The 32 bit int option is now fully implemented and supported. Code generated using 32 bit integers is now much smaller. Goodnow did mention, however, that the Dhrystone benchmark done with the 16-bit library is still 50 percent faster than with the 32-bit library.

Also, the compiler now generates information for use with a source level debugger which will be released in the near future. The new assembler provides full support for the 68010, 68020 and 68881. Version 3.4 of the linker supports four different models for scatter loading, ranging from one big hunk to every module in its own hunk.

This release also includes complete documentation for DB, the Manx symbolic debugger. This is a full-featured debugger. It is command line oriented, similar to Wack. One of the most exciting features Jim demonstrated was that DB can now be started after a program has generated a DOS software error requester, and can show where the error occurred. In some cases DB can even recover from the error.

Other Amiga specific sessions included 'Word Processors for the Amiga' by Louis Wallace, and 'CAD on the Amiga' by Bill Volk.

•AC•

## Amazing Reviews...

# GRIDIRON!

by Keith Conforti  
AC Art Director

*It's fourth down and ten on the thirty five yard line with fifteen seconds left in the game. You're down by two, out of field goal range and there's not enough time to try for a first down, so you're going for broke. Now is the time to use your secret weapon play! You hand off to your fullback who laterals the ball to your tight end who launches a bomb to your wide receiver all alone in the end zone... TOUCHDOWN!!*

As far as sports games go for the Amiga, there really is not a big selection available when you consider how many games are on the market. As a result, when a satisfying sports simulation comes along, it is usually an instant success. GRIDIRON! enjoys this luxury; it is one of the most exciting simulations available, and it also has the benefit of being the only programmable football simulation for the Amiga on the market today.

### ON THE SIDELINES

GRIDIRON! was designed and written by Ed Fletcher and distributed by Bethesda Softworks. The digitized audio, which adds another dimension to the reality, was created by C.S. Weaver. Presently, GRIDIRON! is available for the Amiga and the Atari520ST only. The game consists of the diskette, user's manual (tactfully referred to as the playbook), and two playsheets. Also, an NFL DataDisc is expected to arrive later this year with accurate play simulations for each of the twenty-eight NFL teams. GRIDIRON! requires a minimum 512K system on your Amiga and at least one mouse. When in the two player mode, a joystick or second mouse is required. In addition to the aforementioned equipment, an AmigaDOS diskette is needed for creating a custom playbook.

### PRE-GAME HIGHLIGHTS

Though GRIDIRON! is such a powerful simulation, it is surprisingly simple to operate. All game interaction is controlled by the mouse. The keyboard is only needed when entering the Validation Sequence and when entering various names (teams, plays, and playbooks). It was purposely designed to be easily played without having to study the lengthy instruction manual. That alone gave me a lift the first time I played. I did not even start to study the playbook until I had already played the game for a couple of hours.

The screenshot shows a game interface for GRIDIRON!. At the top, it displays 'CHARGERS 10:10 GIANTS'. Below that, it shows '6 Qtr 1' and 'Down 2 To Go 9 Ball On 44'. There are also indicators for '3 time outs 3'. The yardage statistics are: '34 yards rushing 0', '46 yards passing 26', and '80 total yards 26'. At the bottom, there is a yardage scale from 10 to 50, with a '21' marker.

Remember, this is real tackle football. The clock, penalties, player abilities, and formations are all consistent with pro football. Fletcher and his crew have made a serious effort to make GRIDIRON! as realistic as possible. There are even forty plays to choose from before you create your own! You are given the option to hand off, pass, pitch, or run as quarterback. Players are rated by strength and speed. For instance, a fullback would not necessarily be tackled on the first hit, and a tight end could break a tackle much easier than a smaller, weaker wide receiver.

### IN THE HUDDLE

Upon booting the game, you must first enter a Validation Sequence. This sequence was programmed into the game in an effort to prevent theft of the software because GRIDIRON! is uncopy-protected (the codes are on pages printed in copy proof blue). Options are then offered for one or two players and the time limit of each quarter - five, ten, or the fifteen-minute standard. Playing fifteen minute quarters is my preference, but a complete game takes nearly two hours!

## Difficulty Levels

There are five difficulty levels to choose from. The basic differences between levels involve, speed, strength, and intelligence of play. The computer's or opponent's team, as well as your team, corresponds with these levels.

The Practice Level allows you to coordinate your movements with the mouse and to see how each play functions. This mode is helpful because it allows you to practice with very little defensive interaction. The defense remains stationary until the ball has crossed the line of scrimmage. And since coordinating the offense is much more difficult than controlling the defense, the Practice Level is limited to offensive plays only.

The Beginner Level is the easiest mode of an actual game. The players in this mode move slow in comparison to the other modes. The greatest advantage of this mode is that your computer opponent has a limited choice of plays and a rather simple strategy. I found the Beginner Level very easy to master after only a few games.

The Intermediate Level runs at medium speed. Your computer opponent is much smarter and the team is stronger and faster compared to your own team. Also, the computer has a wider variety of plays to choose from which complicates the strategy. The first time I played at the Intermediate Level, I was humiliated by a score of 66 to 0! I learned fast that I had better develop a more intense strategy. By the time I reached this level, I can truly say that I became addicted to GRIDIRON! and had to stop playing for a few days because I was neglecting all my other duties. This addiction is a fact that Bethesda warns of in their current ad.

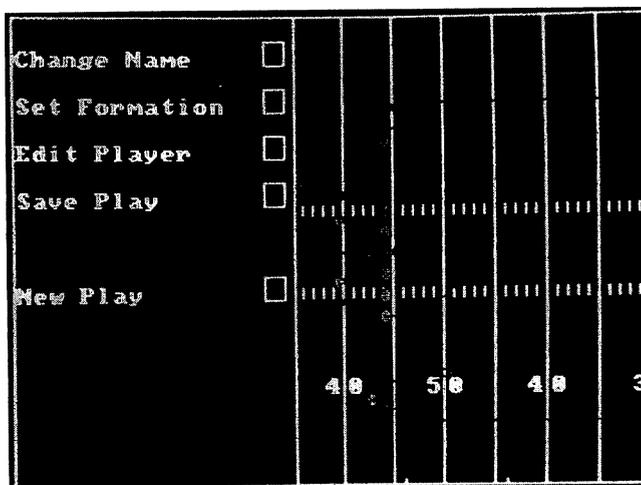
In the Advanced Level, you face the 'semi-pro' team. The opposition is lightning fast and even stronger than at the Intermediate Level. You have to be quick-witted and able to make split-second decisions in order to succeed at this level. Playing this mode, I generally end up on the losing side. I guess I have got to develop my strategy a little bit!

The Pro Level is truly for the seasoned veteran only. In this mode, the defense is superior to the offense and the computer's team has the advantages in strength and speed. It takes intense strategy and extraordinary skill to become a winner at the Pro Level. Anytime I have attempted to play a 'Pro' game, my opponent has made a fool of me. I think Tom Landry would have problems winning in this mode!

Also offered is the option to include random penalties and fumbles in the game. By choosing this option, you are forced to think more cleverly because the slightest error in your strategy could result in a fumble or an interception.

## Player Draft and the PCU

Once you have chosen the number of players, time limit, and difficulty level, you are shown the Playbook and Team Selection options. A choice is offered between using the standard team, drafting a new team, and loading or deleting a custom team. The standard team is equal to the opponent's team in Beginner and Intermediate modes, and weaker than the opponent in the Advanced and Pro modes.



If you decide to draft your own team, the variation potential is quite extensive. Each player on offense and defense is listed with a rating of strength and speed ranging from one to twenty PAP's (Player Ability Points). You are given the option to increase or decrease by any number, a player's strength and speed. These changes will not be seen on the playing field; they are only relative to the other players on the field. Now you have created a 'custom' team which can be saved on a data disk. To save a custom team, simply name the team and give it a suffix (for example, Chargers.1). This allows you to distinguish between teams with the same name on your data disk. Even if you decide not to save the custom team, making one is an effective way of handicapping between players of different experience. In other words, if you are playing against a novice opponent, you may want to "customize" a stronger team for him, or a weaker team for yourself.

Just as you can modify the abilities of a player, you are able to modify, or create custom plays and playbooks by using the Play Creation Utility (PCU). The PCU is the most powerful asset of GRIDIRON!. It allows you to leave the realm of the standard playbook and become your own coach. Before creating a custom playbook however, you must initialize a disk in AmigaDOS to use as your playbook (it may also be used for custom teams). Fortunately, the PCU will initialize any disk for you.

Modifying plays is no simple task. There are many commands which will affect the actions of an offensive or defensive player. A wide variety of offensive and defensive commands is available such as blocking in a certain direction for a run or a pass, delaying a player's actions for up to nine seconds, setting the direction for a receiver to run to, which player will have the ability to pass, hand off, punt, or kick. Those particular commands are for the offense only! Defensive players can be programmed to pursue the ball, either directly or laterally, delay actions for up to nine seconds, which direction to move toward, and what type of coverage to employ, man to man or zone. There are five different coverage zones. As you can see, the PCU is extremely flexible and powerful, but once you read the instructions, it is rather easy to use.

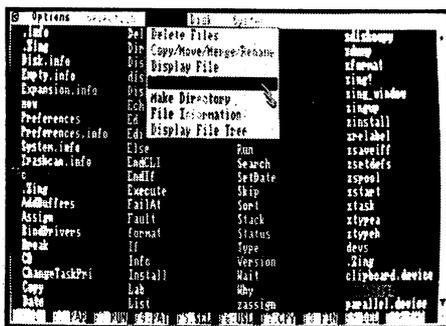
**ZING!**<sup>TM</sup> is an exciting new software package which provides a fast and powerful interface between the user and the computer. You'll be amazed at the power packaged in this little disk; yet it's so simple to use, you'll

# ZING!

<sup>TM</sup> wonder why no one else has thought of it before. You no longer have to resort to typing cryptic commands through CLI.

**ZING!** uses Intuition<sup>TM</sup> which provides you with easy window, icon, menu and mouse controlled features.

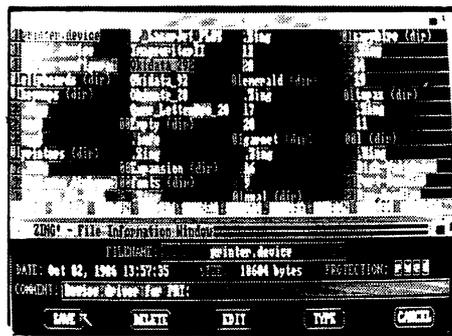
## PACKED WITH POWER!



"Sending Files to the Print Spooler"



"ZING! Hot Keys"



"Modifying a File"

## FEATURES

### Install Disks

- Copy Disks
- Relabel Disks
- Rename files
- Display a file tree
- Select files by mouse
- Select files by pattern
- Select files by time
- Set file protection

- Delete files
- Move files
- Create directories
- Change directories
- Piping of file names
- Built-in screen saver (dimmer)
- Fast Sort directory display
- Start editor with no typing

- Hot Keys
- Merge files
- Copy files
- Support full multi-tasking
- SPOOL files to the printer
- Save screens to the printer
- Save screens to IFF files
- Reassign function keys

### Format Disks

- Fancy file browser
- Monitor system tasks
- Set file comments
- Run programs from **ZING!**
- Show status of devices
- Assign internal symbols
- Show available memory.
- Copy all or PARTIAL file trees

**ZING!** offers these and hundreds of other capabilities without preventing you from running other applications simultaneously. Order **ZING!** and transform your mild mannered CLI into the fastest and most powerful computer interface ever conceived! It's available now for the **special introductory price of**

**\$79.95** plus \$3.00 shipping and handling.

**MERIDIAN<sup>TM</sup>  
SOFTWARE  
INC.**

P.O. Box 890408  
Houston, TX. 77289-0408

ZING! is a trademark of MERIDIAN SOFTWARE, INC.  
AMIGA is a registered trademark of Commodore-AMIGA, Inc.

(713) 488-2144  
Credit Cards and  
Dealer Inquiries Welcome!

Even when you create your own plays, you are still governed by official rules. Players cannot line up offside and linemen are not eligible as receivers. I have created a few of my own trick plays using the PCU and most of my plays are alright, but on one of them I decided to do a little razzle-dazzle, and each time I use it, I get penalized ten yards for having an illegal receiver downfield. Fine offensive coordinator I would make!

### PLAY BY PLAY ACTION

Now that all your decisions have been made, you are ready to begin playing. Your first impression of the graphics may leave you wanting for more, but after playing it a few times you will understand why the graphics are designed in such a plain fashion. Why waste all the time staring at fancy graphics when your main concern is strategy? You need a clear view of all players at all times if you want to keep abreast of the action.

Between each play a large scoreboard with two smaller boxes at each is displayed. The scoreboard carries vital information for each player. The score, quarter, time remaining, time outs, and total yards broken down into number gained by rushing and by passing is displayed. The smaller boxes are used by the player(s) when selecting plays. The offensive box also has a section for hiking the ball. When selecting a play, the offensive controller must always wait at least eleven seconds to allow the computer to set the formations. All plays are timed by a thirty second clock located at the bottom of the screen. If you click within the thirty second clock an officials time out is called. This option is provided to allow the players a chance to enter the PCU, view statistics, or to make a fumble run. It is beneficial if you have to leave the computer for some reason because it stops play without forcing you to use one of your time outs.

As the game progresses, listen to some of the audio effects provided. They are magnificent. The game starts with the national anthem. The crowd cheers. The referee blows his whistle. The quarterback yells out the signals before the snap. The announcer can be heard addressing a first down or a touchdown. You can even hear the kicker and the punter as they boot the ball skyward. The greatest thing about these effects is that they sound incredibly accurate - complete with echos.

Controlling player movements on either the offense or defense is easy, but requires a little practice and coordination. On offense, you are automatically assigned control of the quarterback. Once the QB has advanced the ball to a second player, you gain control of that particular player. To make a player move upfield, simply lead him with the pointer. On defense, only one player is controlled by the mouse, but you have a choice of which one it will be by simply clicking the mouse button over that particular player just before the snap. The mouse controlled player is distinguished by either a white diamond or the football.

All field functions are controlled by the mouse. When passing, you must aim the pointer in the direction of a receiver and at the length you desire in order to achieve a

completion. Try to be accurate because you must keep in mind that receivers have momentum and can easily overrun your pass. Handing off is easier because there is not as much guesswork involved. But you must be sure that your running back is moving in the direction that your quarterback is handing to.

Game simulation is quite remarkable. All twenty two players move independently on the field. Defensive players are not controlled by the mouse when making a tackle and offensive receivers run patterns and will follow passes on their own. All these features are great because they help maintain the continuity of play and make the game even more realistic. After all, every professional football player can function on his own.

### POST GAME COMMENTS

GRIDIRON! is not the only football simulator I have tested for the Amiga. I also tried the newly released GFL Championship Football by Gamestar. It would not be fair to compare the two as GRIDIRON! excels in strategy while GFL Football excels in graphics. A preference between games depends strictly on what type of format you favor.

In my opinion, GRIDIRON! is a fantastic football simulator. You may not be impressed by its chalkboard style graphics, but you have to be amazed at its ingenious programming abilities. I have been a football fanatic for years and I know nearly everything about the game, and GRIDIRON! really impresses me with its accuracy. One of my favorite features is the option that allows me to check statistics as I play the game.

Even though I am impressed by GRIDIRON!, I do question a couple of things. The random penalties and fumbles usually seem to occur against my team when I play against the computer. I have yet to see the computer team commit an offensive penalty, but occasionally it is cited for a defensive foul. Maybe the computer team is less fallible in an attempt to make it more superior. Also, I seem to lose the introductory coin toss more often than my computer opponent. But these are just minor quirks that do not affect the game negatively in any way. In fact, they merely pose a greater challenge for my team.

The difficulty levels on GRIDIRON! are superior to those of many other games because they are so differentiated. The Beginner Level is great fun for two players with little experience while the Pro Level seems to be a challenge for even the most seasoned veteran.

At the present, GRIDIRON! is the best football simulator available for your Amiga. The only possible improvement that I believe could be made would be the introduction of full screen graphics such as of a huddle, kickoff, or touchdown. But the game endures well without such additions.

GRIDIRON! is not a simulation that one can easily master and eventually outgrow because the only limit is your own imagination.

•AC•

## Amazing Reviews...

# Star Fleet I

Version 2.1

by John Tracey

One of the latest entries in the steadily-building line of Amiga software is Interstel's STAR FLEET I. This program is more than just a game. STAR FLEET I is the first in a series of complete space career simulations.

The object of STARFLEET I is to advance in rank in the United States Galactic Alliance. The budding "STAR FLEETER" advances by successfully completing missions assigned by the Star Fleet Command. The disk does not have Workbench on it, but you can quickly install it, making a bootable copy. You may install an "SF1" command in the startup-sequence so that the program runs automatically when you warm boot the computer.

### WELCOME TO THE STAR FLEET COMMAND

The first time you boot Star Fleet I, it asks you for your last name and if you are a new recruit (If you are a new recruit, you then enter your own personal password of eight letters or less). The sign-on options appear on the screen next. Here you may choose to view your (S)ervice record, check your current standing towards (P)romotion, (R)esume a previously saved game, or (C)ontinue your mission, among other miscellaneous options.

When you choose to (C)ontinue, you will be asked for your mission level, which is essentially a difficulty scale from one to ten (one being cadet, ten being Admiral). If you are a cadet, you can choose only level one, whereas if you are an Admiral, you may pick any level you wish. After each mission you are given a Performance Rating. You must have a rating of 75% or higher to be officially promoted. If you successfully complete five missions with an overall rating of 75%, an elegant screen will appear listing your promotion with "Pomp and Circumstance" playing in the background. Quite unique and flattering.

You may also receive official awards for outstanding mission ratings. These awards include a Combat Action Ribbon, a Silver Palm, and an Iron Cross. Also, if you command self-destruction of your StarShip when all hope is lost, you may be awarded the Alliance Defense Cross with Gold Star ... but the computer will decide whether your decision was justified or just plain cowardly!

### BLAST OFF!!

Now on to the game play. At mission levels from Lieutenant Junior on up, you choose the starship you will command. If a StarShip was destroyed in a previous mission, it will be outlined in red, and will be unavailable until repair is made. You also choose the length of your mission and whether or not you want a Captain's log recorded for future viewing. (This is like videotaping your game.)

Next, your official Star Fleet Command orders appear on the screen. Your orders outline how many StarBases are in your area, how many aliens you must eliminate, and how many Stardates you will be allowed to complete your mission. The aliens are a mixture of Krellans and Zaldrans, the Zaldrans being the more difficult to destroy. You'll find out why in a minute.

Your starship is equipped with both short and long range scanners, shields, torpedoes, phasers, mines, transporters, tractor beams, and even probes for extending scanning. There are also two life-support systems and, of course, main and auxilliary engines. The screen is divided into several sections. There is a region map, displaying the entire star system in an 8 by 8 grid. There is also a tactical "sector" display which consists of the 100 positions, where all local maneuvering is done. Finally, your chance at survival is enhanced by a warning message indicator and StarDate gauge. The scene is completed by an outline of the ship and the current status of the shields. This scene will instantly remind you of Star Trek with Kirk commanding "Shields Up!!"

At the bottom of the screen is the command menu. The menu consists of 10 primary commands, 10 secondary commands, and 6 computer commands. Navigation can be controlled in two ways: You can take the lazy man's route by using the built-in target calculator or you can take matters into your own hands by steering manually using the NAVcommand, a compass direction and a C-factor (distance).

Damage control is one of STAR FLEET I's most effective features, especially since your ship has thirteen different systems to maintain! When you choose DAMAge control, the onboard computer prints a chart of all the systems and their status. If any systems are inoperational, you will be given the option to repair. Be careful, though! Repairing a system uses up both time and precious power. When you

*continued...*

**You lock your car.  
You lock your door.  
Do you lock your data?**

*MagiCode is an encryption system  
MagiCode protects:*

<i>Data files</i>	<i>Software files</i>
<i>Source code</i>	<i>Picture files</i>
<i>Text files</i>	<i>Executable</i>

Send check or money order for \$30.00 to:

**Magic Circle Software**

37 Ralidne Road  
Hyde Park, MA 02136

Specify Kickstart V1.1 or V1.2 512K recommended.  
MA residents add 6% tax.  
Please allow 3-6 weeks for delivery  
Dealer inquiries welcomed.

begin your mission and after docking at a spacedock (on a StarBase) you have 5000 units of power. Depending on the system, repair may take anywhere from 100 units to 600 units of power. The basis of your survival is your power level, so it is vital to limit repairs and to know where a starbase is in case you really need it.

### **MAN YOUR WEAPONS!**

Now for what you've been waiting for: the battles! As already stated, there are two enemies of the United Galactic Alliance, Krellans and Zaldrons. Your arsenal consists of phasers and torpedoes. You select either automatic or manual fire control. Phasers are more accurate, but use lots of power; torpedoes are great at the lower levels, but as you increase in rank the aliens move constantly, so your precious torpedoes may sail harmlessly off into the galaxy. You start each mission with twenty torpedoes, but by laying mines, restocking at starbases, and retrieving those mines, you can build your arsenal up to thirty torpedoes. You choose how many to fire, because it often takes more than one to blow up an enemy ship.

Your primary objective is to eliminate a certain number of aliens per game. Your mission percentage rating will be increased, however, if you capture aliens and their ships and drop them off at StarBases, rather than just blowing them away! When you are firing phasers, the computer asks you if you wish to disable any targets. If so, the phasers will reduce the power level of the ship rather than destroying it. Once the enemy ship is disabled, you may transport your seventy marines aboard to seize the ship. If you are successful, the marines will transfer prisoners and power to your ship. You then use your tractor beam to capture the vessel and drop it off at the nearest base.

The Krellans are fairly easy to destroy, but they are almost always in groups. The Zaldrons require much more strategy and space savvy---after all, they are invisible. Your ship's Auto-alert feature will detect them, so they cannot just sneak up on you. The manual states that invisibility requires much energy so after a while, the Zaldrons will appear, making them easy targets. If you don't have the time to wait for the Zaldrons to appear, you can pick-and-shoot, manually firing torpedoes and phasers randomly.

Often, a StarBase under attack will call and ask for your help. If you save the wounded StarBase, your percentage improves greatly. You will also be congratulated by Star Fleet Command.

### **STAR FLEET I EXTRAS**

STAR FLEET I also has many different game options. You can save a game in progress, initiate an emergency hyperspace maneuver, or ask for a status command. This status command feature updates you on your projected rating, days remaining, marines remaining, condition of the aliens in your sector, number of inoperative systems, shuttles remaining (used to recover mines), prisoners and number of probes remaining.

The STAR FLEET I disk also has the advantage of being uncopied-protected. This feature is important because the game keeps track of many players, and if a disk becomes full, errors can occur, and bang the disk is "corrupt." I suggest that you Diskcopy STAR FLEET I immediately.

Even the packaging of STAR FLEET I is superb. The Amiga version comes with an Officer's Manual, Training Manual, Quick Reference Card, and Amiga supplement. The manuals make up over 170 pages of useful and entertaining information.

Now that the game description is done, I must say that there are some problems with Star Fleet I. I worked with version 2.1 (by the way, the warranty card promises future upgrades). When you attempt to set the target calculator (navigation device) by the keyboard, the system crashes. When you attempt to use Auto phaser control with the invisible Zaldrons, the system crashes. Also, for no apparent reason, the system will simply freeze during the sign-on or save mode. A quick look at the Workbench screen reveals a "Software error" window. The only solution is to reboot. After playing for two hours, losing your ninety percent rating to a system error is very frustrating.

### **AN OVERVIEW**

Despite these minor faults, Interstel Corp. did a first-rate job with STAR FLEET I. From the graphics and sound, to play action, to documentation, STAR FLEET I is extremely well done. The game cannot be mastered quickly, thus it will hold your interest hour after hour. I highly recommend Star Fleet I for anyone who enjoys a Star Trek-like space adventure. After all, this is only Star Fleet I....Who knows what Star Fleet II will be like?

•AC•

**Star Fleet I \$55.00**  
*Requires 512K*

**Interstel Corporation**  
P.O. Box 57825  
Webster, Texas 77598  
(713)333-3909

# The Household Inventory System

in AmigaBASIC™

by Bryan Catley

**The Household Inventory System** is a collection of three inter-related AmigaBASIC™ programs which provide the means of maintaining an inventory of household possessions, along with the capability of updating that inventory and printing inventory reports in a number of possible sequences. Program #1 is the main program which provides for the entry and maintenance of inventory items, while Program #2 allows reports in any of three sequences, and Program #3 is a file maintenance program which is used to maintain the actual files of data; (as against the data itself). All three programs provide ample online help facilities.

A little more specifically, Program #1 allows for the entry of individual items, browsing through the file of data, and selecting any record for modification or deletion. When deleted, a record is tagged as such but is actually still available (and may be restored) until the file is reorganized. Program #2 prints inventory reports in entry, room, or item sequence, and Program #3 provides a number of file maintenance options such as reorganization, displaying the count of records in the file, initialization, etc.

Program #1 uses just about all of the memory available to Basic programs which explains why the other two functions were not included in that program! Further, since a second program was needed (at least), why not separate the major functions into distinct programs? Hence, there are three of them!

## **Entering the Three Programs**

Should you choose to enter the three programs from the reproduced listings accompanying this article, there are a couple of things you should note. *(Please be aware these three programs are also available on an Amicus™ disk should you prefer not to type in the rather lengthy listings. See elsewhere in this magazine for more information regarding Amicus™ disks).*

There are a number of subprograms which are common to all three programs and there is simply no need to enter them all for each program. So, before you get into entering the three main programs, enter Listing #1. This listing is composed of the common subprograms, and when you have finished entering it, be sure and save it with the "A" option. For example, SAVE "subs",A. The "A" option causes

the entered subprograms to be saved in an ASCII format which will subsequently allow them to be MERGED with the other three programs when they are entered. Note that you must enter an immediate SAVE command to accomplish this. The SAVE and SAVE AS menu items do not allow the use of this option.

Once you have successfully entered this listing, proceed with Listings #2, #3, and #4. These three listing are the main portions of the three programs under discussion. When each has been entered, click in the Output window and enter the immediate command MERGE "subs", and then save the merged programs with an appropriate name. How about HInvMain, HInvPrint, and HInvMaint? If you receive an error message after entering the merge command, the chances are you forgot to save "subs" with the "A" option. This is no great problem. Just save what you have been working on, load "subs" (or whatever you called it), re-save it with the "A" option, load what you were working on previously, and issue the merge command again. When each of the programs has been completed, be sure to save it before you try to RUN (or execute) it.

One thing you must be aware of is that memory really is tight (if you have a 512K machine) with the main program. If you have been doing a lot of editing of other programs, it is quite probable there will not be enough memory available for the program to execute successfully. Therefore, if you receive an "out of memory" message, you should not presume there is an error with the program. However, you will probably find that you have to reboot the machine in order to get the program to run correctly. For better or worse, this is the only way to free some of the memory that is allocated during a number of functions (such as cut and paste), and is never freed (presumably to save having to load it again). Anyway, once all three programs have been entered and saved, you are ready to start using them.

## **Using the Household Inventory System**

Before we get into the whys and wherefores, a couple of general comments are in order. The major purpose of any system such as this is to keep a record of your household inventory for insurance purposes in case of burglaries, fire, etc. This being the case, it does not make a lot of sense to keep that record in the house itself! We strongly recommend that you keep this record on a separate disk, and that you

*continued...*

## Haven't You Set Your *AMIGA'S* Time And Date Once Too Often?

Introducing

# A - T I M E

*A clock/calendar card with battery back-up,  
so you will never have to set the time and date  
in your AMIGA, EVER AGAIN!*

- Plugs into the parallel port.
- A completely transparent printer port is provided, with total compatibility to all I/O operations.
- Battery back-up keeps the clock/calendar date valid on power down.
- Custom case with a footprint of only 2 1/4" x 7/8" x 3 1/4" (W x D x H) in standard *AMIGA* color.
- Leap year capability.
- A - T I M E package contains:
  - 1-A - T I M E clock/calendar module
  - 1-3.5" DS Utilities Disk
  - Operating instructions

PRICE \$59.95

AVAILABLE: NOW

Mail check to:

**AKRON SYSTEMS DEVELOPMENT (ASD)**  
P. O. BOX 6408 (409) 833-2686  
BEAUMONT, TEXAS 77705

include \$3.50 for shipping and handling  
For MCVISA orders call (409) 833-2686

*AMIGA* is a trademark of Commodore - Amiga inc.

extensive use of many of the Amiga's special features and follows what is emerging as the "standard" Amiga user interface. The result is that a lot of memory is used! In fact, you will need at least 512K of memory, and even then things will be a little tight. What this means is that you will probably be unable to have any other task, such as a clock, running in the background, and if you have previously used some Amiga functions which load disk based routines in to memory (such as speech, cut and paste, etc) you may even find you have to reboot the machine. Some memory usage notes follow:

- The default Basic program memory allocation of 25000 is just, and only just, sufficient to hold the program.
- The first thing the program does is to give itself some more memory for the arrays it will dimension. This, in turn, reduces the amount of system heap space available for the high resolution screens and windows, and the file buffers.
- The actual amount of memory required depends on the number of records in the inventory file. By adjusting the SIZE command at the beginning you may increase or decrease the size of the program space as it becomes appropriate. Remember, as the program space increases, the heap space decreases, and vice versa.

When you run the program, you will be presented with a title screen, and a request to enter the date. Should you press the right mouse button you will also see you have three menus available to you: Project, Help, and Data Files.

If this is the first time you have used the programs, you should do as the screen suggests and read all the Help menu items. You may also want to select Initialize under the Data File menu to create your inventory file. Should you not do the latter yourself, and no inventory file exists, the program will create one for you automatically a little later. Normally, this will only happen once but it does provide you with the means of starting over should you ever wish to.

While on the subject, let's take a closer look at the Project and Data File menus; (the Help menu is self explanatory). The Project menu has a single item: Quit. And it does just that, terminating the program and returning control to Basic. The Data File menu has the following items:

**Initialize.** A new and empty inventory file established. If a file presently exists, you will be asked if it is OK to delete it before the new one is created.

**Update Count.** The program keeps track of the number of records in the inventory file by means of a single record "count" file. When adding records to the file, this count file is NOT updated with every single addition; instead, an internal counter keeps track of the number of records being added, and the "count" file is only updated when you select a new function or quit the program. The purpose of this menu item

keep a copy of that disk in a safe-deposit box, or somewhere equally safe. Once you have initially created your inventory file, you will probably find it appropriate to update that file once or twice a year, replacing the disk in the safe deposit box at the same time.

To create this separate disk, simply initialize (or Format) the disk you wish to use, and copy Amiga Basic and the three programs onto it. If you wish, use this disk in place of your normal Basic disk to enter the programs originally. In fact, this is a good idea because it keeps everything separate right from the start. Don't forget to make backup copies of this disk along with the copy you will be placing in the safe-deposit box!

You will find the programs easy to use. All have help menus which are available at all times, and which should be read in their entirety the first time each program is used. Beyond this, other menus and various gadgets provide a very straightforward working environment. Let's take a look at each of the three programs so you will know what to expect when you actually use them.

### **The Data Entry and Maintenance Program**

This is the main, and largest, of the three programs. Its purpose is to provide you with the means of entering new inventory items, and updating (changing and/or deleting) items you have previously entered. In doing this, it makes

is to allow you to force the "count" file to be updated from time to time when adding a substantial number of new records. This provides a degree of protection in case of a power failure, system crash, etc.

**No. of Records.** This item simply provides you with the current count of committed and non-committed records. To commit the non-committed records, use the Update Count item described above.

**Now it's time to enter the date as requested.** The request is to enter it in form mm/dd/yy (e.g. 09/01/87 for Jan 9, 1987), however what is actually entered is not edited, but is simply used to identify when records in the file have been added and/or updated. So you may really enter whatever you wish, except for a null value. Once the date has been accepted, three gadgets will appear on the screen and you are into the meat of the program. The three gadgets are:

**Add.** Selecting this gadget will allow you to add new inventory items to the existing file. Once selected, an "Add a New Item" window will open and you will be prompted for the name of the item, the room the item belongs in, the date of purchase, its original cost, its current worth, its replacement cost, its serial number, and any comments you may wish to include. Note that only the first two items are required, the remainder may be left blank or updated at a later time. While entering information, you may select from Cancel, OK, and Modify gadgets at any time.

Cancel does just that. The current item is forgotten, and you are asked if you wish to enter another item, or if you wish to Quit.

OK accepts the item you have entered and adds it to the inventory file.

Modify allows you to update the item you are entering before adding it to the file (should you enter an incorrect piece of information, for example). Once selected, you will be presented with a window which displays each piece of information included with the record in question. To update a piece of information (correct or add), just click in that item's box and then edit the existing information or enter the new information.

Once you have used any of the above three gadgets, you will then be given a choice of another two: Add More and Quit.

**Add More.** This selection allows you to start entering a new item into your inventory.

**Quit.** Selecting this item will cause the record count file to be updated (thereby committing all the records you have just entered), and will subsequently return you to the title screen.

Note that dates are not entered manually, nor are they updateable. Dates are handled automatically, using the date you entered when you first initiated the program.

## 1 MEGABYTE - \$119.95

**SQUEEZE-RAM** is a 1 Megabyte internal RAM board for your Amiga. Because it goes inside, it leaves your expansion port free for Sidecar or an external Zorro box. It auto-configures under AmigaDOS 1.2, and there is a memory disable switch option. It is compatible with external memory boards, so you can add up to 8M more. There is no modification of your Amiga necessary for installation. We sell it in kit form to save you money... assembly and installation only takes an evening. Kits include high-quality PC board, all parts including sockets for all IC's, utilities disk and manual.

**SQUEEZE RAM Kit w/o RAM Chips: \$119.95**

Requires 32 4464-150 ns RAM Chips  
Call us for RAM Chip sources and prices

Please add \$3.50 for UPS Ground shipping  
Calif. residents please add 6 1/2% sales tax

**Aminetics**  
P.O. Box 982-205  
Whittier, CA 90608  
(213) 698-6170

**Review:** Selecting this option will cause the entire inventory file to be read into memory. Once this has been done, you will be presented with a full sized window which displays the first 18 records in the file, along with some further information about the file, a scroll bar on the right hand side of the screen, and a "Finished" gadget in the upper right corner. Selecting this latter gadget will return you to the title screen (the internal representation of the file is also erased).

The scroll bar consists of upper and lower arrow-heads with a "bar" between them. Clicking on the arrow-heads will cause the display to move up or down by a single record. The bar is used to display the position of the 18 displayed records relative to the entire file. The bar itself is colored blue, spans the space between the two arrow-heads, and represents the full file of inventory records. The display of the 18 records on the screen is represented by a smaller white bar which is superimposed on the blue bar. The size of this white bar is the same percentage of the blue bar as the 18 records are of the number of records in the file. Thus, if the file consists of 18 records or less, the white bar will cover the blue bar in its entirety, but the more records there are in the file (over 18) the smaller in length the white bar will become. So, by clicking in the blue portion of the bar, you may choose any set of 18 records from the file to display (although you may have to fine tune with the arrow-heads).

*continued...*



## Introducing Robot Readers a powerful new way for your child to learn to read

Even if your child isn't a reader yet he can read these classic stories at his own speed through interactive speech. And he can play a game that builds vocabulary and reading ability. These beautifully illustrated stories are designed to be used by children with little or no help. More stories will soon be available. To introduce the series and help build a library for your children we make this  
**LIMITED TIME OFFER: Buy one, get one free**

- |                    |                     |
|--------------------|---------------------|
| *CHICKEN LITTLE    | <b>\$29.95 each</b> |
| *LITTLE RED HEN    | for the Amiga 512k  |
| *AESOP'S FABLES    | call or write today |
| *THREE LITTLE PIGS | (specify titles)    |

HILTON ANDROID CORPORATION  
PO Box 7437 • Huntington Beach, CA 92615-7437  
(714) 960-3984

Now that you can browse through the file, what can you do with the records? Well, if you wish to do something (like modify or delete) with one of the records on the screen, just click on it!

You will receive a display of the record with each field contained within a box. There will also be three gadgets to choose from: Modify, OK, and Delete. If you choose Modify, you will be requested to click in the field you wish to update, and you may then update that field. When all is well, select the OK gadget.

Should you wish to discard the record, just click in the Delete gadget. You will then be asked if this is really what you want to do (just to be on the safe side), and if you select yes the record will be "logically" removed from the file. What this means is that the record is "marked" for deletion, rather than actually (physically) being deleted. Subsequent reviews of the inventory file will show deleted records with asterisks between the fields of information. If you click on one of these records, you will be told the record has been deleted and requested if you wish to restore the record to the file. Just select the Yes or No gadget! Note that logically deleted records will not be physically deleted from the file until it is reorganized. See a little later in the article for more information.

Selecting OK will return you to the original display. Please note that the "Review" display will always reflect the file as it

existed when Review was originally selected. Deletions or changes you may have made will not show up until the file is re-reviewed at a later time.

**Quit.** Selecting this gadget provides the same result as selecting the Project/Quit menu item: the program terminates and control is returned to Basic.

### **The Print Reports Program**

This program has no special memory requirements, is entirely menu driven, will provide you with inventory reports in any of Entry, Room, or Item sequence, and will work with any printer supported by the Amiga.

Once invoked, you will be presented with a title screen which is similar to that of The Data Entry and Maintenance Program however, there will be no gadgets to select and you must use the three menus.

The Project menu has a single Quit item which, when selected, will terminate the program and return you to Basic. The Help menu provides information about the options available under the third menu, and should be read in its entirety before actually printing a report. The final menu addresses the main purpose of the program and allows you to produce printed reports of your inventory in the sequence in which you entered them, in sequence by the room in which they reside, or in sequence by the items themselves.

Probably the most important thing to note is that Order Entry reports may only be produced before either of the other two reports are produced. In fact, once a report in room or item sequence is requested, the Order Entry option is made non-selectable. (If the inventory array is sorted, as it must be for room or item sequence reports, it becomes impossible to return it to its original order entry sequence without reloading it from disk).

It may also be appropriate (even if a little obvious) to mention that a printer must be attached to your Amiga in order to make effective use of this program.

### **The File Maintenance Program**

Like the report program described above, this program has no special memory requirements, and is entirely menu driven. Once again there are three menus: The Project menu has a Quit item for terminating the program. The Help menu provides information about using the third menu. While the Maintenance menu itself provides a number of items designed to assist in maintaining the file of inventory items. Note that this program is more concerned with maintaining the inventory file itself, rather than the data which is contained therein.

Within the Maintenance menu, you will find the following items:

**Initialize.** This repeats the function available in The Data Entry and Maintenance Program. It effectively deletes any existing file of inventory items and allows you to start from scratch, should you so desire.

**Reorganize.** This item will only be necessary after a number of records have been "logically" deleted from the inventory file, and the time has come to actually and physically remove them from the file. This is done by copying the file and dropping the "logically" deleted records during the process. Please note that once a file has been reorganized, it becomes impossible to reclaim logically deleted records.

**Record Count.** This an informational item which simply provides you with a count of the number of inventory records on file.

**Update Count.** This item will only be required if something out of the ordinary happens to your inventory file and it gets out of step with the count file. The system uses this single record count file to keep track of the number of records in the inventory file. As a general rule, this one record file will accurately reflect the number of inventory records being carried. However, it is conceivable that the two could get out of step as the result of some unexpected event, (such as a power failure). Should this ever happen, just use this menu item to reset the count file.

This program will probably not require a lot of use, but don't forget you have it. And if you do delete records from time to time, please don't forget to reorganize the file occasionally!

### Programming Notes

Most of the "interesting" programming techniques take place within the various subprograms which are a part of the three programs. The three Gadget subprograms and the Getp subprogram are prime examples. Further, these subprograms have all been explained in some detail in previous issues of Amazing Computing. So, if you would like more information about these subprograms, please refer back to the more recent issues of AC.

Elsewhere, variations on the basic gadgets have also been employed and they all appear in the Review window!

First, the area occupied by each record which is displayed is defined as a gadget, but is never actually drawn. This allows us to check for "clicks" on individual records without having actual gadgets on the screen!

Second, the scroll bar on the right side of the screen is really three individual gadgets. The top and bottom "arrow heads" are square gadgets which are drawn with the interior and border in the background color (so you can't see them). Immediately following this, the triangles which you see are drawn independently by the main program, in white. What this means is that you can really click anywhere in the square rather than just in the triangle! The middle rectangle is a "normal" gadget with the white bar being superimposed on it, once again, by the main program. The size of the white bar, in relation to the blue rectangle, represents the number of records displayed, in relation to the total number of records in the file. Further, the location of the white bar within the gadget represents the location of the records displayed within the overall file. To accomplish this, the

*continued...*

ATTN:  
PASCAL  
USERS

# MODULA-2

## the successor to Pascal

- FULL interface to ROM Kernel, Intuition, Workbench and AmigaDos
- Smart linker for greatly reduced code size
- True native code implementation (Not UCSD p-Code or M-code)
- Sophisticated multi-pass compiler allows forward references and code optimization
- RealIn/Out, LongIn/Out, In/Out, Strings, Storage, Terminal
- Streams, MathLib0 and all standard modules
- Works with single floppy/512K RAM
- Supports real numbers and transcendental functions ie. sin, cos, tan, arctan, exp, ln, log, power, sqrt
- 3d graphics and multi-tasking demos
- CODE statement for assembly code
- Error lister will locate and identify all errors in source code
- Single character I/O supported
- No royalties or copy protection
- Phone and network customer support provided
- 350-page manual

Pascal and Modula-2 source code are nearly identical. Modula-2 should be thought of as an enhanced superset of Pascal. Professor Niklaus Wirth (the creator of Pascal) designed Modula-2 to replace Pascal.

### Added features of Modula-2 not found in Pascal

- CASE has an ELSE and may contain subranges
- Programs may be broken up into Modules for separate compilation
- Machine level interface
  - Bit-wise operators
  - Direct port and Memory access
  - Absolute addressing
  - Interrupt structure
- Dynamic strings that may be any size
- Multi-tasking is supported
- Procedure variables
- Module version control
- Programmer definable scope of objects
- Open array parameters (VAR r: ARRAY OF REALS;)
- Elegant type transfer functions

Ramdisk Benchmarks (secs)	Compile	Link	Execute	Optimized Size
Sieve of Eratosthenes:	6.1	4.9	4.2	1257 bytes
Float	6.7	7.2	8.6	3944 bytes
Calc	5.7	4.8	3.6	1736 bytes
Null program	4.8	4.7	—	1100 bytes

<pre> MODULE Sieve; CONST Size = 8190; TYPE FlagRange = [0..Size]; FlagSet = SET OF FlagRange; VAR i: FlagRange; Prime, k, Count, Iter: CARDINAL; BEGIN   FOR Iter:= 1 TO 10 DO     Count:= 0;     Flags:= FlagSet(); (* empty set *)     FOR i:= 0 TO Size DO       IF (i IN Flags) THEN         Prime:= (i * 2) + 3; k:= i + Prime;         WHILE k &lt;= Size DO           INCL (Flags, k);           k:= k + Prime;         END;         Count:= Count + 1;       END;     END;   END; END Sieve. </pre>	<pre> MODULE Float; FROM MathLib0 IMPORT sin, ln, exp, sqrt, arctan; VAR x,y: REAL; i: CARDINAL; BEGIN (*ST-,\$A-,\$S-*)   x:= 1.0;   FOR i:= 1 TO 1000 DO     y:= sin (x); y:= ln (x); y:= exp (x);     y:= sqrt (x); y:= arctan (x);     x:= x + 0.01;   END; END float. </pre>
<pre> MODULE calc; VAR a,b,c: REAL; n, i: CARDINAL; BEGIN (*ST-,\$A-,\$S-*)   n:= 5000;   a:= 2.71828; b:= 3.14159; c:= 1.0;   FOR i:= 1 TO n DO     c:= c*a; c:= c*b; c:= c/a; c:= c/b;   END; END calc. </pre>	

### Product History

The TDI Modula-2 compiler has been running on the Pinnacle supermicro (Aug. '84), Atari ST (Aug. '85) and will soon appear on the Macintosh and UNIX in the 4th Qtr. '86.

**Regular Version \$89.95 Developer's Version \$149.95 Commercial Version \$299.95**  
 The regular version contains all the features listed above. The developer's version contains additional Amiga modules, macros and demonstration programs - a symbol file decoder - link and load file disassemblers - a source file cross referencer - the kermit file transfer utility - a Modula-2 CLI - modules for IFF and ILBM. The commercial version contains all of the Amiga module source files.

### Other Modula-2 Products

Kermit	- Contains full source plus \$15 connect time to Compuserve.	\$29.95
Examples	- Many of the C programs from ROM Kernel and Intuition translated into Modula-2.	\$24.95
GRID	- Sophisticated multi-key file access method with over 30 procedures to access variable length records.	\$49.95

**TDI SOFTWARE, INC.**

10410 Markinson Road, Dallas, Texas 75238, U.S.A. (214) 340-4942  
 Telex: 888442 Compuserve Number: 75026-133

# 20 Meg Hard Drive

For the Commodore Amiga™

Uses parallel port  
Parallel printer still works  
Simple Startup File Installation  
Backup, Physical Format, and Park  
programs included

Only \$895.00

From Jefferson Enterprises  
2043 N. Schell Drive - Phoenix,  
AZ 85023 Phone (602) 993-4009

# THE BLACKJACK



## Simulator

Now play **BLACKJACK** on your AMIGA just as if you were in Nevada. Deals up to nine players using up to nine decks. This program actually analyzes and reports on your progress during the game so you can mathematically build your own system of betting and winning. Doubles and splits up to four pairs while playing all nine hands on the screen at the same time. This program is easy enough for children to play and is ideal for helping them with their addition. \$24.95

**POKER** — Your AMIGA will play the roll of up to four saloon card sharks who are after all your money. Playing against these four will truly get you warmed up for a real game of Poker. Call your game when your time comes around to deal. \$ CALL

**SOLITAIRE** is easy to play when you have a mouse to move all the rows of cards around and a computer to organize them for you. Only one problem with this program, it won't let you cheat. \$14.95

All programs require 512K and have built-in instructions that you can recall at anytime during operation.

Free shipping in the U.S. Dealer inquiries invited.

Send check or money order to:



THE SOFTWARE FACTORY

4574 Linda Vista  
Boise, ID 83704  
(208) 322-4958

Amiga is a trademark of Commodore-Amiga, Inc.

GetGadget sets a global variable to the "y" value of the mouse click. The DrawBar subroutine then uses this value to determine exactly where to draw the white bar within the gadget.

Thirdly, the inventory file itself is a relative file which is always read from, or written to, via a relative record number. Since the number of records in the file is continually variable, the three programs need to be able to keep track of exactly how many records have been stored at any one time. This is done by having a second file which contains a single record which indicates just how many records are in the main file. This, in turn, makes it impossible to read beyond the end of the file. When a record is deleted, a delete flag is set within the record. Thus, the record is still available, (and may be reinstated), until the file is reorganized when the deleted record is actually removed from the file. If we didn't do this, we would have no alternative but to reorganize the file each time a record was deleted, and this is really not a very good idea, especially when the file starts to become fairly large!

The only other item of note in this area is that the count file is only updated when the "Add" function is terminated, or when the appropriate menu item is selected. Doing it after every record addition adds a tremendous amount of overhead and is simply not worth it. This does open the door to allowing the count and data files to get out of step. However, this should not happen very often, but if it does, the Update Count menu item of the File Maintenance Program will take care of it!

Other than what we have just mentioned, the programs are really very straightforward, and a little time spent studying the listings should provide a complete understanding of how the programs works.

### Some Final Comments

If you have been looking for a sound, automated method of storing your household inventory for insurance purposes, or if after reading this article, it sounds like a good idea, I'm sure you'll find these three programs very useful. Further, if you are at all interested in Basic programming, studying these programs should also prove to be very educational.

As I'm sure you can see, a substantial amount of time and effort went into developing these three programs, and for that reason I would ask that you do not modify the title screen in any shape or fashion. Thank you, and enjoy!

P.S. This system was written and tested using the version of AmigaBasic which accompanies Workbench 1.1. As of this writing, I have only just received my copy of Workbench 1.2, and while all three programs appear to function correctly, no exhaustive testing under 1.2 has been performed. However, it does appear that AmigaBasic 1.2 uses a little more memory than the previous version. This, in turn, means that memory sensitive programs (such as the main program here), become even more sensitive. Just beware!

## Listing One

```

'
' These sub-programs are common to all three programs
' making up
' The Household Inventory System.
'
' Bryan D. Catley
' January 1987
'
SUB Logo80 (Depth%) STATIC
SHARED Blk,Blu,Grn,Cyn,Red,Mag,Yel,Wht
IF First=0 THEN
  First=1
  SCREEN 1,640,200,Depth%,2
  WINDOW 2,,16,1
  COLOR ,0:CLS
  PALETTE 0,0,0,0 :Blk=0:'Black
  PALETTE 1,0,0,1 :Blu=1:'Blue
  PALETTE 2,0,.75,0:Grn=2:'Green
  PALETTE 3,0,1,1 :Cyn=3:'Cyan
  PALETTE 4,1,0,0 :Red=4:'Red
  PALETTE 5,1,0,1 :Mag=5:'Magenta
  PALETTE 6,1,.8,0 :Yel=6:'Yellow
  PALETTE 7,1,1,1 :Wht=7:'White
END IF
COLOR ,Blk:CLS
AREA (376,8):AREA STEP (64,0):AREA STEP (-20,16)
AREA STEP (0,24):AREA STEP (-24,0):AREA STEP (0,-24)
COLOR Blu:AREAFILL
AREA (360,8):AREA STEP (32,0):AREA STEP (0,12)
AREA STEP (-16,0):AREA STEP (0,4):AREA STEP (8,0):AREA
STEP (0,8)
AREA STEP (-8,0):AREA STEP (0,4):AREA STEP (24,0):AREA
STEP (0,12)
AREA STEP (-40,0):COLOR Grn:AREAFILL
AREA (328,8):AREA STEP (24,0):AREA STEP (0,28)
AREA STEP (24,0):AREA STEP (0,12):AREA STEP (-48,0)
COLOR Red:AREAFILL
AREA (272,8):AREA STEP (64,0):AREA STEP (0,12)
AREA STEP (-20,0):AREA STEP (0,28):AREA STEP (-24,0):AREA
STEP (0,-28)
AREA STEP (-20,0):COLOR Cyn:AREAFILL
AREA (264,8):AREA STEP (16,0):AREA STEP (24,40)
AREA STEP (-16,0):AREA STEP (-8,-12):AREA STEP (-16,0):AREA
STEP (-8,12)
AREA STEP (-16,0):COLOR Mag:AREAFILL
AREA (200,8):AREA STEP (56,0):AREA STEP (0,16)
AREA STEP (-24,0):AREA STEP (0,-4):AREA STEP (-8,0):AREA
STEP (0,16)
AREA STEP (8,0):AREA STEP (0,-4):AREA STEP (24,0):AREA
STEP (0,16)
AREA STEP (-56,0):COLOR Yel:AREAFILL
COLOR Blu,Blk:LOCATE 24,7
PRINT"Bryan D. Catley 2221 Glasgow Road Alexandria
Virginia 22307-1819";
END SUB

SUB BldGadgets (Num,t1(),t2$( )) STATIC
FOR n=0 TO Num-1
  FOR m=0 TO 6
    READ t1(n,m)
  NEXT m
  READ t2$(n)
NEXT n
END SUB

SUB DrawGadgets (Ga%,Gb%,t1(),t2$( )) STATIC
FOR n=Ga% TO Gb%
  x1=t1(n,0):y1=t1(n,1):x2=x1+t1(n,2):y2=y1+t1(n,3)
  bg=t1(n,4):fg=t1(n,5):bo=t1(n,6)
  LINE(x1,y1)-(x2,y2),bg,bf:LINE(x1,y1)-(x2,y2),fg,B
  IF bo>-1 THEN
    LINE(x1+2,y1+2)-(x2-2,y2-2),fg,B
    LINE(x2+1,y1+1)-(x2+1,y2+1),bo
    LINE(x2+1,y2+1)-(x1+1,y2+1),bo

```

```

COLOR fg,bg:row%=INT(y1/8+2):col%=INT(x1/8+2)
LOCATE row%,col%:PRINT t2$(n)
END IF
NEXT n
END SUB

SUB GetGadget (Ga%,Gb%,t1(),t2$( ),type) STATIC
SHARED MouseX%,mouseY%,MouseInd
WHILE MOUSE(0)=0:WEND
r%=CSRLIN:c%=POS(0)
mx=MOUSE(1):my=MOUSE(2)
MouseX%=mx:mouseY%=my:MouseInd=0
FOR n=Ga% TO Gb%
  IF mx>t1(n,0) AND mx<t1(n,0)+t1(n,2) THEN
    IF my>t1(n,1) AND my<t1(n,1)+t1(n,3) THEN
      bg=t1(n,4):fg=t1(n,5):bo=t1(n,6)
      IF bo>-1 THEN
        x1=t1(n,0)+2:y1=t1(n,1)+2
        x2=x1+t1(n,2)-4:y2=y1+t1(n,3)-4
        LINE(x1,y1)-(x2,y2),fg,bf
        COLOR bg,fg:row%=INT(y1/8+2):col%=INT(x1/8+2)
        LOCATE row%,col%:PRINT t2$(n)
      ELSE
        IF bo=-1 THEN
          x1=t1(n,0):y1=t1(n,1):x2=x1+t1(n,2):y2=y1+t1(n,3)
          LINE(x1,y1)-(x2,y2),fg,bf:LINE(x1,y1)-(x2,y2),bg,B
        END IF
      END IF
      type=n-Ga%+1:n=Gb%:MouseInd=1
      IF bo<-1 THEN
        n%=type+Ga%-1:DrawGadgets n%,n%,t1(),t2$( )
      END IF
    END IF
  END IF
NEXT n
WHILE MOUSE(0)<>0:WEND
LOCATE r%,c%
END SUB

```

## Listing Two

```

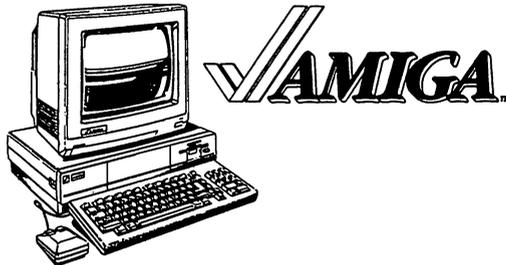
' The Household Inventory System
' -----
' This is Program #1 of 3: "HouseInv" -the main program
' Program #2 is the "HouseInvPrint" program
' Program #3 is the "HouseInvMaint" program
'
' Please do not modify the title screen in any way.
' January 1987
'
CLEAR ,35000&
numbx=52:NumFlds=8:maxdisp=18
MouseInd=0:FldNum=0:RecCnt=0:CurCol=0
TxtCol=0:NewCur=0:type=0
n=0:m=0:first=0:last=0:OldCnt=0:ErrSw=0:RecNum=0:RC=0:siz=0
x%=0:Lgth%=0:clm%=0:row%=0:offset%=0:offsetp%=0:MouseY%=0
Sldlghth%=0:SldStprt%=0:SldLft%=0:BarStprt%=0:GRx%=0:GRy%=0
A%=0:B%=0:SvA%=0:SvB%=0:BarLgth%=0:Spare%=0:HilSt%=0
Today$="" :strg$="" :type$="" :x$="" :Sep$=""
DIM bx (numbx-1,6),bxtxt$(numbx-1)
DIM FldTxt$(NumFlds-1,1),FldVal$(NumFlds-1)
DIM Fldlen (NumFlds-1),HldVal$(NumFlds-1)
Logo80 3:TxtCol=Blk
COLOR Red,Blk:LOCATE 9,34:PRINT"Please Standby."
Bldgadgets numbx,bx(),bxtxt$( )
' Main Title
DATA 120, 60, 80, 16,0,2, 7," Add"
DATA 280, 60, 80, 16,0,2, 7," Review"
DATA 440, 60, 80, 16,0,4, 7," Quit"
' Add
DATA 90,148, 56, 16,7,4, 0,"Cancel"
DATA 148,148, 52, 16,7,2, 0," OK"
DATA 202,148, 57, 16,7,1, 0,"Modify"

```

continued...

# Century Systems

has the best prices and service in the USA on the incredible AMIGA Computer.



Many Amiga's and Amiga expansion products available.

Prices too low to print. And service too good to ignore.

Century Systems  
8033 University Ave.  
Des Moines, IA 50311

SALES 1-800-223-8088

24 HR. SERVICE 1-515-223-8088

```
DATA 114,148, 73, 16,7,1, 0,"Add More"
DATA 189,148, 54, 16,7,4, 0," Quit"
' Add/Change
DATA 166, 7,100, 9,1,7,-1,""
DATA 166, 23,132, 9,1,7,-1,""
DATA 166, 39, 76, 9,1,7,-1,""
DATA 166, 55, 60, 9,1,7,-1,""
DATA 166, 71, 60, 9,1,7,-1,""
DATA 166, 87, 60, 9,1,7,-1,""
DATA 166,103,132, 9,1,7,-1,""
DATA 166,119,172, 9,1,7,-1,""
DATA 124,148, 55, 16,7,4, 0,"Cancel"
DATA 181,148, 54, 16,7,2, 0," OK"
' Review
DATA 8, 40,576, 8,3,3,-2,""
DATA 8, 48,576, 8,3,3,-2,""
DATA 8, 56,576, 8,3,3,-2,""
DATA 8, 64,576, 8,3,3,-2,""
DATA 8, 72,576, 8,3,3,-2,""
DATA 8, 80,576, 8,3,3,-2,""
DATA 8, 88,576, 8,3,3,-2,""
DATA 8, 96,576, 8,3,3,-2,""
DATA 8,104,576, 8,3,3,-2,""
DATA 8,112,576, 8,3,3,-2,""
DATA 8,120,576, 8,3,3,-2,""
DATA 8,128,576, 8,3,3,-2,""
DATA 8,136,576, 8,3,3,-2,""
DATA 8,144,576, 8,3,3,-2,""
DATA 8,152,576, 8,3,3,-2,""
DATA 8,160,576, 8,3,3,-2,""
DATA 8,168,576, 8,3,3,-2,""
DATA 8,176,576, 8,3,3,-2,""
DATA 508, 12, 88, 16,0,2, 7, "Finished"
DATA 600, 40, 24, 12,0,0,-2,""
DATA 600, 52, 24,120,1,7,-2,""
DATA 600,172, 24, 12,0,0,-2,""
```

```
' Count Initialization
DATA 28, 36, 24, 16,7,4, 0,"No"
DATA 116, 36, 24, 16,7,2, 0,"OK"
' Count Display
DATA 116, 76, 24, 16,7,2, 0,"OK"
' Update a Record
DATA 90,148, 56, 16,7,1, 0,"Modify"
DATA 148,148, 52, 16,7,2, 0," OK"
DATA 202,148, 57, 16,7,4, 0,"Delete"
' Record Delete Query
DATA 20, 36, 24, 16,7,4, 0,"No"
DATA 116, 36, 24, 16,7,2, 0,"OK"
' Restore Deleted Record Query
DATA 68,100, 48, 16,7,2, 0," Yes"
DATA 228,100, 48, 16,7,4, 0," No"
' Help Windows
DATA 36,172, 40, 16,7,4, 0,"More"
DATA 156,172, 40, 16,7,2, 0," OK"
MTA%=0:MTB%=2
AddA%=3:AddB%=4:AddC%=5:AddD%=6:AddE%=7
ChgA%=8:ChgB%=15:ChgC%=16:ChgD%=17
RevA%=18:RevB%=35:RevC%=36:RevD%=37:RevE%=38:RevF%=39
InA%=40:InB%=41
CnA%=42
UpdA%=43:UpdB%=45
DelA%=46:DelB%=47
DQA%=48:DQB%=49
HlpA%=50:HlpB%=51
FOR n=0 TO NumFlds-1
    READ FldTxt$(n,0),FldTxt$(n,1),FldLen$(n)
NEXT
DATA "Room:","(Required)",10
DATA "Item(s):","(Required)",15
DATA "Date of Purchase:","(mm/dd/yy)", 8
DATA "Original Cost:","(In Dollars)", 6
DATA "Current Worth:","(In Dollars)", 6
DATA "Replacement Cost:","(In Dollars)", 6
DATA "Serial Number:","",15
DATA "Comments:","",20
Sldlght%=INT (bx (RevE%,3)-2):SldStrt%=INT (bx (RevE%,1)+1)
SldLft%=INT (bx (RevE%,0)+4)
MENU 1,0,1,"Project:"
MENU 1,1,1,"Quit      "
MENU 2,0,1,"Help:"
MENU 2,1,1,"General    "
MENU 2,2,1,"Add        "
MENU 2,3,1,"Review     "
MENU 2,4,1,"Data File  "
MENU 2,5,1,"HouseInvPrint"
MENU 2,6,1,"HouseInvMaint"
MENU 3,0,1,"Data File:"
MENU 3,1,1,"Initialize  "
MENU 3,2,0,"Update Count "
MENU 3,3,0,"No. of Records"
MENU 4,0,0,""
ON MENU GOSUB MenuRtns:MENU ON
COLOR Yel,Blk
LOCATE 22,19
PRINT"First use of program?  Please use HELP menu!"
COLOR Cyn,Blk
LOCATE 9,34:PRINT SPACE$(15)
LOCATE 9,11:PRINT"Please enter today's date:"
LOCATE 9,49:PRINT" (in mm/dd/yy format)"
LINE (304,64)-STEP (74,7),Cyn,bf
WHILE Today$="" OR LEN(Today$)<6
    COLOR Blk,Cyn:LOCATE 9,39:GetIp Today$,"CHAR",8
WEND
LINE (80,56)-STEP (488,24),Blk,bf
ON ERROR GOTO InitError
GOSUB GetRecCnt
InitCnt:
ON ERROR GOTO 0
IF ErrSw=1 THEN ErrSw=0:RecCnt=0:GOSUB PutRecCnt
OldCnt=RecCnt
GOSUB OpenData
DrawGadgets MTA%,MTB%,bx(),bxtxt$()
LOCATE 22,19:PRINT SPACE$(44)
```

```
MENU 3,2,1:MENU 3,3,1
ON MOUSE GOSUB GetMouse
GOTO Main1
```

```
' Error Routine Used When Opening Count File During
Initilization
InitError:
WINDOW 2
IF ERR=53 THEN
  ErrSw=1:RESUME InitCont
ELSE
  ON ERROR GOTO 0
END IF
```

```
' Open Main Data File
' -----
```

```
OpenData:
OPEN "R",#1,"HouseInv.Data",103
FIELD #1,1 AS FFlg$,10 AS RFlD$(0),15 AS RFlD$(1),8 AS
RFlD$(2),6 AS RFlD$(3),6 AS RFlD$(4),6 AS RFlD$(5),15 AS
RFlD$(6),20 AS RFlD$(7),8 AS RAdd$,8 AS RChg$
RETURN
```

```
' Start of Main Processing
' -----
```

```
Main:
Logo80 3
DrawGadgets MTA%,MTB%,bx(),btxtxt$()
Main1:
COLOR Blu,Blk
LOCATE 12,39:PRINT"THE"
LOCATE 14,22:PRINT"H O U S E H O L D   I N V E N T O R Y"
LOCATE 16,38:PRINT"SYSTEM"
COLOR Mag,Blk
LOCATE 18,18
PRINT"D A T A   E N T R Y   &   M A I N T E N A N C E"
COLOR Yel,Blk:LOCATE 22,25:PRINT"Use HELP menus for
assistance."
Main2:
A%=MTA%:B%=MTB%:MOUSE ON
type=0:WHILE type=0:SLEEP:WEND:MOUSE OFF
ON type GOTO Addrtn,ReviewRtn,Quit
```

```
' Routine to ADD a New Record
' -----
```

```
Addrtn:
WINDOW 3,"Household Inventory: Add a New Item",
(40,0)-(400,184),0,1
AddClear:
COLOR Blk,Cyn:CLS
DrawGadgets AddA%,AddA%,bx(),btxtxt$()
MouseInd=0:type=0
A%=AddA%:B%=AddA%:MOUSE ON
COLOR Blk,Cyn:x%=ChgA%-1
FOR n=0 TO NumFlds-1:FldVal$(n)="" :NEXT
FOR n=0 TO NumFlds-1
  FldNum=n
  LOCATE 2+n*2,20-LEN(FldTtxt$(n,0))+1:PRINT FldTtxt$(n,0)
  LOCATE 3+n*2,20-LEN(FldTtxt$(n,1))+1:PRINT FldTtxt$(n,1)
  x%=x%+1:DrawGadgets x%,x%,bx(),btxtxt$()
  strg$="" :Lgth%=Fldlen$(n):COLOR Blk,Blu
  IF n>2 AND n<6 THEN type$="REAL" ELSE type$="CHAR"
  LOCATE 2+n*2,22:GetIp strg$,type$,Lgth%
  IF n<2 THEN
    IF MouseInd=0 THEN
      WHILE strg$="" :LOCATE 2+n*2,22
        GetIp strg$,type$,Lgth%:WEND
      END IF
    END IF
  IF MouseInd=0 THEN
    FldVal$(n)=strg$
    COLOR Blk,Cyn:LOCATE 3+n*2,1:PRINT SPACE$(20)
    IF n=1 THEN
      B%=AddC%:DrawGadgets AddB%,AddC%,bx(),btxtxt$()
      COLOR Blk,Cyn
    END IF
  ELSE

```

## SET UP INTUITION IN HALF THE TIME!

- Up to 75% reduction of time and code required to build Menus.
- Up to 50% reduction of time and code required to set up Screens, Windows, Requesters, any type of Gadget, Borders, Text and Images.
- Building of most Intuition graphic elements - screens, windows, gadgets, etc. - using only one or two lines of code.
- Automatic linking and updating of Gadgets added to any Window or Requester.

## IntuiSeeds

*IntuiSeeds* is an object code 'C' library designed to minimize the amount of time and energy you have to spend setting up an Intuition user Interface. Most of the structure building and complexity inherent to Intuition has been eliminated by using functions similar to those of Basic, Pascal, Fortran, etc.

Much attention was put on designing functions which would enhance, not limit, Intuition's potential while providing you with a fast and easy method of getting your application 'up and running' with properly established 'C' structures.

Also included is source code for a two window management program which demonstrates the use of *IntuiSeeds* functions and also acts as an aid to understanding Intuition's communication messages.

To order, send \$69.95 (US) check or money order (plus \$2.50 S/H) to:

*GreenThumb Software*  
P.O. Box 2949  
Binghamton, NY 13902

```
COLOR Blk,Blu:n=NumFlds-1
END IF
NEXT
WHILE MouseInd=0:SLEEP:WEND
ON type GOTO AddCancel,AddOK,AddChange
AddCancel:
FOR n=0 TO NumFlds-1:FldVal$(n)="" :NEXT
GOTO AddEnd
AddChange:
COLOR Blk,Cyn
IF FldNum+1=<NumFlds-1 THEN
  LOCATE 1+(FldNum+1)*2,1:PRINT SPACE$(20)
  FOR n=FldNum+1 TO NumFlds-1
    LOCATE 2+n*2,20-LEN(FldTtxt$(n,0))+1:PRINT FldTtxt$(n,0)
    x%=x%+1:DrawGadgets x%,x%,bx(),btxtxt$()
  NEXT
END IF
MOUSE OFF:GOSUB DoAChange:COLOR Blk,Blu:MOUSE ON
LINE (85,147)-(260,170),Cyn,bf
LOCATE 18,4:COLOR Cyn,Blu:PRINT SPACE$(12)+"Select
Option"+SPACE$(13)
DrawGadgets AddA%,AddC%,bx(),btxtxt$()
type=0:WHILE type=0:SLEEP:WEND:MOUSE OFF
ON type GOTO AddCancel,AddOK,AddChange
AddOK:
LSET FFlg$="0"
FOR n=0 TO NumFlds-1
  IF n<3 OR n>5 THEN
    LSET RFlD$(n)=FldVal$(n)
  ELSE
    RSET RFlD$(n)=FldVal$(n)
  END IF
NEXT
LSET RAdd$=Today$:LSET RChg$=""
RecCnt=RecCnt+1
```

*continued...*

```

PUT #1,RecCnt
AddEnd:
LOCATE 18,4:COLOR Cyn,Blu
PRINT" Your Request has been Complied With. "
LINE (85,147)-(260,170),Cyn,bf
DrawGadgets AddD%,AddE%,bx(),bxtxt$()
A%=AddD%:B%=AddE%:MOUSE ON
type=0:WHILE type=0:SLEEP:WEND:MOUSE OFF
ON type GOTO AddClear,AddExit
AddExit:
IF RecCnt<>OldCnt THEN GOSUB PutRecCnt
WINDOW CLOSE 3:GOTO Main2

' Routine to REVIEW All Current Records
' -----
ReviewRtn:
MENU 3,0,0
IF RecCnt=0 THEN RRNoRecs
n=Sldlqth%
BarLqth%=CINT((n/100)*((maxdisp/RecCnt)*100))
IF BarLqth%>Sldlqth% THEN BarLqth%=Sldlqth%
IF BarLqth%<5 THEN BarLqth%=5
Spare%=Sldlqth%-BarLqth%
COLOR Mag,Blk:CLS
LOCATE 1,26:PRINT"REVIEW OF HOUSEHOLD INVENTORY"
COLOR Yel,Blk:LOCATE 2,26
PRINT"Click on record to select it."
LOCATE 3,25
PRINT USING "Number of items on file: #####";RecCnt
DrawGadgets RevC%,RevC%,bx(),bxtxt$()
Hilst%=1:IF RecCnt>maxdisp THEN Hilst%=RecCnt-maxdisp
COLOR Blu,Blk
LOCATE 5,2:PRINT"Room----- Item----- Pur-Date"
LOCATE 5,38:PRINT"O-Cost CWorth R-Cost Serial-Number--"
LINE (0,40)-(592,184),Cyn,bf
DrawGadgets RevD%,RevF%,bx(),bxtxt$()
COLOR Wht
AREA (612,40):AREA STEP(-12,12):AREA STEP(24,0):AREAFILL
AREA (612,184):AREA STEP(-12,-12):AREA STEP(24,0):AREAFILL
DIM DispFile$(RecCnt-1)
COLOR Red,Cyn:LOCATE 7,31:PRINT"Loading File"
FOR n=0 TO RecCnt-1
  GET #1,n+1
  IF FFlg$="0" THEN Sep$=" " ELSE Sep$="*"
  DispFile$(n)=Sep$
  FOR m=0 TO NumFlds-2
    DispFile$(n)=DispFile$(n)+RFld$(m)+Sep$
  NEXT
NEXT
LOCATE 7,31:PRINT SPACE$(12)
first=0
IF RecCnt<maxdisp+1 THEN
  last=RecCnt-1
  COLOR Yel,Blk:LOCATE 4,31:PRINT"All Items Displayed"
ELSE
  last=maxdisp-1
END IF
DispScreen:
GOSUB DispRecNo
COLOR Blk,Cyn:row%=6-1
LINE(0,40)-(592,184),Cyn,bf
FOR n=first TO last
  row%=row%+1:LOCATE row%,1:PRINT DispFile$(n)
NEXT
GOSUB DrawBar
RevGdgt:
A%=RevA%:B%=RevF%:MOUSE ON
type=0:WHILE type=0:SLEEP:WEND:MOUSE OFF
IF type<maxdisp+1 THEN GetRecord
type=type-maxdisp
ON type GOTO RevExit,ScrollUp,ChgDisp,ScrollDwn
ScrollUp:
IF first+1>1 THEN
  first=first-1:last=last-1
  GOTO DispScreen
ELSE

```

```

GOTO RevGdgt
END IF
ScrollDwn:
IF RecCnt>last+1 THEN
  first=first+1:last=last+1
  GOTO DispScreen
ELSE
  GOTO RevGdgt
END IF
ChgDisp:
IF maxdisp=>RecCnt THEN RevGdgt
offset%=MouseY%-(SldStrt%-1)
offsetp%=CINT(offset%/Sldlqth%)*100
first=(RecCnt/100)*offsetp%
last=first+(maxdisp-1)
IF last>RecCnt THEN last=RecCnt-1
first=(last-maxdisp)+1
GOTO DispScreen
GetRecord:
IF type>(last-first)+1 THEN RevGdgt
RecNum=first+type
WINDOW 8,"Household Inventory: Update Item", (40,0)-
(400,184),0,1
COLOR Blk,Cyn:CLS
GET #1,RecNum
IF FFlg$="1" THEN
  LOCATE 6,5:PRINT"The selected record has been deleted."
  LOCATE 8,10:PRINT"Do you wish to restore it?"
  DrawGadgets DQA%,DQB%,bx(),bxtxt$()
  A%=DQA%:B%=DQB%:MOUSE ON
  type=0:WHILE type=0:SLEEP:WEND:MOUSE OFF
  IF type=1 THEN
    LSET FFlg$="0":LSET RChg$=Today$
    PUT #1,RecNum
    COLOR Blk,Cyn:CLS
    GOTO ShoRec
  ELSE
    GOTO GRExit
  END IF
END IF
ShoRec:
x%=ChgA%-1
FOR n=0 TO NumFlds-1
  FldVal$(n)=RFld$(n)
  COLOR Blk,Cyn
  LOCATE 2+n*2,20-LEN(FldTxt$(n,0))+1:PRINT FldTxt$(n,0)
  x%=x%+1:DrawGadgets x%,x%,bx(),bxtxt$()
  IF n>2 AND n<6 THEN
    offset=0
    FOR m=1 TO LEN(FldVal$(n))
      IF MID$(FldVal$(n),m,1)<>" " THEN
        offset=m:m=LEN(FldVal$(n))
      NEXT
      IF offset=0 THEN FldVal$(n)="" ELSE
        FldVal$(n)=MID$(FldVal$(n),offset)
      ELSE
        siz=0
        FOR m=LEN(FldVal$(n)) TO 1 STEP -1
          IF MID$(FldVal$(n),m,1)<>" " THEN siz=m:m=1
        NEXT
        IF siz=0 THEN FldVal$(n)="" ELSE
          FldVal$(n)=MID$(FldVal$(n),1,siz)
        END IF
        COLOR Blk,Blu:LOCATE 2+n*2,22:PRINT FldVal$(n)
      NEXT
LOCATE 18,4:COLOR Cyn,Blu
PRINT" Please select desired function. "
DrawGadgets UpdA%,UpdB%,bx(),bxtxt$()
A%=UpdA%:B%=UpdB%:MOUSE ON:type=0:WHILE type=0:SLEEP:WEND
ON type GOTO GRMod,GRExit,GRDel
GRMod:
MOUSE OFF:GOSUB DoAChange:MOUSE ON
IF RC<>0 THEN GRExit
LSET FFlg$="0"
FOR n=0 TO NumFlds-1

```

```

IF n<3 OR n>5 THEN
  LSET Rfld$(n)=FldVal$(n)
ELSE
  RSET Rfld$(n)=FldVal$(n)
END IF
NEXT
LSET RChg$=Today$
PUT #1,RecNum
GOTO GRExit
GRDel:
LINE (85,148)-(300,172),Cyn,bf
WINDOW 9,,(440,40)-(608,98),0,1
COLOR Blu,Yel:CLS
LOCATE 2,2:PRINT"OK to delete record"
LOCATE 3,2:PRINT"shown at left?"
DrawGadgets DelA%,DelB%,bx(),bxtxt$( )
A%=DelA%:B%=DelB%:type=0:WHILE type=0:SLEEP:WEND
ON type GOTO GRDelEnd,GRDelYes
GRDelYes:
LSET FFlg$="1"
PUT #1,RecNum
GRDelEnd:
WINDOW CLOSE 9
GOTO GRExit
GRExit:
MOUSE OFF:WINDOW CLOSE 8
GOTO RevGdgt

' No Records to Review
RRNoRecs:
MENU OFF
WINDOW 4,,(440,40)-(608,80),0,1
COLOR Blu,Yel:CLS
LOCATE 2,3:PRINT"No Data to Review"
LOCATE 4,3:PRINT"Press left Button"
LOCATE 5,3:PRINT"to Continue."
WHILE MOUSE(0)=0:WEND
WINDOW CLOSE 4
MENU ON
GOTO Main2

DispRecNo:
IF RecCnt>maxdisp THEN
  COLOR Yel,Blk:LOCATE 4,22
  PRINT USING "Record number at top of display:####";first+1
END IF
RETURN

DrawBar:
BarStrt%=SldStrt%+(Spare%-(RecCnt-
(last+1))*(Spare%/Hilst%))
DrawGadgets RevE%,RevE%,bx(),bxtxt$( )
LINE(SldLft%,BarStrt%)-STEP(17,BarLgth%),Wht,bf
RETURN

' Leave Review Routine
RevExit:
ERASE DispFile$
MENU 3,0,1
GOTO Main

' Change an Existing Record (During Either ADD or REVIEW)
' -----
DoAChange:
SvA%=A%:SvB%=B%
FOR n=0 TO NumFlds-1:HldVal$(n)=FldVal$(n):NEXT
LINE(85,147)-(260,170),Cyn,bf
DrawGadgets ChgC%,ChgD%,bx(),bxtxt$( )
DACGdgt:
COLOR Cyn,Blu:LOCATE 18,4
PRINT" Click in box it is desired to change "
A%=ChgA%:B%=ChgD%:MOUSE ON
type=0:WHILE type=0:SLEEP:WEND:MOUSE OFF
IF type>8 THEN
  type=type-8
  ON type GOTO ChngCancel,ChngOK

```

## A-TALK™

### Communication and Terminal Program

- KERMIT - XMODEM - XMODEM/CRC - ASCII
- DIAL-A-TALK - Script language. 20 function keys.
- FULL VT100/VT52/H19/ANSI/TTY emulations.
- Concurrent printing and capture. Voice option. CB mode.

## A-TALK PLUS

### Tektronix 4010/4014 Graphics Emulation

- ALPHA/GRAPH/GIN standard modes, plus enhanced graphics POINT PLOT and INCREMENTAL PLOT.
- All vector line formats. Screen size up to 700 by 440.
- Four character sizes. Printer support. Store screens in IFF or Aegis Draw format. All A-TALK features supported.

A-TALK lists for \$49.95. A-TALK PLUS lists for \$99.95.  
\$2.00 shipping; CA residents add 6.5% sales tax.

Felsina Software  
3175 South Hoover Street, #275  
Los Angeles, CA 90007  
(213) 747-8498

```

END IF
COLOR Cyn,Blu:LOCATE 18,4
PRINT" Press RETURN after making change "
n=type-1:Lgth%=Fldlen$(n):strg%=FldVal$(n):COLOR Blk,Wht
IF n>2 AND n<6 THEN type$="REAL" ELSE type$="CHAR"
LOCATE 2+n*2,22:GetIp strg%,type%,Lgth%
IF n<2 THEN
  WHILE strg%="":LOCATE 2+n*2,22:GetIp
  strg%,type%,Lgth%:WEND
END IF
FldVal$(n)=strg%
GOTO DACGdgt
ChngCancel:
COLOR ,Cyn:LOCATE 18,4:PRINT SPACES(38)
COLOR Blk,Blu:x%=ChgA%-1
FOR n=0 TO NumFlds-1
  FldVal$(n)=HldVal$(n)
  x%=x%+1:DrawGadgets x%,x%,bx(),bxtxt$( )
  LOCATE 2+n*2,22:PRINT FldVal$(n)
NEXT
FldNum=n:RC=4
GOTO DACExit
ChngOK:
RC=0
DACExit:
A%=SvA%:B%=SvB%:COLOR Blu,Cyn
RETURN

' INPUT and WRITE Record Count
' -----
GetRecCnt:
OPEN "HouseInv.Count" FOR INPUT AS #2
INPUT #2,RecCnt

```

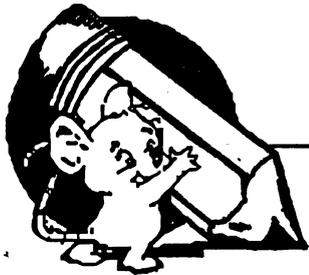
*continued...*



# Go With The *IMPULSE*

## ProClip

Artwork for  
DeskTop Publishing  
Sales Reports  
Greeting Cards  
Computer Video  
Animation



Now with PROCLIP you can produce documents that deliver your message!

## 5 PROCLIP DISKS

Over 200 images  
per disk!

Space and Technology  
Leisure Time  
Sports  
Transportation  
Cartoons



Any one of these disks  
**Only \$29.95 Each**

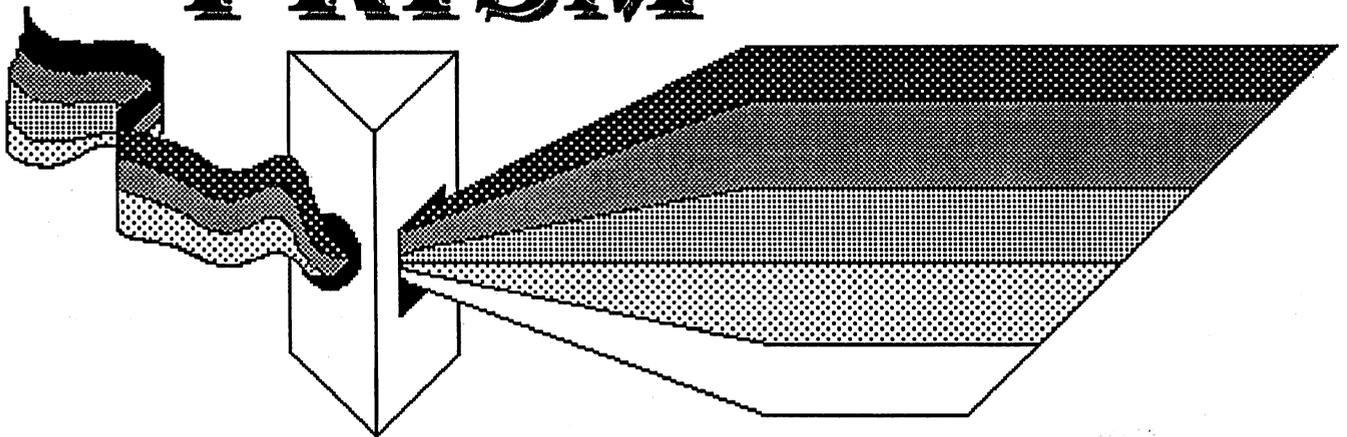
TO ORDER CALL THE *IMPULSE* HOTLINE

**1-800-328-0184**

IMPULSE, INC. 6860 Shingle Creek Parkway #110 Mpls, MN 55430

# PRISM

By Impulse



You have a computer that operates in a limited mode with most of the paint programs that are presently available. With *PRISM* you can operate in true *HAM* mode of 4096 colors. No longer are you limited to 32 colors to paint with, now you have the full Amiga™ color palette. Prism is designed to use the entire keyboard as well as the mouse to make painting and editing your artwork a real pleasure. Filp images, use a real lasso for grabbing objects and you can use any file that is IFF standard. Get the full power of your computer with *PRISM* see what you have been missing !!!

# PRISM

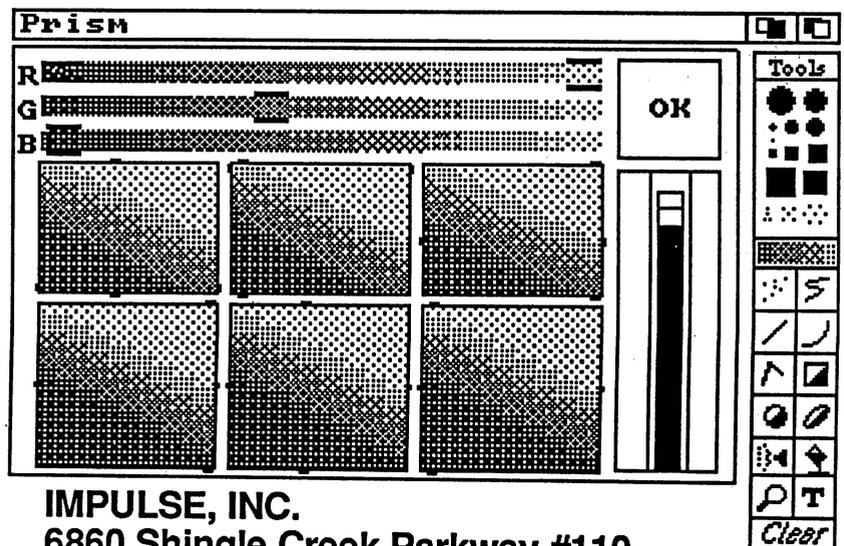
only \$69.95

Here is the 4096 palette screen. To pick your colors all you have to do is move your mouse and click that's all there is to using the power of the full 4096 palette.

Don't delay, because now there are no limits on your creative talents. Go with your *IMPULSE* and get your copy of *PRISM™* today.

TO ORDER CALL THE IMPULSE  
HOTLINE

1-800-328-0184



IMPULSE, INC.  
6860 Shingle Creek Parkway #110  
Minneapolis, MN 55430

```

CLOSE #2
RETURN
PutRecCnt:
WINDOW 5,,(440,40)-(608,80),0,1
COLOR Blu,Yel:CLS
LOCATE 2,4:PRINT"Updating Count"
LOCATE 3,9:PRINT"File"
OldCnt=RecCnt
OPEN "HouseInv.Count" FOR OUTPUT AS #2
WRITE #2,RecCnt
CLOSE #2
WINDOW CLOSE 5
RETURN

' Time to QUIT and Return to Basic
' -----
Quit:
IF RecCnt<>OldCnt THEN GOSUB PutRecCnt
CLOSE #1:CLOSE #2
MENU OFF:MENU RESET
CLS:WINDOW CLOSE 2:SCREEN CLOSE 1:END

' Mouse Interrupt Routine
' -----
GetMouse:
GetGadget A%,B%,bx(),bxtxt$(),type
RETURN

' Menu Request Handling Routines
' -----
MenuRtns:
MOUSE OFF:SvA%=A%:SvB%=B%
m=MENU(0):i=MENU(1)
MENU OFF
ON m GOTO ProjRtn,HelpRtn,DataRtn
ProjRtn:
ON i GOTO Mquit
Mquit:
GOTO Quit
HelpRtn:
GOSUB DoHelp:GOTO MRExit
DataRtn:
ON i GOTO MInit,MUpdtCnt,MNum
MInit:
WINDOW 6,,(440,40)-(608,96),0,1
COLOR Blu,Yel:CLS
LOCATE 2,2:PRINT"This option will"
LOCATE 3,2:PRINT"delete all records"
LOCATE 4,2:PRINT"on file."
DrawGadgets InA%,InB%,bx(),bxtxt$()
A%=InA%:B%=InB%:GOSUB MenuMouse
IF RecCnt>0 AND type=2 THEN
  CLOSE #1
  KILL"HouseInv.Data.info"
  KILL"HouseInv.Data"
  GOSUB OpenData
END IF
WINDOW CLOSE 6
ON type GOTO MInExit,MInOK
MInOK:
RecCnt=0:GOSUB PutRecCnt
MInExit:
GOTO MRExit
MUpdtCnt:
GOSUB PutRecCnt
GOTO MRExit
MNum:
WINDOW 6,,(440,40)-(608,136),0,1
COLOR Blu,Yel:CLS
LOCATE 2,4:PRINT"Record Counts"
LOCATE 4,2:PRINT USING"#####";OldCnt
LOCATE 5,2:PRINT USING"#####";RecCnt
IF OldCnt<>RecCnt THEN
  LOCATE 7,2:PRINT"Use 'Update Count'"
  LOCATE 8,2:PRINT"menu to commit all"

```

```

LOCATE 9,2:PRINT"all records."
END IF
DrawGadgets CnA%,CnA%,bx(),bxtxt$()
A%=CnA%:B%=CnA%:GOSUB MenuMouse
WINDOW CLOSE 6
GOTO MRExit
MRExit:
A%=SvA%:B%=SvB%:type=0:MOUSE ON:MENU ON
RETURN

' Help Routines (requested via Menu Routines)
DoHelp:
WINDOW 9,,(408,0)-(631,186),0,1
COLOR Blu,Yel:CLS:LOCATE 2,1
ON i GOTO HlpGen,HlpAdd,HlpRev,HlpData,HlpIPrt,HlpIMaint
HlpGen:
PRINT" 'The Household Inventory"
PRINT" System' is made up of"
PRINT" three programs: HouseInv,"
PRINT" HouseInvPrint, and"
PRINT" HouseInvMaint.":PRINT" "
PRINT" These programs allow users"
PRINT" to create, maintain, and"
PRINT" report on a file of their"
PRINT" Household Inventory. The"
PRINT" data are stored in a"
PRINT" direct access file on the"
PRINT" same disk as that used by"
PRINT" the program.":PRINT" "
PRINT" Other Help menu items pro-"
PRINT" vide further information."
DrawGadgets HlpA%,HlpB%,bx(),bxtxt$()
A%=HlpA%:B%=HlpB%:GOSUB MenuMouse
ON type GOTO HlpGen1,HlpExit
HlpGen1:
COLOR Blu,Yel:CLS:LOCATE 2,1
PRINT" "
PRINT" These programs, the data"
PRINT" files, and a copy of Basic"
PRINT" should be kept on a sep-"
PRINT" arate disk, a copy of"
PRINT" which is stored in a"
PRINT" Safety Deposit Box."
DrawGadgets HlpB%,HlpB%,bx(),bxtxt$()
A%=HlpB%:B%=HlpB%:GOSUB MenuMouse
GOTO HlpExit
HlpAdd:
PRINT" Only Room & Item are Man-"
PRINT" datory. Either enter the"
PRINT" requested data, (RETURN"
PRINT" must be pressed for it to"
PRINT" register), or click in"
PRINT" the desired gadget.":PRINT" "
PRINT" Cancel -clear the item"
PRINT" OK -add item to file"
PRINT" Modify -modify data just"
PRINT" entered":PRINT" "
PRINT" If Cancel or OK selected:"
PRINT" Add More -add another item"
PRINT" Quit -end add; update"
PRINT" count file"
DrawGadgets HlpA%,HlpB%,bx(),bxtxt$()
A%=HlpA%:B%=HlpB%:GOSUB MenuMouse
ON type GOTO HlpAdd1,HlpExit
HlpAdd1:
COLOR Blu,Yel:CLS:LOCATE 2,1
PRINT" When 'Modify' is selected,"
PRINT" click on item it is"
PRINT" desired to change, and"
PRINT" enter or update the value."
PRINT" When updating, ESC will"
PRINT" clear the value to null."
PRINT" 'RETURN' must be pressed"
PRINT" for change to register."

```

```

PRINT" Click on desired gadget"
PRINT" when appropriate:":PRINT" "
PRINT" Cancel -Changes are"
PRINT" removed"
PRINT" Add -Changes remain":PRINT" "
PRINT" In both cases you are"
PRINT" returned to the original"
PRINT" window."
DrawGadgets HlpB%,HlpB%,bx(),bxtxt$( )
A%=HlpB%:B%=HlpB%:GOSUB MenuMouse
GOTO HlpExit:
HlpRev:
PRINT" 'Review' allows you to"
PRINT" browse through the data"
PRINT" file. Click in the slider"
PRINT" to move several records at"
PRINT" a time, or the arrow heads"
PRINT" to scroll one record."
PRINT" Clicking on a record will"
PRINT" give you an Update Window"
PRINT" with gadgets:":PRINT" "
PRINT" Modify -modify record"
PRINT" OK -accept changes and"
PRINT" return to Review"
PRINT" Delete -delete record":PRINT" "
PRINT" See Add Help for more"
PRINT" information on Modify."
DrawGadgets HlpA%,HlpB%,bx(),bxtxt$( )
A%=HlpA%:B%=HlpB%:GOSUB MenuMouse
ON type GOTO HlpRev1,HlpExit
HlpRev1:
COLOR Blu,Yel:CLS:LOCATE 2,1
PRINT" Delete is a logical delete"
PRINT" I.E. it is not actually"
PRINT" deleted from the file"
PRINT" until the file is reorg-"
PRINT" anized, when it is gone"
PRINT" forever! Instead, it will"
PRINT" show up with fields sep-"
PRINT" arated by asterisks. It"
PRINT" may be restored by"
PRINT" clicking on it, AND then"
PRINT" the subsequent 'Yes'"
PRINT" gadget.":PRINT" "
PRINT" 'Review' shows the file as"
PRINT" it was at the time of the"
PRINT" request. Changes and"
PRINT" deletes will not show"
PRINT" until the next 'Review'."
DrawGadgets HlpB%,HlpB%,bx(),bxtxt$( )
A%=HlpB%:B%=HlpB%:GOSUB MenuMouse
GOTO HlpExit
HlpData:
PRINT" The 'Data File' menu "
PRINT" provides:":PRINT" "
PRINT" Initialize -creates a zero"
PRINT" count file. Use very"
PRINT" first time or when re-"
PRINT" starting from scratch."
PRINT" Update Count -count file"
PRINT" is updated to reflect"
PRINT" newly added records."
PRINT" No. of Records -lists"
PRINT" number of records in"
PRINT" system and known by"
PRINT" count file."
DrawGadgets HlpB%,HlpB%,bx(),bxtxt$( )
A%=HlpB%:B%=HlpB%:GOSUB MenuMouse
GOTO HlpExit
HlpIPrt:
HlpIMaint:
PRINT" In the interests of memory"
PRINT" conservation and increased"
PRINT" flexibility, the printer"
PRINT" and file maintenance"
PRINT" routines are contained in"
PRINT" separate programs named"

```

## AMIGA HARD DISK BACKUP HARDHAT

Full/Incremental/Directory/Single File backup to microdisks. Option list allows skipping of files by name with wildcards. Catalog file provides display of backed up files by name with size, location and datestamp. Double data compression reduced disk space. Printer interface. Uses CLI or Workbench. Multitasking provides background operation. — \$69.95

## AMIGA DISK FILE ORGANIZER ADFO

Having trouble finding that file somewhere in your stack of floppys? Can't find all the copies of a particular file? ADFO maintains a database of directories and filenames from your collection of disks. Fast response inquiries return location and last update information. Printer interface. Uses CLI or Workbench. 512K ram and 2 drives recommended — \$59.95.

## AMIGA SPELLING CHECKER SPEL-IT

Uses 40,000 word primary dictionary and optional second dictionary. Add/Delete words to both dictionaries. Includes plurals. Text wordcount totals. Uses CLI or Workbench, Mouse or keyboard. — \$49.95

Include \$3.50 S&H Mastercard/Visa Accepted  
Calif. Residents Add 6½% Sales Tax

*Westcom Industries*

3386 Floyd  
Los Angeles, CA 90068 (213) 851-4868  
Order phone 1 800 621-0849 Ext. 494

```

PRINT" 'HouseInvPrint' and"
PRINT" 'HouseInvMaint'.":PRINT" "
PRINT" Please use the Help menus"
PRINT" in those programs for"
PRINT" further information."
DrawGadgets HlpB%,HlpB%,bx(),bxtxt$( )
A%=HlpB%:B%=HlpB%:GOSUB MenuMouse
GOTO HlpExit
HlpExit:
WINDOW CLOSE 9:RETURN

' Mouse Routine for Menu Routines
MenuMouse:
type=0
WHILE type=0:GetGadget A%,B%,bx(),bxtxt$( ),type:WEND
RETURN

' Various SUBPROGRAMS
' -----
SUB GetIp (Text$,DataType$,MaxLen%) STATIC
SHARED TxtCol,NewCur,MouseInd
start=POS(0):cur=0:COLOR TxtCol
xpix=(start-1)*8:ypix=(CSRLIN-1)*8
IF FirstTime=0 THEN FirstTime=1:NewCur=1:DIM IPcursor%(46)
IF NewCur=1 THEN
NewCur=0
CurCol=TxtCol-1:IF CurCol<0 THEN CurCol=TxtCol+1
LINE(xpix,ypix)-STEP(7,7),CurCol,bf
GET(xpix,ypix)-STEP(7,7),IPcursor%
END IF
ShoText:
GOSUB DisplayText
NxtChar:
x$="":MouseInd=0:LeftPart$="":RightPart$=""

```

*continued...*

```

WHILE x$="" AND MouseInd=0:x$=INKEY$:WEND
IF MouseInd<>0 THEN GetDone ' Mouse was clicked
IF x$=CHR$(30) THEN CurRight ' Right-cursor
IF x$=CHR$(31) THEN CurLeft ' Left-cursor
IF x$=CHR$(8) THEN DelLeft ' Back-space key
IF x$=CHR$(127) THEN DelRight ' Delete key
IF x$=CHR$(27) THEN ClrText ' Escape key
IF x$=CHR$(13) THEN GetDone ' Return key
IF DataType$="CHAR" THEN
  IF x$<CHR$(32) OR x$>CHR$(127) THEN
    BEEP:GOTO NxtChar
  END IF
ELSEIF DataType$="REAL" THEN
  IF x$<CHR$(48) OR x$>CHR$(57) AND (x$<>".") THEN
    BEEP:GOTO NxtChar
  END IF
ELSEIF DataType$="INT" THEN
  IF x$<CHR$(48) OR x$>CHR$(57) THEN
    BEEP:GOTO NxtChar
  END IF
END IF
InsertChar:
IF LEN(Text$)=MaxLen% THEN BEEP:GOTO NxtChar
IF cur>0 THEN LeftPart$=MID$(Text$,1,cur)
IF LEN(Text$)>0 THEN
  RightPart$=MID$(Text$,cur+1,LEN(Text$)-
  LEN(LeftPart$))
Text$=LeftPart$+x$+RightPart$:cur=cur+1
GOTO ShoText
CurRight:
IF cur=LEN(Text$) THEN NxtChar
cur=cur+1:GOTO ShoText
CurLeft:
IF cur=0 THEN NxtChar
cur=cur-1:GOTO ShoText
DelLeft:
IF LEN(Text$)=0 OR cur=0 THEN BEEP:GOTO NxtChar
IF cur>1 THEN LeftPart$=MID$(Text$,1,cur-1)
IF LEN(Text$)>cur THEN
  RightPart$=MID$(Text$,cur+1,LEN(Text$)-cur)
Text$=LeftPart$+RightPart$
cur=cur-1:GOTO ShoText
DelRight:
IF LEN(Text$)=0 OR cur=LEN(Text$) THEN BEEP:GOTO NxtChar
IF cur>0 THEN LeftPart$=MID$(Text$,1,cur)
IF cur+1<LEN(Text$) THEN
  RightPart$=MID$(Text$,cur+2,LEN(Text$)-cur+1)
Text$=LeftPart$+RightPart$
GOTO ShoText
ClrText:
PRINT SPACES(MaxLen%+1);:LOCATE ,start
cur=0:Text$="":GOTO ShoText
DisplayText:
PRINT Text$+SPACES(MaxLen%+1-LEN(Text$));:LOCATE ,start
xpix=(start+cur-1)*8:PUT(xpix,ypix),IPcursor%
RETURN
GetDone:
PUT(xpix,ypix),IPcursor%
END SUB

```

### Listing Three

```

' Listing #3 - Merge in Listing #1 before using.
'
' The Household Inventory Report Program
' -----
' This is Program #2 of 3: "HouseInvPrint" -reports
' Program #1 is the "HouseInv" (main) program
' Program #3 is the "HouseInvMaint" program
'
' Please do not modify the title screen in any way.
' January 1987

```

```

'
numbx=2:RecCnt=0:n=0:m=0:i=0:ErrSW=0
type=0:PageCnt=0:LineCnt=0
LoadSW=0:Limit=0:Ptr1=0:Ptr2=0:offset=0:lgth=0
A%=0:B%=0
NewSeq$="" : OldSeq$="" : Sep$="" : x$=""
DIM bx (numbx-1,6),bxtxt$(numbx-1)
Logo80 3
BldGadgets numbx,bx(),bxtxt$()
' Help Gadgets
DATA 36,172,40,16,7,4,0,"More"
DATA 156,172,40,16,7,2,0,"Ok"
HlpA%=0:HlpB%=1
COLOR Blu,Blk
LOCATE 9,39:PRINT"THE"
LOCATE 11,22:PRINT"H O U S E H O L D I N V E N T O R Y"
LOCATE 13,38:PRINT"SYSTEM"
COLOR Mag,Blk
LOCATE 16,28:PRINT"P R I N T R E P O R T S"
MENU 1,0,1,"Project:"
MENU 1,1,1,"Quit "
MENU 2,0,1,"Help:"
MENU 2,1,1,"General "
MENU 2,2,1,"Order Entry"
MENU 2,3,1,"By Room "
MENU 2,4,1,"By Item "
MENU 3,0,1,"Report:"
MENU 3,1,1,"Order Entry"
MENU 3,2,1,"By Room "
MENU 3,3,1,"By Item "
MENU 4,0,0,""
ON MENU GOSUB GetMenu
ON MOUSE GOSUB GetMouse
COLOR Yel,Blk:LOCATE 21,23
PRINT"Use Menus to select program function"
MENU ON
WaitHere:
MENU 2,0,1:MENU 3,0,1
m=0:i=0:WHILE m=0:SLEEP:WEND
MENU 2,0,0:MENU 3,0,0
ON m GOTO Quit,Help,Report

' Menu Event Routine
' -----
GetMenu:
m=MENU(0):i=MENU(1)
RETURN

' Mouse Event Routine
' -----
GetMouse:
GetGadget A%,B%,bx(),bxtxt$(),type
RETURN

' Wait for Mouse Click
' -----
WaitMouse:
MOUSE ON
type=0:WHILE type=0:SLEEP:WEND
MOUSE OFF
RETURN

' Open and Load Data File
' -----
OpenAndLoad:
ON ERROR GOTO CountError
OPEN"HouseInv.Count" FOR INPUT AS #2
ON ERROR GOTO 0
INPUT #2,RecCnt
CLOSE #2
IF RecCnt=0 THEN OALXit
WINDOW 3,,(440,40)-(608,96),0,1
COLOR Blu,Yel:CLS
LOCATE 2,2:PRINT>Loading File..."
OPEN "R",#1,"HouseInv.Data",103

```

```
FIELD #1,1 AS FFlg$,10 AS d$(0),15 AS d$(1),8 AS d$(2),6 AS
d$(3),6 AS d$(4),6 AS d$(5),15 AS d$(6),20 AS d$(7),8 AS
d$(8),8 AS d$(9)
```

```
DIM Records$(RecCnt)
```

```
FOR n=1 TO RecCnt
```

```
GET #1,n
```

```
IF FFlg$="0" THEN Sep$=" " ELSE Sep$="*"
```

```
Records$(n)=Sep$
```

```
FOR m=0 TO 9
```

```
Records$(n)=Records$(n)+d$(m)+Sep$
```

```
NEXT
```

```
NEXT
```

```
CLOSE #1:LoadSW=1:ErrSW=0
```

```
WINDOW CLOSE 3
```

```
GOTO OALXit
```

```
CountError:
```

```
WINDOW 2
```

```
IF ERR=53 THEN
```

```
LoadSW=0:ErrSW=1:RESUME OALXit
```

```
ELSE
```

```
ON ERROR GOTO 0
```

```
END IF
```

```
OALXit:
```

```
RETURN
```

```
' Time to Quit and Return to Basic
```

```
Quit:
```

```
MENU OFF:MENU RESET
```

```
WINDOW CLOSE 2:SCREEN CLOSE 1
```

```
END
```

```
' Help Routines
```

```
Help:
```

```
GOSUB DoHelp
```

```
GOTO WaitHere
```

```
' Generate Requested Report
```

```
Report;
```

```
IF i=1 THEN NewSeq$="ASIS"
```

```
IF i=2 THEN NewSeq$="ROOM"
```

```
IF i=3 THEN NewSeq$="ITEM"
```

```
IF LoadSW=0 THEN GOSUB OpenAndLoad
```

```
IF ErrSW=1 OR RecCnt=0 THEN
```

```
WINDOW 3,,(440,40)-(608,92),0,1
```

```
COLOR Blu,Yel:CLS
```

```
LOCATE 2,3:PRINT"File is empty or"
```

```
LOCATE 3,3:PRINT"does not exist."
```

```
LOCATE 5,3:PRINT"Press left button"
```

```
LOCATE 6,3:PRINT"to continue."
```

```
WHILE MOUSE(0)=0:WEND
```

```
WINDOW CLOSE 3
```

```
GOTO RptXit
```

```
END IF
```

```
IF NewSeq$=OldSeq$ OR NewSeq$="ASIS" THEN DoPrint
```

```
' Looks Like We Have to Sort the File
```

```
WINDOW 3,,(440,40)-(608,92),0,1
```

```
COLOR Blu,Yel:CLS
```

```
LOCATE 2,3:PRINT"Sorting File..."
```

```
OldSeq$=NewSeq$
```

```
Limit=1:WHILE Limit<=RecCnt:Limit=2*Limit:WEND
```

```
HalfIt:
```

```
Limit=INT(Limit/2)
```

```
IF Limit=0 THEN SortDone
```

```
FOR n=1 TO RecCnt-Limit
```

```
Ptr1=n
```

```
WHILE Ptr1>0
```

```
Ptr2=Ptr1-Limit
```

```
IF NewSeq$="ROOM" THEN offset=2:lgth=10
```

```
IF NewSeq$="ITEM" THEN offset=13:lgth=15
```

# AMIGA MEMORY 2 FULL MEGABYTES \$450\*

- Small expansion box fits right side of Amiga
- No wait states
- Auto Configuring
- 1 year guaranteed parts & labor
- Fully stuffed
- Adds 2 full megabytes to existing Amiga's memory

## Alphanetics

APPARATUS ©

800-321-5346 (outside California)

(707) 887-7237

P.O. Box 339, Forestville, CA 95436

\*California residents add 6% sales tax

**SOFTWARE SAVINGS!** 25% discount on  
all Amiga & Atari software.

```
IF MID$(Records$(Ptr1),offset,lgth)>
MID$(Records$(Ptr2),offset,lgth) THEN
```

```
SWAP Records$(Ptr1),Records$(Ptr2)
```

```
Ptr1=Ptr1-Limit
```

```
ELSE
```

```
Ptr1=0
```

```
END IF
```

```
WEND
```

```
NEXT
```

```
GOTO HalfIt
```

```
SortDone:
```

```
MENU 3,1,0
```

```
WINDOW CLOSE 3
```

```
' And Now to Print the Report
```

```
DoPrint:
```

```
WINDOW 3,,(440,40)-(608,92),0,1
```

```
COLOR Blu,Yel:CLS
```

```
LOCATE 2,4:PRINT"Now Printing..."
```

```
OPEN "PRT:" FOR OUTPUT AS #3
```

```
GOSUB PageHdg
```

```
FOR n=1 TO RecCnt
```

```
IF LineCnt>55 THEN GOSUB NewPage:GOSUB PageHdg
```

```
IF NewSeq$="ITEM" THEN
```

```
x$=MID$(Records$(n),12,16)
```

```
x$=x$+MID$(Records$(n),1,11)
```

```
x$=x$+MID$(Records$(n),28,46)
```

```
ELSE
```

```
x$=MID$(Records$(n),1,73)
```

```
END IF
```

```
PRINT #3," "+x$
```

```
PRINT #3,SPACE$(30)+MID$(Records$(n),74)
```

```
LineCnt=LineCnt+2
```

```
NEXT
```

```

GOSUB NewPage:PageCnt=0:CLOSE #3
GOTO RptXit

NewPage:
PRINT #3,CHR$(12);
RETURN

PageHdg:
PRINT #3," ":PageCnt=PageCnt+1
PRINT #3,SPACE$(37);
PRINT #3,USING"-##";PageCnt;:PRINT #3,"-"
PRINT #3,SPACE$(8);
PRINT #3,CHR$(27)+"[6w";: 'Set Double width
PRINT #3,"The Household Inventory Program";
PRINT #3,CHR$(27)+"[5w"; 'Set normal width
PRINT #3," "
IF NewSeq$="ITEM" THEN
  PRINT #3," Item=====> Room====>";
ELSE
  PRINT #3," Room====> Item=====>";
END IF
PRINT #3," Pur-Data OrCost C Worth RpCost Serial-Number=>"
PRINT #3,SPACE$(31)+"Comments=====> Add-Date Chg-Date"
PRINT #3," "+STRING$(72,"-")
LineCnt=9
RETURN

RptXit:
WINDOW CLOSE 3
GOTO WaitHere

```

```

' Help Routines (requested via Help Menu)
' -----

```

```

DoHelp:
WINDOW 4,,(408,0)-(631,186),0,1
COLOR Blu,Yel:CLS:LOCATE 2,1
ON i GOTO HlpGen,HlpASIS,HlpROOM,HlpITEM
HlpGen:
PRINT " 'HouseInvPrint' uses a"
PRINT " data file created by"
PRINT " 'HouseInv' as input and"
PRINT " produces three reports"
PRINT " from the contents of"
PRINT " that file.":PRINT " "
PRINT " A third program,"
PRINT " 'HouseInvMaint' is used"
PRINT " for functions related to"
PRINT " the maintenance of the"
PRINT " file itself, (rather than"
PRINT " the contents). "
DrawGadgets HlpB%,HlpB%,bx(),bxtxt$()
A%=HlpB%:B%=HlpB%:GOSUB WaitMouse
GOTO HlpXit
HlpASIS:
PRINT " The report is produced in"
PRINT " the same sequence as that"
PRINT " in which the items were"
PRINT " entered.":PRINT " "
PRINT " This option is only valid"
PRINT " before a report of any"
PRINT " other sequence is pro-"
PRINT " duced.":PRINT " "
GOSUB HelpCommon
DrawGadgets HlpB%,HlpB%,bx(),bxtxt$()
A%=HlpB%:B%=HlpB%:GOSUB WaitMouse
GOTO HlpXit
HlpROOM:
PRINT " The list of items is"
PRINT " sorted into room sequence"
PRINT " before the report is"
PRINT " printed.":PRINT " "
GOSUB HelpCommon:PRINT " "
PRINT " N.B. An 'Order Entry'"
PRINT " report is no longer"
PRINT " available."

```

```

DrawGadgets HlpB%,HlpB%,bx(),bxtxt$()
A%=HlpB%:B%=HlpB%:GOSUB WaitMouse
GOTO HlpXit
HlpITEM:
PRINT " The list of items is"
PRINT " sorted into item sequence"
PRINT " before the report is"
PRINT " printed.":PRINT " "
GOSUB HelpCommon:PRINT " "
PRINT " N.B. An 'Order Entry'"
PRINT " report is no longer"
PRINT " available."
DrawGadgets HlpB%,HlpB%,bx(),bxtxt$()
A%=HlpB%:B%=HlpB%:GOSUB WaitMouse
GOTO HlpXit
HelpCommon:
PRINT " Items which have been"
PRINT " deleted, but not yet re-"
PRINT " moved from the file,"
PRINT " (see 'HouseInvMaint'),"
PRINT " are listed with an '"
PRINT " separating the pieces of"
PRINT " information."
RETURN

HlpXit:
WINDOW CLOSE 4
RETURN
' Various Subprograms
' -----

```

## Listing Four

```

' Listing #4 - Merge in Listing #1 before using.
'
' The Household Inventory File Maintenance Program
' -----
' This is Program #3 of 3: "HouseInvMaint" -file
' maintenance
' Program #1 is the "HouseInv" (main) program
' Program #2 is the "HouseInvPrint" program
'
' Please do not modify the title screen in any way.
' January 1987
'
numbx=4:RecCnt=0:NewCnt=0:m=0:i=0:ErrSw=0:type=0
A%=0:B%=0
DIM bx(numbx-1,6),bxtxt$(numbx-1)
Logo80 3
BldGadgets numbx,bx(),bxtxt$()
' No and OK Gadgets
DATA 28,36,24,16,7,4,0,"No"
DATA 116,36,24,16,7,2,0,"Ok"
' Help Gadgets
DATA 36,172,40,16,7,4,0,"More"
DATA 156,172,40,16,7,2,0,"Ok"
InA%=0:InB%=1
HlpA%=2:HlpB%=3
COLOR Blu,Blk
LOCATE 9,39:PRINT"THE"
LOCATE 11,22:PRINT"H O U S E H O L D I N V E N T O R Y"
LOCATE 13,38:PRINT"SYSTEM"
COLOR Mag,Blk
LOCATE 16,25:PRINT"F I L E M A I N T E N A N C E"
MENU 1,0,1,"Project:"
MENU 1,1,1,"Quit "
MENU 2,0,1,"Help:"
MENU 2,1,1,"General "
MENU 2,2,1,"Initialize "
MENU 2,3,1,"Reorganize "
MENU 2,4,1,"Record Count"
MENU 2,5,1,"Update Count"
MENU 3,0,1,"Maintenance:"
MENU 3,1,1,"Initialize "

```

```

MENU 3,2,1,"Reorganize "
MENU 3,3,1,"Record Count"
MENU 3,4,1,"Update Count"
MENU 4,0,0,""
ON ERROR GOTO InitError
ErrSw=0:GOSUB GetRecCnt
InitCnt:
ON ERROR GOTO 0
IF ErrSw=1 THEN ErrSw=0:RecCnt=0:GOSUB PutRecCnt
ON MOUSE GOSUB GetMouse
ON MENU GOSUB GetMenu
COLOR Yel,Blk:LOCATE 21,23
PRINT"Use Menus to select program function"
WaitHere:
MENU ON:m=0:i=0:WHILE m=0:SLEEP:WEND
MENU OFF:ON m GOTO Quit,Help,Maintain

' Count File Error Routine
' -----
InitError:
WINDOW 2
IF ERR=53 THEN
  ErrSw=1:RESUME InitCnt
ELSE
  ON ERROR GOTO 0
END IF

' Menu Event Routine
' -----
GetMenu:
m=MENU(0):i=MENU(1)
RETURN

' Mouse Event Routine
' -----
GetMouse:
GetGadget A%,B%,bx(),bxtxt$(),type
RETURN

' Wait for Mouse Click
' -----
WaitMouse:
MOUSE ON
type=0:WHILE type=0:SLEEP:WEND
MOUSE OFF
RETURN

' Open Main Data File
' -----
Opndata:
IF RecCnt=0 THEN
  WINDOW 3,,(440,40)-(608,96),0,1
  COLOR Blu,Yel:CLS
  LOCATE 2,3:PRINT"File is empty."
  DrawGadgets InB%,InB%,bx(),bxtxt$()
  A%=InB%:B%=InB%:GOSUB WaitMouse
  ErrSw=1:WINDOW CLOSE 3
  GOTO ODXit
END IF
OPEN "R",#1,"HouseInv.Data",103
FIELD #1,1 AS d1$,10 AS d2$,15 AS d3$,8 AS d4$,6 AS d5$,6 AS
d6$,6 AS d7$,15 AS d8$,20 AS d9$,8 AS d10$,8 AS d11$
ErrSw=0
ODXit:
RETURN

' Open Temporary Data File
' -----
OpenTemp:
OPEN "R",#9,"Temp.Data",103
FIELD #9,1 AS t1$,10 AS t2$,15 AS t3$,8 AS t4$,6 AS t5$,6 AS
t6$,6 AS t7$,15 AS t8$,20 AS t9$,8 AS t10$,8 AS t11$
RETURN

```

```

' Get Record Count
' -----
GetRecCnt:
OPEN"HouseInv.Count" FOR INPUT AS #2
INPUT #2,RecCnt
CLOSE #2
RETURN

' Update Record Count
' -----
PutRecCnt:
WINDOW 3,,(440,40)-(608,96),0,1
COLOR Blu,Yel:CLS
LOCATE 2,4:PRINT"Updating Count"
LOCATE 3,4:PRINT"File."
OPEN"HouseInv.Count" FOR OUTPUT AS #2
WRITE #2,RecCnt
CLOSE #2
WINDOW CLOSE 3
RETURN

' Time to Quit and Return to Basic
' -----
Quit:
MENU RESET
WINDOW CLOSE 2:SCREEN CLOSE 1
END

' Help Routines
' -----
Help:
GOSUB DoHelp
GOTO WaitHere

' Data File Maintenance Routines
' -----
Maintain:
ON i GOTO MInit,MReorg,MRecCount,MUpdCount

' Initialize the Count File to Zero
MInit:
WINDOW 3,,(440,40)-(608,92),0,1
COLOR Blu,Yel:CLS
LOCATE 2,3:PRINT"This option will"
LOCATE 3,3:PRINT"delete any records"
LOCATE 4,3:PRINT"on file."
DrawGadgets InA%,InB%,bx(),bxtxt$()
A%=InA%:B%=InB%:GOSUB WaitMouse
WINDOW CLOSE 3
ON type GOTO MInXit,MInOK
MInOK:
IF RecCnt>0 THEN
  KILL"houseInv.Data":KILL"HouseInv.Data.info"
END IF
RecCnt=0:GOSUB PutRecCnt
MInXit:
GOTO MaintXit

' Reorganize the Data File
MReorg:
GOSUB Opndata:IF ErrSw=1 THEN MRXit
WINDOW 3,,(440,40)-(608,92),0,1
COLOR Blu,Yel:CLS
LOCATE 2,3:PRINT"Reorganizing the"
LOCATE 3,3:PRINT"Data File."
GOSUB OpenTemp:NewCount=0
FOR n=1 TO RecCnt
  GET #1,n
  IF d1$="0" THEN
    NewCnt=NewCnt+1
    LSET t1%=d1$:LSET t2%=d2$:LSET t3%=d3$:LSET t4%=d4$
    LSET t5%=d5$:LSET t6%=d6$:LSET t7%=d7$:LSET t8%=d8$
    LSET t9%=d9$:LSET t10%=d10$:LSET t11%=d11$
    PUT #9,NewCnt
  END IF
NEXT

```

continued...

```

CLOSE #1:CLOSE #9
KILL"HouseInv.Data.info":KILL"HouseInv.Data"
NAME"Temp.Data.info" AS "HouseInv.Data.info"
NAME"Temp.Data" AS "HouseInv.Data"
WINDOW CLOSE 3
RecCnt=NewCnt:GOSUB PutRecCnt
MRXit:
GOTO MaintXit

' Show Count of Records on File
MRecCount:
WINDOW 3,,(440,40)-(608,92),0,1
COLOR Blu,Yel:CLS
LOCATE 2,4:PRINT"Record Count"
LOCATE 4,2:PRINT USING"Committed: #####";RecCnt
DrawGadgets InB%,InB%,bx(),bxtxt$()
A%=InB%:B%=InB%:GOSUB WaitMouse
WINDOW CLOSE 3
GOTO MaintXit

' Update Count File to Match Data File
MUpdCount:
GOSUB Opendata:IF ErrSw=1 THEN UCXit
WINDOW 3,,(440,40)-(608,92),0,1
COLOR Blu,Yel:CLS
LOCATE 2,3:PRINT"Counting..."
n=1
UCGet:
GET #1,n
IF EOF(1) THEN UCDone
n=n+1:GOTO UCGet
UCDone:
WINDOW CLOSE 3
RecCount=n-1:GOSUB PutRecCnt
UCXit:
GOTO MaintXit

MaintXit:
GOTO WaitHere

' Help Routines (requested via Help Menu)
' -----
DoHelp:
WINDOW 4,,(408,0)-(631,186),0,1
COLOR Blu,Yel:CLS:LOCATE 2,1
ON i GOTO HlpGen,HlpInit,HlpReorg,HlpRecCnt,HlpUpdCnt
HlpGen:
PRINT" 'HouseInvMaint' performs"
PRINT" the necessary maintenance"
PRINT" functions on the data"
PRINT" file created by"
PRINT" 'HouseInv'. It is not"
PRINT" concerned with the"
PRINT" contents of the file, but"
PRINT" rather with the file"
PRINT" itself.":PRINT" "
PRINT" A third program,"
PRINT" 'HouseInvPrint' is used"
PRINT" to print reports based on"
PRINT" the contents of the data"
PRINT" file."
DrawGadgets HlpB%,HlpB%,bx(),bxtxt$()
A%=HlpB%:B%=HlpB%:GOSUB WaitMouse
GOTO HlpXit
HlpInit:
PRINT" This function, available"
PRINT" in this and the main pro-"
PRINT" gram, allows you to start"
PRINT" from scratch.":PRINT" "
PRINT" The main data file is a"
PRINT" random access file with a"
PRINT" 'count' file being used"
PRINT" to keep track of the num-"
PRINT" ber of data records.":PRINT" "
PRINT" 'Initialize' causes the"

```

```

PRINT" data file to be deleted"
PRINT" and the count file to be"
PRINT" reset to zero.":PRINT" "
PRINT" It should be the first"
PRINT" function used."
DrawGadgets HlpB%,HlpB%,bx(),bxtxt$()
A%=HlpB%:B%=HlpB%:GOSUB WaitMouse
GOTO HlpXit
HlpReorg:
PRINT" When records are deleted"
PRINT" they remain in the data"
PRINT" file with a 'deleted'"
PRINT" flag turned on. When"
PRINT" 'reviewed' they show up"
PRINT" with '*' field separators"
PRINT" and may be restored by"
PRINT" selecting them.":PRINT" "
PRINT" 'Reorganize' copies the"
PRINT" data file dropping all"
PRINT" deleted records. There-"
PRINT" fore, after reorganiza-"
PRINT" tion, any previously del-"
PRINT" eted records are gone"
PRINT" forever."
DrawGadgets HlpB%,HlpB%,bx(),bxtxt$()
A%=HlpB%:B%=HlpB%:GOSUB WaitMouse
GOTO HlpXit
HlpRecCnt:
PRINT" 'Record Count' provides"
PRINT" you with a count of the"
PRINT" number of records pres-"
PRINT" ently in the data file."
PRINT" The count will include"
PRINT" any records that may have"
PRINT" been previously deleted"
PRINT" if the file has not been"
PRINT" reorganized."
DrawGadgets HlpB%,HlpB%,bx(),bxtxt$()
A%=HlpB%:B%=HlpB%:GOSUB WaitMouse
GOTO HlpXit
HlpUpdCnt:
PRINT" 'Update Count' counts the"
PRINT" number of records in the"
PRINT" data file, and updates or"
PRINT" creates the count file"
PRINT" with that number.":PRINT" "
PRINT" Should be used only if"
PRINT" the count file is"
PRINT" deleted or becomes"
PRINT" unreadable."
DrawGadgets HlpB%,HlpB%,bx(),bxtxt$()
A%=HlpB%:B%=HlpB%:GOSUB WaitMouse
GOTO HlpXit

HlpXit:
WINDOW CLOSE 4
RETURN

```

' Various Subprograms

' -----

•AC•

## Amazing Reviews...

# TIC

## A Tiny Battery-Backed Clock/Calendar for the Amiga™

By John Foust

*Byte-by-Byte* has introduced the *TIC*, a tiny battery-backed clock that plugs into the second mouse port. The Tic retails for \$59.95.

The Tic is small black plastic box with a nine-pin DB-9 connector, the same connector as on joysticks and the mouse. It fits in the second mouse port with room to spare for the mouse connector itself.

During boot-up, the Tic program reads the current date and time from the box. After this, the Tic can be safely removed, freeing the use of the joystick port. The date and time is preserved even when the Tic is disconnected from the Amiga.

Mine has been in and out of the Amiga for several weeks without trouble, and it retained the date and time when I carried it to the local user group meeting. According to *Byte-by-Byte*, it should retain the time for at least a year away from the Amiga, or otherwise without power. The battery recharges itself with power from the Amiga port.

Upon booting the Tic disk, an extremely large icon named 'Intro' fills almost a quarter of the screen. Double clicking it brings up a single IFF screen with instructions leading you to the Tic documentation, and the standard warning about making a backup of this disk.

All Tic documentation is on disk; it comes with no printed documentation. The text and pictures of the manual are organized with a demo version of the Infominder program, another program from *Byte-by-Byte*. This works well. The Infominder is nearly self-running for even the novice user. Just clicking the mouse and selecting menu items step through the manual. The pictures are almost too simple. Reading English is no requirement for this manual.

The Tic software is easy to use. The program named 'Tic' works both from an icon on the Workbench and from the CLI. The program 'TimePiece' is a rehash of the Workbench Clock program, with more Tic-specialized options.

There are two options for the Tic software, in all its incarnations. You can either set the Amiga system time from the time stored and maintained in the Tic, or set the time stored in the Tic from the system time. When double-clicked, the program presents the options 'Get Tic' and 'Set Tic' in a window.

Presuming you have already set the system time from the CLI, or from another program such as Preferences, selecting 'Set Tic' would copy the system time to the Tic. The program then displays a window with a message from the Tic. In most cases, this is the time and date. If the Tic is not present, or not working, a message appears to that effect. The errors are explained in the on-disk documentation.

From the CLI, the 'Tic' program responds to the 'tic ?' method of learning which options the program can use, just as many AmigaDOS CLI commands do. It accepts two options, 'tic get' or 'tic set'. With the command 'tic' alone, the program defaults to 'tic get', so adding just the word 'tic' to your startup sequence will read the time from the Tic.

The TimePiece program looks much like the Workbench Clock program. It has extra options for setting the colors of the clock, and selecting military and 24 hour time. Unlike the Workbench clock, and similar to a real-world analog clock, you can move the hands of the clock to set the time. Aside from these creeping featurisms, it has 'get Tic' and 'set Tic' menu options.

A third program exists on the disk, an automatic installation program. 'Install Tic' will add the command sequence 'tic get' to the file 'startup-sequence' in the 's' scripts directory of the current Workbench disk.

I will not trust my disks to a program that purports to modify my startup sequence file. With several Amiga peripherals, I have had to spend extra time with the manual to determine exactly what this program planned to add to my startup sequence, so I could make the changes myself.

It only took a few minutes with the Tic manual to determine that it only planned to add the line 'tic get' to the beginning of the startup sequence. What if I had a peripheral that needed to be initialized before I wanted to run the Tic program? The automatic nature of the installation program fails in that case. Of course, the user with that many fussy Amiga peripherals will be able to customize their own startup sequence, so this is no detriment of the Tic.

Frankly, I think Amiga developers are not presented with a reasonable solution to the installation of devices like the Tic. So many products want their initialization software to be

*continued...*

added to the startup sequence file. Developers must present an easy path for novice users to do so. Expert users will want to modify the startup sequence themselves. In truth, the situation is only worse on most other computers. On the Amiga, one can imagine a system of moving an icon to a special directory so that dumb peripherals are initialized in a smart way on startup.

The disk also contains a program called ScreenSaver. It blanks the screen after three minutes of non-use. After no mouse movements or keystrokes, the screen is changed to black, and a dark purple Byte-by-Byte logo jumps from spot to spot on the screen. (The logo is a pie chart with one slice pulled away.) You can enable and leave the program with Amiga and shift keys.

The second mouse port is connected directly to the Paula chip, so they had to be extra careful in designing the circuit, according to a Byte-by-Byte spokesperson. On the networks, I have heard reports of users who have blown their Paula chips, necessitating a trip to the repair shop, but this was probably due to a faulty joystick, or unusual amounts of static electricity.

There is one big minus to the Tic for some people. The Tic did not collide with my expansion bus memory board, but it DOES collide with several other popular memory boards, including the C Ltd. aMEGA, Microbotics and Alegra memory boards.

According to Scott Peterson, president of Byte-by-Byte, they did not realize this conflict until the last minute. To correct it, they have developed a program that switches the functions of the mouse ports, so that the Tic can go in the front mouse port, and the mouse can function normally in the rear mouse port.

Engineer Jude Katsch who designed the Tic has done many other hardware designs for Amiga and Byte-by-Byte. Byte-by-Byte hints that Katsch has many other Amiga projects in development.

Time and date stamping is a necessity in a computer that does real work. In all, the Tic represents a great value for people who need the correct time and date on their files. The Tic software is incredibly easy to use, and best of all, unobtrusive in the Amiga environment.

•AC•

**TIC \$59.95**

**Byte-by-Byte**

Suite 150

9442 Capital of Texas Highway

Austin, Texas 78759

(512) 343-4357

# 8 MEGABYTES

## Now RS DATA's New POW•R•CARD

### Let's You Play Like The Big Boys.

Playing games on your Amiga can be a great deal of fun, but let's be honest — there's more to life than playing games. Now you can turn your computer into a real-life professional machine with the POW•R•CARD from RS DATA Systems.

The POW•R•CARD is a powerful new expansion board which allows you to mature in your computer use with greater flexibility in multi-processing and multi-tasking.

POW•R•CARD starts you off with a 2 Meg capability and allows you to grow with upgrades to a huge 8 Meg RAM expansion, all on the same board so you don't waste valuable slot space. That means you can run more software without fear of Guru Meditation Numbers, out-of-memory crashes or any other small system

boo-boos! What's more, you won't have to rob your piggy bank because POW•R•CARD offers this tremendous growth at a cost lower per megabyte than you'll find anywhere.

With your new POW•R•CARD, memory expansion is as easy as 1-2-3. The POW•R•CARD and enclosure will pass through the Buss without modification for even greater expansion. So you don't have to play games with your data anymore. Graduate to bigger and better things with the POW•R•CARD from RS DATA!

Upcoming Products from RS DATA:

- New Hard Disk System, 20 & 40 megabyte memory.
- 4 Port Parallel card.

- 4 Port Serial Card, allowing more serial type peripheral use.
- 4 Slot Expansion System with horizontal board placement for system height reduction.
- Much, much more!!!

The POW•R•CARD is available now from your local Amiga dealer . . . or call RS DATA today!

**RS DATA™**  
**Systems**

7322 Southwest Freeway  
Suite 660  
Houston, Texas 77074  
713/988-5441

# SECRETS OF SCREEN DUMPS

*" What happens if the ratio of height to width on the screen is different from the ratio that your printer can print? "*

*by Nathun Okun*

One of the most important uses to which the Amiga has been put and in which it excels is graphics. The beautiful images that pass onto the monitor screen are literally wonders to behold. However, barring the carrying of your Amiga everywhere you go (and, while much lighter than an IBM-type PC, it is definitely NOT a laptop machine), how do you show these images to others or use them in documents of your creation?

The way most people do it is to use one or another special printer routines that make a copy of the screen onto a piece of paper: the so-called "screen dump" to their printer. Some of these printer routines are special-purpose, such as the program GRABBIT by Discovery Software International or several other public domain programs. Others are incorporated into more general-purpose programs, such as Deluxe Paint by Electronics Arts, where the screen dump is merely one of several program functions.

The alternative to the screen dump is to photograph the screen with a good camera and then have the photographs developed and duplicated, as necessary. This option is obviously the best for many uses, but it is expensive and it requires a dedicated application that usually only a professional or highly-motivated and skilled hobbyist can supply. Screen dumps are the method of choice by most users.

The program performing the screen dump does not "see" the screen, but knows where in memory the screen is being mapped and which pixels are being colored in which color. It must also be able to make the printer select the proper print color or, if a black-only printer, the proper dot pattern to approximate a shade of grey for the color of the given pixel. In most cases, the screen dump logic in these programs uses the extensive firmware in the operating system to drive the printer that was selected for the PRT: device and that was defined using the printer set-up sections of the Preferences program built into the Amiga's operating system.

Once a specific printer has been selected and special user-definable options given, the operating system turns over details of printer commands to a special-purpose program called the "printer driver." Unfortunately, there is rather little standardization in such critical areas as graphics and initialization from printer to printer, even on occasion with printers made by the same manufacturer! Therefore, each printer driver is tailored to translate the generic internal printer commands used by the Amiga PRT: device to single or multiple command chains that cause the printer to do that particular action, be it to print, move the print head without printing, move the paper up and down, or perform whatever else the printer in question can do. Obviously, some printers simply cannot perform some tasks, so it is still up to the Amiga user to intelligently select his printer for a given task and to match it to the software that he wants to run. Graphics, for example, virtually always means some sort of dot-matrix printer (be it impact, ink jet, thermal transfer, or laser transfer) that can create randomly shaped dot patterns under program control to build up a picture a line at a time.

The controls for text printing are rather simple and conform for the most part to the almost-universal "American Standard Code for Information Interchange,"--ASCII for short. Most of the settings for the printers found in Preferences are for such text printout, with graphics as an afterthought. Text printing can be done by all printers, while graphics is somewhat more restricted by the printer's design, as mentioned above.

As we are interested here in screen dumping, it is obvious that the Amiga user who wants to use this form of screen duplication MUST have a printer capable of this dot-matrix manipulation. He also MUST have a printer driver on his program disk that can be selected to control his particular printer (some printers not on the list can be controlled by selecting the driver for a printer that is on the list that most closely matches their capabilities, but there is no guarantee that such substitutions will work in the demanding job of translating the orders for a pixel-by-pixel screen duplication effort). Alternatively, if the ability to save the screen to disk

*continued...*

exists, then the user can of course save the screen to disk for eventual printing on somebody else's printer, though the same printer/printer driver requirements still apply.

I use an 80-column Epson FX-80 printer and have access to an Epson JX-80 color printer. The following discussion will therefore be oriented to these widely used printers. While the general ideas will be the same for all printers, the exact numbers given may have to be adjusted if another, non-Epson-compatible printer is to be used. Experiment!

I do not know if the other printer drivers have problems, but the "Epson" and "Epson\_JX-80" printer drivers in the Workbench V1.1 "devs/printers" directory have a major bug that causes them to overlap the bottom row of dots of the previous 9-dot-high pass with the top row of this pass. (This may not have been by accident. The printers occasionally leave a narrow gap between lines due to slop in the head movement and perhaps the printer driver programmer wanted to prevent this. The overlap technique is not the way to do this, however!) On all Workbench V1.2 disks, new Epson and Epson\_JX-80 drivers have fixed this problem. If available, anyone using these printer drivers should copy the V1.2 drivers onto all of their V1.1 disks, replacing the old V1.1 versions. To my knowledge, all of the printer drivers on the V1.2 disks can be backfitted to the V1.1 (and V1.0?) disks. I will use the V1.2 drivers exclusively in this discussion.

One of the major considerations in a screen dump is making the paper image have the same ratio of height to width as the screen image so as to prevent circles from becoming ellipses and so forth when printed. The screen dump programs that I am familiar with do this perfectly, which is nice. However, this same function has another side that can ruin a screen dump! This other side is simply: What happens if the ratio of height to width on the screen is different from the ratio that your printer can print? Lurking behind this simple question is a whole new set of print functions that few people know about because these functions are not advertised in the documentation (if any) concerning these programs.

Each printer has a resolution of dots/inch in both the horizontal and vertical directions (they may be different in each direction!) and the printer driver knows how many printer dots to use to print a screen pixel. The crux is that the number of dots on the printed page per pixel is fixed regardless of the size of the printed image. This means that to print a postage-stamp-sized screen dump or a banner-sized screen dump each pixel on the screen has a fixed dot size on the page.

How then does the screen dump program adjust for the differences in the height to width ratio and page size? Simple: THE PROGRAM DOES SURGERY ON YOUR SCREEN PICTURE AS IT IS PRINTING IT!!! What you see on the screen is not necessarily what you get on paper, which can cause you a lot of heartburn, believe me! If the picture is too small on the page, the program selectively deletes entire rows of pixels and closes the gap using some algorithm that evenly spaces these "cuts" over the entire page--the smaller the page image, the more numerous the cuts. Conversely, if

the image on the page is larger than the pixel/dot ratio allows, then the program duplicated rows of dots in a similar evenly-spaced manner, thickening the objects along the rows by this amount to form "stretch marks" on the page. It does this in both the horizontal direction (adds or deletes entire top-to-bottom columns of pixels along the long page axis) and the vertical direction (adds or deletes entire left-to-right rows of pixels from margin to margin along the short page axis), whichever is necessary to fit the printed page. (One is reminded of an old Greek legend about a bed to which one is stretched or cut to fit...) If you have a screen with text on it, this is especially distressing, as the changes are HIGHLY visible!!!

### **What to do?**

Obviously, if the printed page image is not changed too much (by your personal definition), you may ignore the problem. Even so, it may be to your benefit to know about the fix and how it can be used to your benefit in the future. If you do not like somebody else taking an axe to your work of art, read on.

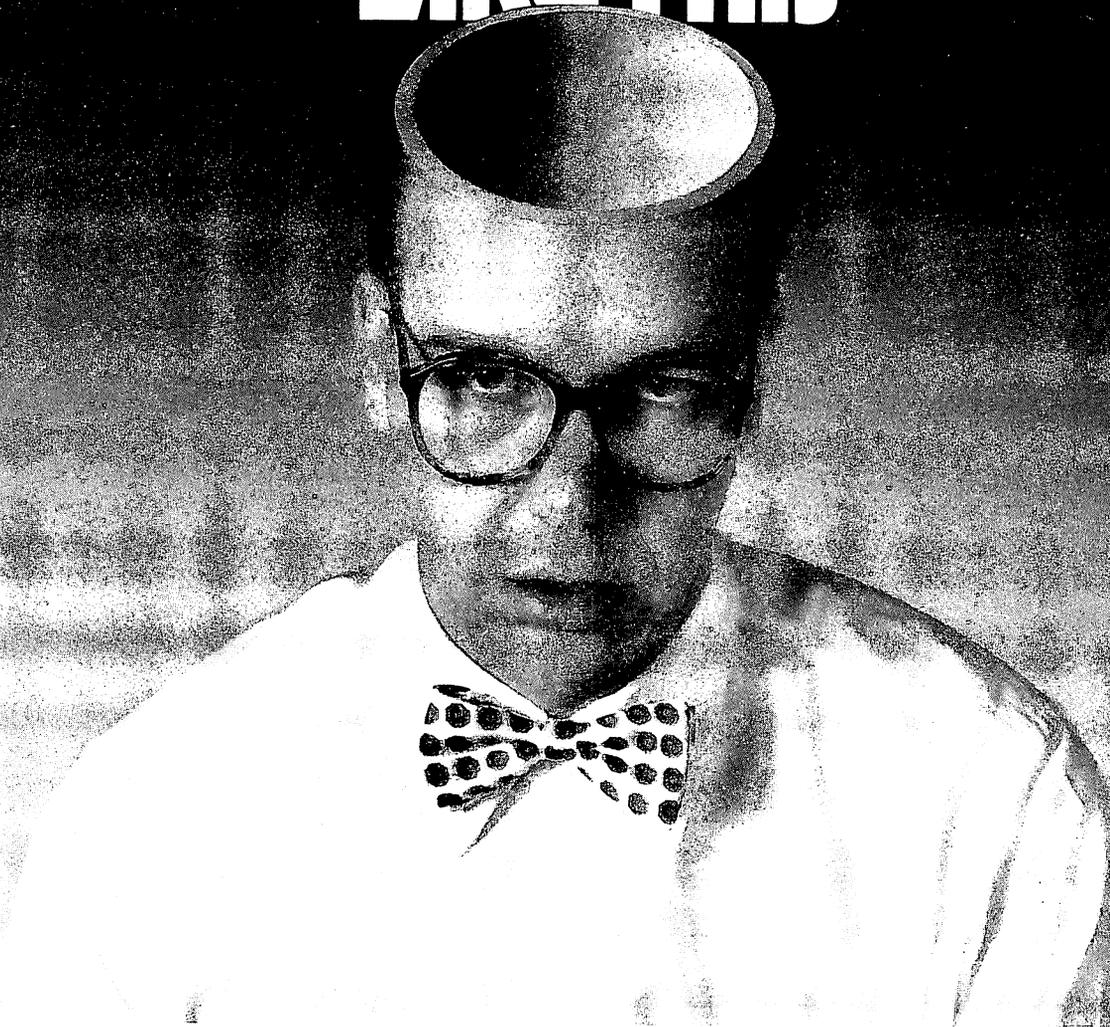
When I first noticed the problem, I thought of sending nasty notes to Commodore-Amiga, but what good would that do now, if ever (those of you who have dealt with Commodore in the past know what I am talking about)? I then sat down and thought that maybe I didn't see the whole picture (!) and perhaps this "bug" of the screen dump routines is an undocumented "feature" of the operating system. Luckily, I was correct!

I looked at the picture and thought, how is it determining the size to print out? The obvious answer, I hoped, was the page size settings in Preferences: Left Margin, Right Margin, Page Type, and Lines per Page. I later added Lines per Inch.

I began by changing the values for the margins. The default settings when you first run Preferences is 5 for the left margin and 75 for the right margin, which is OK for text printing most of the time, such as when using the V1.2 "Notepad" program. Beginning with the normal horizontal aspect screen dump, I set the left margin to 1 and began to adjust the value of the right margin. Lo and behold, the printed screen dump image changed with it, having one or more columns of dots cut or stretched across the printed page. The height of the image increased or decreased, too, to keep the image ratio constant, but it also ceased having any stretch marks or cuts when the surgery ended for the width. After a lot of waste paper through my printer, I hit upon the value of the right margin of 74 virtually eliminating the surgery of the image's width. With the horizontal aspect screen dump, that takes up about half of a U.S. standard 8.5" x 11" page, this was enough to fix the entire problem. If you wish to move the left margin, you must add the same amount to the right margin (out to the maximum of 80 columns) to keep the width constant.

Having successfully conquered the horizontal screen dump surgery problem, I set the print aspect to vertical in Preferences and tried seeing what changing the page length accomplished, initially keeping my 1 and 74 margin settings and the normal 6 lines/inch line density. I found that the

# You Wouldn't Work Like This



## Why Should Your Amiga?

These days an Amiga with only 512K memory is operating at a fraction of its' potential.

Fortunately we're here to change all that.

We're Expansion Technologies, and we've developed the best RAM expansion board available for your Amiga 1000.

The Escort 2.

It's a 2 megabyte, auto-configuring card that meets all the known standards and then some.

It's also a uniquely designed two-slot card cage that offers incredible flexibility.

Like the ability to upgrade to a whopping 4 megabytes of memory. Or if you prefer you can add a hard disk controller card, or an 86-pin buss return or...well, you get the idea.

And it's fast.

We've utilized a no wait-state design so it keeps perfect pace with your Amiga.

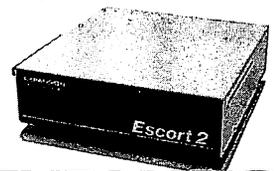
All this for less than \$650.

We're even readying products for the new 500 and 2000 series Amigas. Among them hard drives, controller cards and memory expansion.

All of this from Expansion Technologies. A company whose people have been making Commodore products for years, not weeks.

So if your Amiga's a bit light headed give us a call at 415/656-2890. Or write us at 46127 Landing Parkway, Fremont, CA 94538.

*Dealer Inquiries Invited.*



**EXPANSION  
TECHNOLOGIES**

Amiga is a trademark of Commodore-Amiga, Inc.

## Rescue Your Recipes!



**Compu Cuisine**  
Over 200 Recipes  
in 8 categories.

Edit and add own recipes.

Organize and search files  
by category or ingredient.

Send check or money order for

for the Amiga! \$29.95 to:  
Adept Software  
P.O. Box 700702  
San Jose, Ca. 95170

special margin settings were still needed to eliminate the cuts or stretch marks of vertical columns of pixels, just like with the horizontal aspect. When I used any other setting for the margins that changed the horizontal width of the image, there were always some cuts or stretch marks on the printed image, no matter what I did with the page length. Obviously these special margin settings are an absolute for all screen dumps if no alteration of the printed page from the screen image is to be allowed. Once I realized this, I left the 1 and 74 margins alone.

However, in the vertical direction with the standard U.S. default value of 66 lines per page I found that several stretch marks were being evenly spaced across the narrow width of the page to enlarge the image (vertical screen dumps are about 50% bigger in both directions than the horizontal screen dumps), in exactly the same way that some columns had been stretched when I used more than 74 with my right margin in the horizontal aspect screen dumps. I steadily cut back the lines per page to 53 and the stretching changed to cutting of several rows of pixels, while at 54 there still were some stretch marks on the printed image. Obviously, the correct value for lines per page was about 53.5, which is not possible to enter into Preferences.

Crossing my fingers, I switched from 6 lines/inch to 8 lines/inch and changed the lines per page to 70. The result had a few rows of pixels cut, so I changed to 71 lines per page. After careful study I found only one row of pixels cut exactly in the center of the printed page. When I chose 72, I

found four rows with stretch marks. As mentioned, trying to adjust the margins only added problems in the other direction.

I next changed to high resolution, 640 x 400, screens of several Deluxe Paint pictures that I had created that had text on them. The results were identical to the low resolution printed images as to the image surgery, indicating that the low resolution and medium resolution screen dump logic automatically doubles the size of the pixels to keep 200 lines, 400 lines, 320 horizontal pixels, or 640 horizontal pixels the same width and height on the printed page! Neat trick!

My conclusions are that the vertical print option is not quite as clean as the horizontal option, but the choice is a single row of pixels cut using 71 lines per page or four spaced rows of stretch marks if 72 is chosen. I decided to use the value of 71 for lines per page because it was closest to being perfect, though there may be situations where stretch marks are preferable to even a single lost row of pixels. The fact that the vertical aspect is so much bigger than the horizontal aspect makes a single row of pixels lost much less important.

Therefore, if you wish to have a screen dump to an Epson-compatible printer using standard U.S. 8.5" x 11" paper with minimum changes to the image as it is printed, you must make the following changes in Preferences:

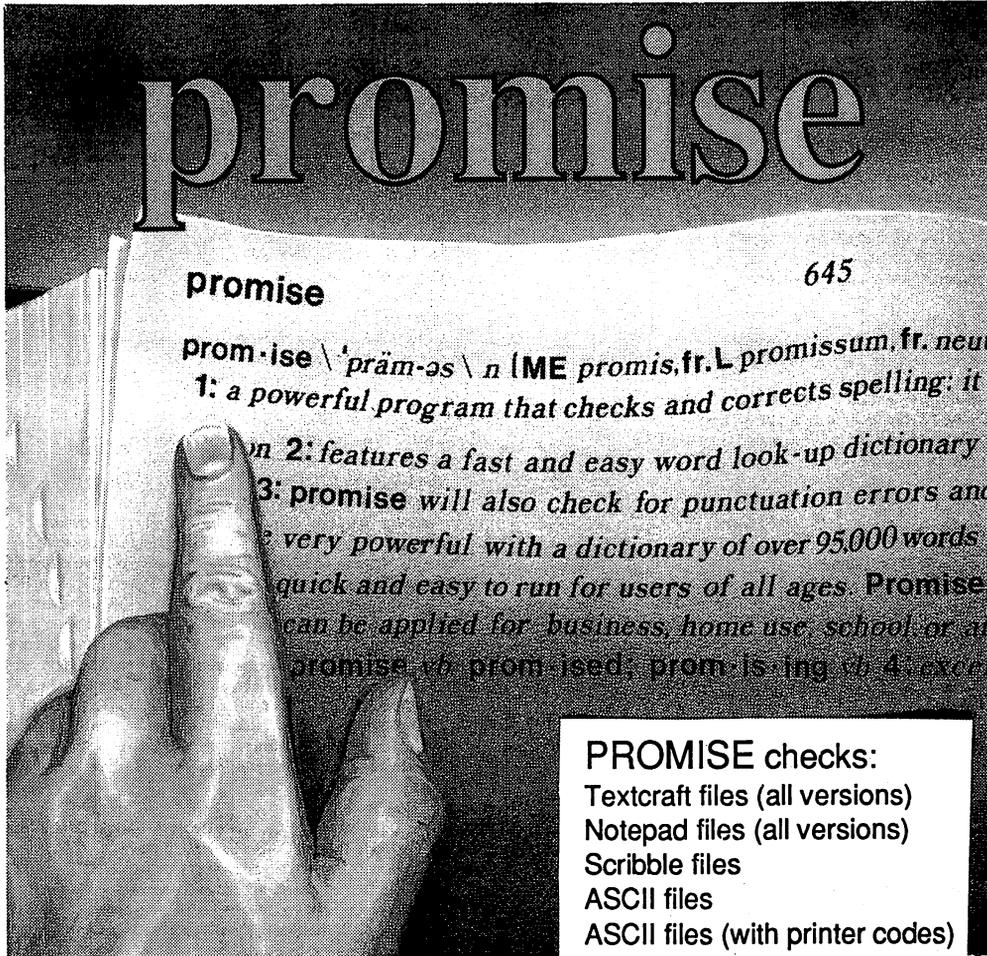
LEFT MARGIN = 1  
RIGHT MARGIN = 74  
LINES/PAGE = 71  
LINES/INCH = 8

The new lines/page and lines/inch are only needed for vertical printing, but I make these my normal settings for all disks where screen dumps are the primary printer use. I also have changed my normal print dumps to vertical aspect. **REMEMBER TO CHANGE THESE BACK TO THE DEFAULT (OR OTHER DESIRED) SETTINGS WHEN TEXT PRINTING IS TO BE DONE USING THAT DISK!** I put a message in my Startup-Sequence showing my new printer settings to remind me. Also remember to test these settings when using non-Epson-compatible printers or any printer with a different page width or length (FX-100, etc.); they may have to be adjusted for your printer.

While this was a problem when trying to do "neutral" screen dumps, the ability to shrink or enlarge a printed screen dump can be useful if you can tolerate the nibbling or stretching of the image that this entails. Obviously, graphics can tolerate this more than text. The problem here was mainly a lack of adequate documentation of this capability of the Amiga coupled with a default printer page set-up table that was not designed for screen dumps (at least not from an Epson printer).

You now have another graphics tool to experiment with and use to your benefit. Good printing!

•AC•



**PROMISE checks:**  
 Textcraft files (all versions)  
 Notepad files (all versions)  
 Scribble files  
 ASCII files  
 ASCII files (with printer codes)

## Introducing promise

At last, the introduction of a new generation of SPELL CHECKING PROGRAMS. Promise contains many features to meet the expectations of demanding users.

### Features:

- A 95,000 word dictionary that resides in memory and uses advanced indexing methods.
- A spell checking rate of 18 words per second! (1080 words per minute).
- **PROMISE** features a SPELL HELP that actually helps you spell any word correctly. SPELL HELP will operate with virtually any word processor.
- SPELL HELP takes full advantage of the Amiga's multi-tasking capabilities, allowing the spell help feature to be used with most word processors.
- The ability to create custom dictionaries at the mere click of the mouse button.
- Total mouse and gadget control, giving an ease of use never before available in spell checking programs.
- **PROMISE** will check for rudimentary punctuation errors.

**Promise** makes word processing faster and easier. *Promise* is a must for any serious writer, student, or business person.

Amiga is a trademark of Commodore Amiga

Requires 512K Amiga.



THE OTHER GUYS

(800) 942-9402

The Other Guys  
 55 N. Main Suite 301D  
 Logan, Utah 84321



NOW SHIPPING

49.99

# If you are reading *Amazing Computing*<sup>TM</sup> for the first time, you have not seen *Amazing Computing*<sup>TM</sup>. Look what you have missed!

## Volume 1 Number 1 Premiere February 1986

**Super Spheres** By Kelly Kauffman An ABASIC Graphics program  
**Date Virus** By John Foust There is a disease that may attack your Amiga  
**EZ-Term** by Kelly Kauffman An ABASIC Terminal program  
**Miga Mania** by Perry Kivolowitz Programming fixes and mouse care  
**Inside CLI** by George Musser a guided insight into the AmigaDOS™  
**CLI Summary** by George Musser Jr. A removable list of CLI commands  
**AmigaForum** by Bela Lubkin A quick trip through Composeve's Amiga SIG  
**Commodore Amiga Development Program** by Don Hicks  
**Amiga Products** A listing of present and expected products.

## Volume 1 Number 2 March 1986

**Electronic Arts Comes Through** A look at the new software from EA  
**Inside CLI: part two** by George Musser George continues his investigation of CLI and ED  
**A Summary of ED Commands**  
**Live!** by Rich Miner A review of the Beta version of the Live! frame grabber  
**Online and the CTS Fabite 2424 ADH Modem** reviewed by John Foust  
**Superterm V 1.0** By Kelly Kauffman A terminal program written in Amiga Basic  
**A Workbench "More" Program** by Rick Wirch  
**Amiga BBS numbers**

## Volume 1 Number 3 April 1986

**Analyze!** a review by Ernest Viveiros  
**Reviews of Racter, Barataccas and Mindshadow**  
**Forth!** The first of our on going tutorial  
**Deluxe Draw!** by Rich Wirch An Amiga Basic program for the artist in us all.  
**Amiga Basic, A beginners tutorial**  
**Inside CLI: part 3** by George Musser George gives us PIPE

## Volume 1 Number 4 May 1986

**SkyFox and Articfox** Reviewed  
**Build your own 5 1/4 Drive Connector** By Ernest Viveiros  
**Amiga Basic Tips** by Rich Wirch  
**Scrimper Part One** by Perry Kivolowitz A C program to print your Amiga screen  
**Microsoft CD ROM Conference** by Jim O'Keane  
**Amiga BBS Numbers**

## Volume 1 Number 5 1986

**The HSI to RGB Conversion Tool** by Steve Pietrowicz Color manipulation in BASIC  
**AmigaNotes** by Rick Rae The first of the Amiga music columns  
**Sidecar A First Look** by John Foust A first "under the hood" look  
**John Foust Talks with R. J. Mcal at COMDEX™**  
**How does Sidecar affect the Transformer** an interview with Douglas Wyman of Simile  
**The Commodore Layoffs** by John Foust John looks at the "cuts" at Commodore  
**Scrimper Part Two** by Perry Kivolowitz  
**Marauder** reviewed by Rick Wirch  
**Building Tools** by Daniel Kary

## Volume 1 Number 6 1986

**Temple of Apehal Trilogy** reviewed by Stephen Pietrowicz  
**The Halley Project: A Mission in our Solar System** reviewed by Stephen Pietrowicz  
**Flow:** reviewed by Erv Bobo  
**Textcraft Plus a First Look** by Joe Lowery  
**How to start your own Amiga User Group** by William Simpson  
**Amiga User Groups**  
**Mailing List** by Kelly Kauffman a basic mail list program  
**Pointer Image Editor** by Stephen Pietrowicz  
**Scrimper: part three** by Perry Kivolowitz  
**Fun With the Amiga Disk Controller** by Thom Sterling  
**Optimize Your AmigaBasic Programs for Speed** by Stephen Pietrowicz

## Volume 1 Number 7 1986

**Aegis Draw: CAD comes to the Amiga** by Kelly Adams  
**Try 3D** by Jim Meadows an introduction to 3D graphics  
**Aegis Images/ Animator:** a review by Erv Bobo  
**Deluxe Video Construction Set** reviewed by Joe Lowery  
**Window requesters in Amiga Basic** by Steve Michel  
**ROT** by Colin French a 3D graphics editor  
**"I C What I Think"** Ron Peterson with a few C graphic programs  
**Your Menu Sir!** by Bryan D. Catley programming menus in Amiga Basic  
**IFF Brush to AmigaBasic "BOB"** Basic editor by Michael Swinger  
**Linking C Programs with Assembler Routines on the Amiga** by Gerald Hull

## Volume 1 Number 8 1986

**The University Amiga** By Geoff Gamble Amiga's inroads at Washington State University  
**MicroEd** a look at a one man army for the Amiga  
**MicroEd, The Lewis and Clark Expedition** reviewed by Robert Frizelle  
**Scribble Version 2.0** a review  
**Computers in the Classroom** by Robert Frizelle  
**Two for Study** by Robert Frizelle a review of Discovery and The Talking Coloring Book  
**True Basic** reviewed by Brad Grier  
**Using your printer with the Amiga**  
**Marble Madness** reviewed by Stephen Pietrowicz  
**Using Fonts from AmigaBasic** by Tim Jones  
**Screen SaVer** by Perry Kivolowitz A monitor protection program in C  
**Lattice MAKE Utility** reviewed by Scott P. Evernden  
**A Tale of Three EMACS** by Steve Poling  
**.bmap File Reader in Amiga Basic** by Tim Jones A look into the .bmap files

## Volume 1 Number 9 1986

**Instant Music** Reviewed by Steve Pietrowicz  
**Mindwalker** Reviewed by Richard Knepper  
**The Alegria Memory Board** Reviewed by Rich Wirch  
**TxED** Reviewed by Jan and Cliff Kent  
**Amazing Directory** A guide to the sources and resources  
**Amiga Developers** A listing of Suppliers and Developers  
**Public Domain Catalog** A condensed listing of Amicus and Fred Fish PDS Disks  
**Dos 2 Dos** review by Richard Knepper Transfer files from PC/MS-DOS and AmigaBasic  
**MaxiPlan** review by Richard Knepper The Amiga version of Lotus 1-2-3  
**Gizmoz** by reviewed by Peter Wayner A collection of Amiga extras!  
**The Loan Information Program** by Brian Catley basic prog. to for your financial options  
**Starting Your Own Amiga Related Business** by William Simpson  
**Keep Track of Your Business Usage for Taxes** by James Kummer  
**The Absoft Amiga Fortran Compiler** reviewed by Richard A. Reale  
**Using Fonts from AmigaBasic, Part Two** by Tim Jones  
**68000 Macros on the Amiga** by Gerald Hull Advance your program's ability.  
**TDI Modula-2 Amiga Compiler** by Steve Faiwieszewski Looking at an alternative to C

## Volume 2 Number 1 1987

**What Digi-View Is... Or, What Genlock Should Be!** by John Foust  
**AmigaBasic Default Colors** by Bryan Catley  
**AmigaBasic Titles** by Bryan Catley  
**A Public Domain Modula-2 System** reviewed by Warren Block  
**One Drive Compile** by Douglas Lovell using Lattice C with a single drive system  
**A Megabyte Without Megabucks** by Chris Irving An Internal Megabyte upgrade  
**Digi-View** reviewed by Ed Jakober  
**Defender of the Crown** reviewed by Keith Conforti  
**Leader Board** reviewed by Chuck Raudonis  
**Roundhill Computer System's PANEL** reviewed by Ray Lance  
**Digi-Paint....** by New Tek previewed by John Foust  
**Deluxe Paint II ....** from Electronic Arts previewed by John Foust

## Volume 2 Number 2 1987

**The Modem** by Joseph L. Rothman efforts of a BBS Sysop  
**MacroModem** reviewed by Stephen R. Pietrowicz  
**GEMINI** or "It takes two to Tango" by Jim Meadows Gaming between machines  
**BBS-PCI** reviewed by Stephen R. Pietrowicz  
**The Trouble with Xmodem** by Joseph L. Rothman  
**The ACO Project....Graphic Teleconferencing on the Amiga** by S. R. Pietrowicz  
**Flight Simulator II....A Cross Country Tutorial** by John Rafferty  
**A Disk Librarian in AmigaBASIC** by John Kennan  
**Creating and Using Amiga Workbench Icons** by Celeste Hansel  
**AmigaDOS version 1.2** by Clifford Kent  
**Amiganotes...The Amazing MIDI Interface build your own** by Richard Rae  
**AmigaDOS Operating System Cells and Disk File Management** by D. Haynie  
**Working with the Workbench** by Louis A. Mamakos Programming in C

## Volume 2 Number 3

**The Amiga 2000™** by John Foust A First look at the new high end of the Amiga™ line.  
**The Amiga 500™** by John Foust A look at the new low priced Amiga  
**An Analysis of the New Amiga PCs** by John Foust Speculation on the New Amigas  
**Gemini! Part II** by Jim Meadows The concluding article on two-player games  
**Subscripts and Superscripts in AmigaBASIC** by Ivan C. Smith  
**The Winter Consumer Electronics Show** by John Foust  
**AmigaTriX** by Warren Block Those little shortcuts that make using the Amiga™ easier  
**Intuition Gadgets** by Harriet Maybeck Tolly A journey through gadget-land using C  
**Shanghai** reviewed by Keith M. Conforti  
**Cheesmaster 2000 & Cheesmate** reviewed by Edwin V. Apel, Jr.  
**Zing!** from Meridian Software reviewed by Ed Bercovitz  
**Forth!** by Jon Bryan Get stereo sound into your Forth programs.  
**Assembly Language on the Amiga™** by Chris Martin  
**Roomers** by the Bandito Genlocks are finally shipping, and MORE!!!  
**AmigaNotes** by Richard Rae Hum Busters... "No stereo? Y not?..."  
**The AMICUS Network** by John Foust "CES, user group issues and Amiga Expo"

*To Be Continued.....*

# Amazing Computing™

Your Resource to the Commodore Amiga™

Amazing Computing™ has been offering the Amiga community the best in technical knowledge and reviews for the Commodore-Amiga™ since our first issue in February 1986.

- We were the first magazine to document CLI
- We were the first to show Sidecar™ from COMDEX™ in full detail.
- We were the first to document a 5 1/4 drive connector
- We were the first with a 1 Meg Amiga upgrade hardware project!
- We were the first magazine to offer serious programming examples and help.
- We were the first magazine to offer Public Domain Software at reasonable prices.
- We were the first magazine with the user in mind!

However, Amazing Computing™ will not rest on past achievements. The Commodore-Amiga™ has more surprises for you and we are ready to cover them. We even have a few tricks that will "Amaze" you.

For a catalog of past issues of Amazing Computing™ and a list of available Public Domain Software, please fill out the form below and send with \$1.00 to:

PiM Publications Inc.  
P.O.Box 869  
Fall River, MA 02722.

Back issues are still available at \$4.00 each (Foreign orders, please add \$1.00 U.S. per issue for P. & H).

## Yes! Amaze Me!

Please start my subscription to Amazing Computing™ with the next available issue. I have enclosed \$24.00 for 12 issues in the U.S. (\$30.00 Canada and Mexico, \$35.00 overseas) All funds must be in U.S. Currency drawn on a U.S. Bank.

Please circle your selection: Special Catalog!

Subscription PDS (as noted) Back Issues Sub Renewal

Public Domain Software:

\$6.00 each for subscribers (yes, even new ones!)

\$7.00 each for non subscribers.

Name \_\_\_\_\_

Street \_\_\_\_\_

City \_\_\_\_\_ St. \_\_\_\_\_ ZIP \_\_\_\_\_

Amount enclosed \_\_\_\_\_

AMICUS: A1 A2 A3 A4 A5 A6 A7 A8 A9 A10 A11 A12 A13 A14 A15 A16  
Fred Fish: FF1 FF2 FF3 FF4 FF5 FF6 FF7 FF8 FF9 FF10  
FF11 FF12 FF13 FF14 FF15 FF16 FF17 FF18 FF19 FF20  
FF21 FF22 FF23 FF24 FF25 FF26 FF27 FF28 FF29 FF30  
FF31 FF32 FF33 FF34 FF35 FF36 FF37 FF38 FF39 FF40  
FF41 FF42 FF43 FF44 FF45 FF46 FF47 FF48 FF49 FF50 FF51 FF52 FF53  
BACK ISSUES: \$4.00 each (foreign orders add \$1.00 each for Postage and Handling  
VOL.1#1 VOL.1#2 VOL.1#3 VOL.1#4 VOL.1#5 VOL.1#6 VOL.1#7  
VOL.1#8 VOL.1#9 VOL.2#1 VOL.2#2 VOL.2#3

Mass. Residents, please add 5% sales tax on PDS orders

AC487

# Micro-Systems Software —Bigger and Better for the Amiga

DATABASE  
SPREADSHEET  
WORD PROCESSOR

## **NEW** - ORGANIZE! DATABASE MANAGER, VERSION 1.0

Mailing lists! Club memberships! Patient records! Client files! Video tape libraries! Phone call logs! Nearly anything that needs to be filed, sorted or calculated is a candidate for **Organize!**

In seconds, **Organize!** can scan your files, locate information, and display or print in the format you want. Use it to print form letters with the Mailmerge function of Scribble!. Or calculate fields and do statistical analyses of your files with many of the same built-in math functions from Analyze!.

Easily design input forms and output reports with the mouse and pull-down menus. Just as simply - store, sort, review and print. The file size is limited only by disk space and the format is compatible with the industry standard dBASE format.

End your paper shuffle! Get **Organize!** today.

**Only \$99.95**

## **UPGRADED** -

### ANALYZE! SPREADSHEET, VERSION 2.0

#### ANALYZE! FEATURES:

- Pulldown menu interface (mouse-driven).
- Large spreadsheets with efficient memory usage.
- Dedicated function keys for common commands.

#### NEW ADDITIONS:

- **Business Graphics;** print bar, stacked bar, pie graphs in 2 or 3-D; line, X-Y, area graphs; all in 4 or 8 colors; data from spreadsheets; IFF format; view up to 4 graphs at same time; instantly redraw graphs when data changes; ranges, labels, titles, legends, rotation, scaling; fast and effective!
- **Command Macros;** save keystrokes; create templates.
- **Sorting;** rearrange row or column data quickly.
- **File Icons;** access spreadsheets via icons or names.

**NEW PRICE Only \$149.95**

## **IMPROVED** - SCRIBBLE!

### WORD PROCESSOR, VERSION 1.0

#### SCRIBBLE! FEATURES:

- Pulldown menu interface (mouse-driven)
- Multiple windows; edit/cut & paste 4 documents on screen.
- Preview; see final form on screen before printing.

#### NEW ADDITIONS:

- **Spellcheck;** expandable 40,000 word dictionary; check word, all words on screen, or entire document; alternative spellings shown.
- **Mailmerge;** print form letters, mailing labels; create data file with Scribble! or Organize!
- **File Icons;** access documents via icons or names; copy documents by pulling icons across workbench.
- Expanded Memory Support; for larger documents.
- More Amiga Keys; menu commands from keyboard or mouse.
- More Flexibility; Wordstar™ commands; scrolling while cut/paste; improved file operations.

**Still only \$99.95**



MICRO-SYSTEMS  
SOFTWARE, INC.

4301-18 OAK CIRCLE, BOCA RATON, FL 33431  
IN FL. CALL (305)391-5077 VISA, MASTERCARD

**For Nearest Dealer Call  
1-800-327-8724**

See your local dealer or call:

**Brown-Wagh  
Publishing**

1-800-451-0900

1-408-395-3838 (in California)

16795 Lark Ave., Suite 210, Los Gatos, CA 95030

# Using Function Keys with MicroEmacs

*"I find using the programmable keys faster and more flexible than trying to use the mouse every time you want to issue a command."*

By Greg Douglas

Workbench 1.2 now includes a version of the MicroEmacs text editor. It uses pull down menus as well as keyboard input for almost 70 command control sequences. I find about 15 of the commands to be the most useful in day to day work.

A nice feature of MicroEmacs is that you can remap or program the function keys, Help key, and all of the numeric keypad keys. This assisting in remembering numerous control sequence commands. I find using the programmable keys faster and more flexible than trying to use the mouse every time you want to issue a command. However, the mouse is handy for the less frequently used commands.

Remapping the function keys is fairly simple. By pressing CTRL-X CTRL-K (Set key), you will see the prompt 'def:' at the bottom of the screen. If you are defining F1 to F10, first insert the <ESC> keystroke (CTRL-Q) to tell MicroEmacs that it is a function key you are defining, then enter the command string sequence described below followed by a <RETURN>. If you define the Help key or any key on the numeric keypad, all you need to enter is the command sequence and a <RETURN>.

After remapping several keys to handle command control sequences, I could not find any way of saving the new key definitions. The manual is not clear on this point. There is a 'Load keys' option on the Extras menu, but no 'Save keys' option. After using the example in the documentation and a couple of hours experimenting with no success, I ended up writing a small program using Lattice C. The program allows me to redefine the functions keys and the numeric keypad very easily.

It involves setting up a string array that contains all 24 definable keys. Each key definition can be up to 80 characters long. My program handles 15 characters, which means you can assign multiple commands to a single programmable key.

For example, to copy the current line to the line below, you would issue the following commands manually:

```
CTRL-A Move to beginning of line
CTRL-K Kill entire line
      (actually copies the line to a Kill Buffer)
CTRL-Y Yank Kill Buffer
      (restores contents of the Kill Buffer)
CTRL-M Insert a carriage return
      (opens a new line)
CTRL-Y Yank Kill Buffer
      (insert the Kill Buffer again on the new line)
```

These commands can form a single function that can be assigned to a programmable key, such as F1. By pressing the F1 function key, the current line is duplicated.

Octal ASCII values represent the control character sequences such as:

```
CTRL-A = 001   CTRL-Y = 031
CTRL-X = 030   CTRL-K = 013
ESC     = 033   CTRL-V = 026
```

The file that contains the key definitions has a particular format. The first element of the array, func\_key[0], is the auto start string. This string tells MicroEmacs what initial commands will be performed when it is executed. If you do not want initial commands to be issued the string should look like this:

```
"~\n"
```

The "~\n" indicates the end of the function definition. In the duplicating-a-line example, the command string would look like this:

```
"\01\13\31\15\31~\n" command string for F1
^A^K^Y^M^Y control characters
```

Since this is the first key definition string it is assigned to the F1 key. The order in which the strings are placed in the array is the order in which they are assigned to the definable keys; starting with F1 through F10, Help key, 0 through 9, :, Enter, and '-' keys.

*continued...*

The command string does not necessarily have to be edit commands, it can be a text string or phrase. The last key definition in the program illustrates this. The '-' key is defined to print the C language comment marks. When the '-' key is pressed, '/\* \*/' is displayed and the cursor is moved to the middle of the comment marks so that you can start entering text in the comment area.

The program lists all of the definable keys, whether they are defined or not. As your requirements change, you can define them later as needed or change any key to something else. You can keep several different key definition files on disk for different tasks.

Once all the keys have been defined, compile the program and execute it. The program creates a definition file called 'emacs\_config'. To load the key definition file from within MicroEmacs, just select the 'Load keys' option from the Extras menu and it will prompt you to enter the key definition filename.

Another way to load the key definition file is to move the 'emacs\_config' file to a directory called 'config' in the root directory of the disk. When MicroEmacs starts up, it looks to see if it can find a file named 'config/emacs\_config'. If it can find it, it automatically loads the file. Now you can use the defined function keys with MicroEmacs.

If you have not discovered MicroEmacs, it is on the Extras disk in the Tools directory. The documentation is about 30 pages and quite thorough. I have errors in the documentation that need to be corrected. The first is on page 25 or so; it defines several Load Key sequences out of order. I have included the corrected sequence order in Listing One.

Second, the documentation indicates that the configuration file should be in the 's' directory as 'config/emacs\_config'. The file 'config/emacs\_config' must be placed at the root directory of the disk for auto load to work properly.

MicroEmacs can be a powerful editing tool. The ability to customize your own function keys gives you significant power and flexibility that most other editors do not have. I hope you find this information helpful when using MicroEmacs.

## Listing One

```

/* File: EMACS_KEYS.C =====
Greg Douglas  2 Jan 87

Description: This program will create a file that
MicroEMACS can read to redefine keyboard functions.
Use the EXTRAS menu LOAD KEY function to load the key
definitions file.

Compiled using Lattice 3.10 and linked using Blink
=====*/

#include <stdio.h>
#define NUMKEYS 24 /* Number of key definitions */

void main()
{
/* string array can be up to 80 characters - I used 15 */
static char func_key[][15] =
/*--- key definitions-----*/
{ "~\n", /* Auto start string */
  "\01\13\31\15\31~\n", /* F1 = ^A^K^Y^M^Y Dup line */
  "\30\04~\n", /* F2 = ^X^D Delete line */
  "\33~\n", /* F3 = ESC- Set Mark */
  "\27~\n", /* F4 = ^W Kill region */
  "\33W~\n", /* F5 = ESC W Copy region */
  "\31~\n", /* F6 = ^Y Yank buffer */
  "\33V~\n", /* F7 = ESC V Screen up */
  "\26~\n", /* F8 = ^V Screen down */
  "\30\23~\n", /* F9 = ^X^S Save file */
  "\30\06~\n", /* F10= ^X^F Save file/Ex */
  "\12~\n", /* HELP = ^J Indent */
  "\30n~\n", /* 0 = ^Xn Next window */
  "\13~\n", /* 1 = ^K Del to EOL */
  "2~\n", /* 2 = 0 */
  "\33>~\n", /* 3 = ESC > End of buff */
  "4~\n", /* 4 = 0 */
  "5~\n", /* 5 = 0 */
  "6~\n", /* 6 = 0 */
  "7~\n", /* 7 = 0 */
  "8~\n", /* 8 = 0 */
  "\33<~\n", /* 9 = ESC < Top of buff */
  "\30p~\n", /* . = ^Xp Prev window */
  "\30\7~\n", /* Enter = ^X^G Goto Line */
  "/* */\2\2\2\2~\n" /* - Comm't mark */
};

int i;
FILE *out_file, *fopen();

out_file = fopen("emacs_config","w");

/* Write func_key array to file */
for (i = 0; i < NUMKEYS + 1; ++i)
    fputs(func_key[i],out_file);

fclose(out_file);
}

/* ----- End of Program -----*/

```

•AC•

# SoundScape . . . Power Play for the AMIGA.

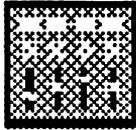


## Pro MIDI Studio



The most powerful performance and recording software on any computer. The recording studio-like environment provides complete facilities for routing, recording, editing, transposition and playback of any musical performance. As new modules are introduced, you can "install" them at any time. Music can be performed by the internal sampled sound synthesizer, or with any external MIDI equipment. Record from the QWERTY keyboard or any external MIDI source, including keyboards, guitar and pitch followers. Synchronize with, or provide MIDI clock information, including MIDI Song Pointers. The complete flexibility of the system makes your imagination the only limit to its power.

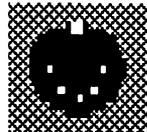
- Number of notes and tracks determined by available memory
- MIDI patch panel links program modules
- Install new modules at any time
- Up to 16 internal instruments at one time
- Complete sample system with editing, looping, ADSR envelopes, velocity sensitivity, and pitchbend.



- Up to 160 sampled sounds at one time
- Save and load IFF note and sample files
- Quantize to any multiple of MIDI clock beats
- "Match" mode eases learning of a song
- Complete MIDI sequence and song editing
- Route, merge, split, or bounce any track to any other.



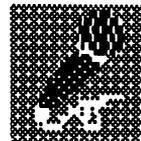
## MIDI Interface



- Necessary for any program which supports MIDI to communicate with MIDI equipment.
- Completely compatible with the standard Amiga MIDI interface
  - MIDI In, Out, and Thru connectors
  - Plugs into the serial port



## Sound Digitizer



With the SoundScape Sound Digitizer, any sound may be sampled and modified by the Amiga, including voice. IFF File compatibility enables these samples to be used as musical instruments, sound effects, or speech with any IFF compatible music or animation system.

- High quality
- Highest possible fidelity from the Amiga
- Stereo or mono
- Variable sample rates
- Mike and line inputs
- Digitally controlled volume on each channel
- IFF Sample File compatible
- Software included for sampling, editing, and MIDI performance functions

### Available From Your AMIGA Dealer.

SoundScape Pro MIDI Studio	\$149.00
AMIGA MIDI Interface	\$ 49.00
SoundScape Audio Digitizer	\$ 99.00

mimetics™  
corporation

...the professional software source!!

P.O. Box 60238 Sta. A, Palo Alto, CA 94306 (408) 741-0117

Amiga is a trade mark of Commodore Business Machines

Prices and availability subject to change without notice

# **New Products! New Pricing! The Best Just Got Better, Again.**

**ASDG Incorporated** is pleased to announce that we're accepting orders for our newest board product, the **8M**.

The **8M** is a Zorro compatible Amiga expansion card which can contain 2, 4, 6, or 8 megabytes of the same fast reliable memory you've come to expect from our **2M**, now recognized as the best quality Amiga ram expansion product produced yet.

The **8M** is the **only** Amiga ram expansion product supporting 6 megabytes on a single board. The Amiga contains autoconfiguring support for boards of 4 Mbytes and 8 Mbytes but not of 6 Mbytes. Our **8M**, populated with 6 Mbytes automatically presents two autoconfigure data sets to your Amiga. The first as a 4 Mbyte board, the second as a 2 Mbyte board.

Thus, if you already have a 2Mbyte expansion board, you can bring your Amiga up to its full potential using only one more board slot. (By the way, the **8M** would make a great addition to any TurboAmiga system!)

All ASDG memory boards have: Zero Wait States, Full Zorro Compatibility, Full Autoconfiguring, The **ASDG Recoverable Ram Disk** and One Year Warranty.

## **Recoverable Ram Disk News!**

The exclusive ASDG Recoverable Ram Disk has recently been updated to allow crash protected ram disks to encompass all 8Mbytes your Amiga can handle. The RRD comes with every ASDG memory board and provides the fastest crash survivable disk anywhere! With the ASDG Recoverable Ram Disk, Guru's won't get you down any more!

**8M/2 \$899    8M/4 \$1349    8M/6 \$1799    8M/8 \$2199**  
**Mini-Rack-C \$195    Mini-Rack-D \$325**  
**2M is now only \$599!**

***ASDG Incorporated***  
***280 River Rd Suite 54A***  
***Piscataway N.J. 08854***  
***(201) 540 - 9670***

Deliveries of the **8M** begins approximately April 30th, 1987. ASDG pays standard UPS delivery in continental U.S. N.J. residents add 6% sales tax. Special pricing for SYSOPS and User Group officers. Amiga trademark of Commodore-Amiga Inc. TurboAmiga trademark of CSA Inc. ASDG Recoverable Ram Disk, 8M/2, 8M/4, 8M/6, 8M/8, Mini-Rack-C, Mini-Rack-D trademarks of ASDG Incorporated.

# AmigaTrix II

*"...with my printer printing out full-page pictures,  
my disk drive gronking mutedly in the background,  
and sound coming from the computer in stereo again,  
it's a lot more fun to use."*

**By Warren Block**

Yes, it's back, and again I am confronted with the dilemma of trying to organize this batch of ideas into some type of order. In order to leave room for hedging, I'll just draw some arbitrary lines, giving each section a subheading.

***One warning:** if you attempt something I've mentioned without adequate knowledge or tools and turn your computer into an expensive paperweight, I'll be very sorry--but that's all. All these tips have worked for me, but some take a delicate touch and should not be attempted by novices. In other words, be careful and cautious, and don't be afraid to seek the help of an expert.*

Keeping this in mind, off we go.

## **Screen-Blanking?**

It seems that many in the microcomputer community are just starting to recognize the dangers of leaving a single unchanging image on a monitor for a long period of time. But shutting the monitor off is hard on the picture tube. Because of this, several "screen-blank" programs that darken the screen after a set period of inactivity have appeared to keep your monitor's phosphors safe from harm. But there is an easier method available to owners of monitors that also act as TV sets (like the Sony KV-1311)--just switch it to the TV mode. Since broadcast video rarely leaves an unchanging image on the screen for any length of time, that expensive tube is safe. The only problem that comes to mind is the individual's "commercial tolerance."

## **Invasion Of The Gronkers**

Once upon a time, Amigas were created with a noisy, gronking little disk drive. I know that this story is true, because I have one of those older Amigas, and that nasty grinding sound has slowly driven me insane (why else do you think I keep writing these AmigaTrix articles?). One day, I had reached my limit--I found some synthetic Teflon oil, attached a thin applicator tube, and squirted small amounts onto the rails on which the head slides. With a penlight and eye-level access to the disk drive slot, they are not difficult to get to (if you aren't sure what the rails look like, get help!).

Watch out for the read/write head--it does not need to be oiled, and in fact dislikes it very much. After this procedure, the gronk is still there, but greatly diminished.

## **Prices, Prices**

"Several companies are manufacturing stereo speaker systems for the Amiga. I was contemplating the purchase of a set of these, but when I realized that the price tag was larger than both speakers combined, it started my admittedly "different" brain working. I went to a local discount store and browsed through the portable cassette player shelves. Some time later, with only a fading memory of a red "Sale!" tag, I found myself standing outside holding a medium-sized AM/FM/cassette unit, with detachable speakers, five-band equalizer, and (here's the important part) auxiliary line input jacks. By plugging the Amiga's sound output into those jacks, I have created a set of "amplified speakers" with a radio and cassette deck thrown in for free, all for a grand total of fifty bucks. The particular unit I got is a Panasonic RX-C36, and I am very pleased with it. Incidentally, I was very surprised by the sound quality of this little gadget.

## **Ink Away**

Am I the only one who owns a Canon PJ-1080A color inkjet printer? A well-made printer with magnificent color that makes no sound other than a polite squirting noise--I could write a column on this little beauty, but that's for another time and place. What I have to relate here are some little details that may help the fortunate few make better use of their little squirts. Others may also glean some information of value from my inky rantings.

First off, let's describe a page as the printer would view it: 640 pixels across and 920 high, with a resolution of 80 dots per inch both ways. So what, right? Well, with a little calculating and experimenting, it turns out that you can print out a full-page image **sideways** with the correct aspect ratio (pretty close, anyway)--but there's a catch. As far as I know, this trick only works with the Images drawing program from Aegis, because Images lets you adjust output size. Still, it may be adaptable. Here we go:

*continued...*

Make sure that you have the correct driver installed in the "devs" directory, and that the Preferences "aspect" is set to vertical.

Load a single sheet of inkjet paper and set it even with the top of the metal bail (this will work with roll paper, too, but you'll have to cut it to the right length after printing).

Turn printer on (paper will advance a bit).

Hand advance paper two "clicks" on the knob--this is a very small amount, necessary to center the image on the paper.

Use the menu option "Set Printer" to set the image size to 840 pixels wide and 640 pixels high. Note that a number even a small bit greater than 640 will not work--the driver knows what the printer is capable of, and will steadfastly refuse to print.

Print the picture (yaaay!).

Those with long attention spans will have noticed a problem--since there is only one sheet of paper in the printer, it will stop with an "out-of-paper" error before it has finished with the picture. Keep reading--help is on the way! (Hey you, in the back--I heard that.)

### Serial Stuff

"A lot of people have made a big deal about how the Amiga's serial port connector is "non-standard." Actually, it does follow the standard, with custom additions of its own. Because of this, it is unwise to try to use 25-conductor serial cables with this port--it can cause a modem or serial printer to make a brief explosive sound... But a solution is available: just use 9-conductor serial cables. This type of cable was very widely available until just lately, when the MS-DOS "standard" intruded and made 25-conductor cables popular. After eliminating impractical choices and shopping around, I came up with two cost-effective possibilities for Amiga serial cables. The first is simply obtaining surplus parts, one-ended cables and such, and wiring up your own. For some reason, this doesn't appeal to many people, but it does have advantages. For one thing, if you have built a cable yourself, you know for sure that there are no "funny" internal connections.

The second possibility is a low-cost, 9-conductor commercial cable. These are not all that common anymore, but I did find some that were economically priced and nationally available. They are sold by *Jameco* (1355 Shoreway Road, Belmont, CA 94002, 415-592-8097), and they are on page 49 of their current catalog, part numbers MMS-2206 and MFS-2206, depending on what type of connector your serial device needs. These cables won't eliminate the white-mice-and-pixie-dust routine it sometimes takes to get RS-232 serial lines working, but they do make it easier.

## SKETerm 2

SKE Software has kept a promise that was made last January. Our customers told us what SKETerm™ needed to make it the most outstanding communications/emulator package available for the Amiga™ and we listened. We listened so well in fact, we completely rewrote SKETerm™ so that now it's faster and more feature enriched than ever before.

Just look at some of the features SKETerm provides:

- Script file processing to speed you through connections with costly bulletin boards.
- User definable macro keys to store Long key sequences for immediate execution.
- Extensive onLine help - no books to lose!
- Text displays on your screen at speeds up to 9600 baud.
- Split screen mode for conference sessions so what you're typing doesn't get mixed in with text you're receiving.
- Windowed Xmodem, Xmodem CRC, Xmodem checksum, Kermit, SKEfer and autochop!
- Baud rates up to 19200; 7 or 8 bit character length; even, odd, mark, or no parity; 1 or 2 stopbits.
- Data capture to any disk or file including the RAM: disk as well as a Hardcopy toggle to print your session as it happens.
- Supports any asynchronous modem up to 19200 baud with auto redial for auto-dial modems. Set up a call list to dial multiple PhoneBook entries as many times as you like or until a complete connection is made.
- Unsurpassed terminal emulation including TTY, ADM3A, ANSI, VT100, with applications mode, function key & character graphic support, and D200 with support for 60 function keys.
- Intuition support using pop-down menus as well as quick "hot" keys so you don't have to take your hands off the keyboard.
- Multitasking so you can run other programs, type or print files while online. You can even start a NewCLI while in the middle of a session while continue to behave as if you were using a stand-alone terminal.

**PRICE \$49.95**

Plus \$2.50 shipping (FL res. add 5%) VISAMC/CHK

SITE LICENSE AVAILABLE  
AVAILABLE NOW!  
Call or write TODAY for your copy!

**SKE Software Company, Inc.**  
2780 Cottonwood Court  
Clearwater, FL 33519  
(813) 787-3111

## Wired For Sound

So you don't want to buy a separate amplifier (even a cheap one like I mentioned above) just to hear your computer in stereo? How about using a pair of the little headphones meant for use with Walkman-type cassette players? All it takes is a trip to the local Radio Shack for some cable adapters. The parts required are:

#274-369 Stereo Playback Y-Adapter

#274-1555 Mini Jack Coupler

plus a dual RCA to RCA cable (any plain old stereo patch cord) and a set of the newer, lightweight headphones with a 1/8" jack. To connect it up, begin by plugging the RCA patch cord into the jacks on the back of the computer. Now look straight-on at the RCA jacks on the Y-adapter; one of them should have a red plastic insulator. Plug the cord from the computer into the Y-adapter, making sure that the plug from the Amiga's right channel goes to the red jack on the Y-adapter. Now plug the Y-adapter into the female-to-female coupler. Plug your headphones into the other end of the coupler (see Figure 1). That's it. Granted, there aren't any luxuries like a volume control with it, but most people find that the volume is about right as it is.

## Paper-Out Problem Patch #1

A quick way of fixing the paper-out problem that most (all?) dot-matrix printers have: tape a 3x5-inch file card over the paper-out switch. Occasionally this can cause paper to feed crookedly, but it beats nothing.

Alternatively, some of those printers have a DIP switch that can be set so the printer ignores the paper-out problem (the Canon doesn't). If neither of these appeals to you, go on to the next hint.

## Paper-Out Problem Patch #2

This one is drastic, but workable. Open up your printer and add a toggle switch in parallel with the paper-out switch. Particular details like where to drill the big hole in the cover for mounting the switch, whether to use a normally-open or normally-closed switch, and how to tap into the paper-out switch wires depend on your printer. In the Canon, the paper-out switch has its own little wiring harness on a connector--very accessible and easy to work with. It makes you wonder if they were planning for people like me.

The big advantage of this method is that it makes it possible to disable the paper-out switch easily, yet it can also be re-enabled easily. The big disadvantages? It voids the warranty, for one thing. Many people may not like a great big toggle switch hanging out of their printer, either.

## Yet Another Cable Hack

This is a good one--less than eight bucks for parts! When I first bought my Amiga, I couldn't afford the whole system, so I used my Commodore 1702 composite-video monitor. Later I sold it to help pay for a Sony monitor. In the meantime, I didn't have a monitor at all--but I still wanted to use my

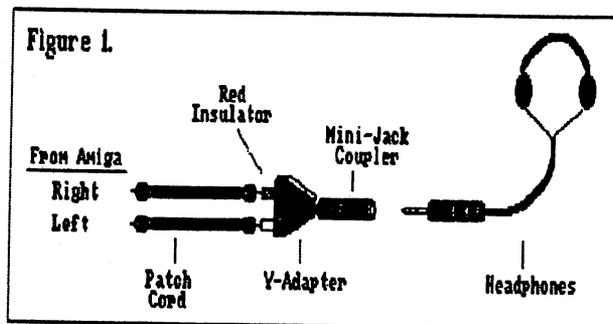
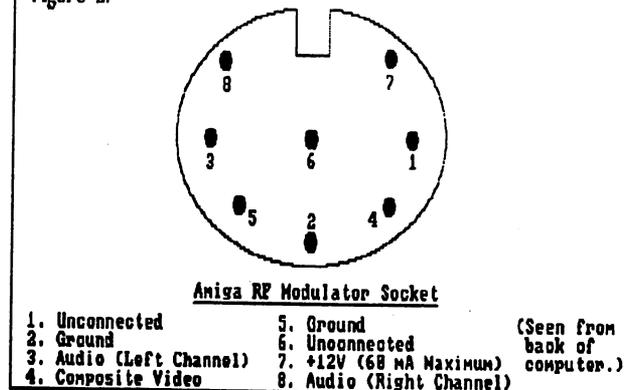


Figure 2.



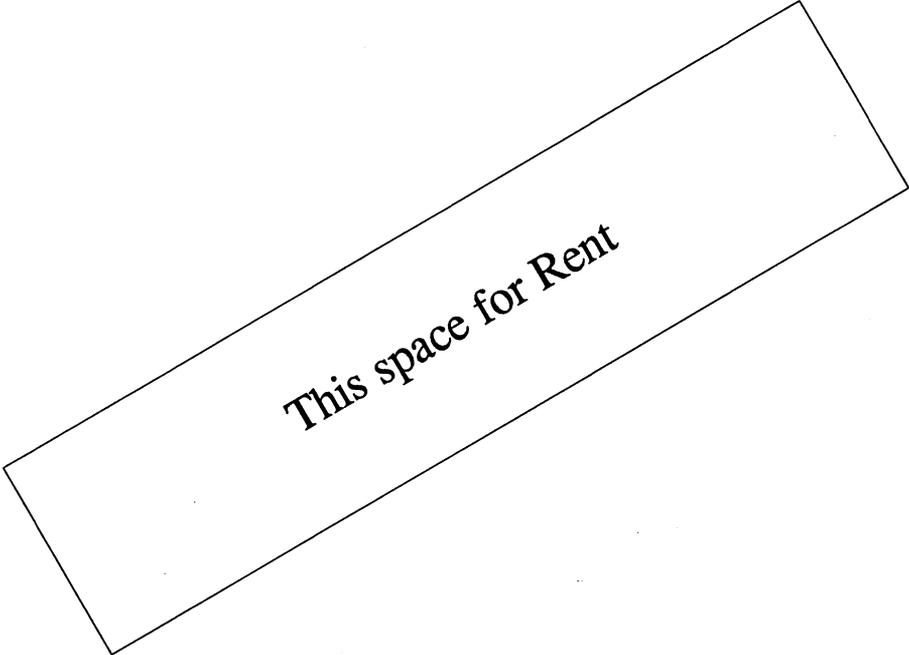
computer. While scanning through a Radio Shack sale flier, I saw an RF modulator on sale for \$2.79, including hookup data. This wondrous little gizmo takes composite video and transforms it magically into an NTSC signal (that's TV, to you and me). The reason for the utterly cheap price: it was meant to be used with the TI99 computer (RIP). I ran down to the local store and picked one up. They have since been discontinued, but a lot of stores and mail-order places still have them. After obtaining the modulator comes the tough part--finding an 8-pin DIN connector to fit the Amiga's port. This takes some doing. Since the C-64 uses an 8-pin DIN plug for video connections, you may be able to find one at a place that sells that type of thing, like a large discount store. Now there is one more tough part--figuring out which pin on the port is which. Luckily, I managed to work this one out (see Figure 2.)

## But That's Not All!

No, I don't work for Radio Shack, Canon, or Panasonic. But with my printer printing out full-page pictures, my disk drive gronking mutedly in the background, and sound coming from the computer in stereo again, it's a lot more fun to use. Now if I could only find my optical mouse...

•AC•

# *Amiga Developers*



*This space for Rent*

This space could have been yours at a rate you would not believe. Amazing Computing™ is the Amiga users' magazine, dedicated to helping the Amiga user get the most out of their machine. Amazing Computing™ is now sold at over 1200 locations world wide, and is read by Amiga users who want to do more with their Amigas. These Amiga users are searching for an application program or hardware item you have developed.

If you are an Amiga Developer who is trying to get the most advertising for a new product without running a budget like the US Government, then give us a call. If money is no object, we are certain other magazines will also be glad to help.

Advertising Sales  
PiM Publications Inc.  
P.O. Box 869  
Fall River, MA 02722  
(617) 678 4200

***Amazing Computing™: Your Resource to the Commodore-Amiga™***

# Roomers

An expansion box for the Amiga 1000 that has a similar PC bus???  
A 68020 upgrade card for the 2000???  
and MORE!!!

*By The Bandito*

*The opinions expressed in this column are expressfully those of the author, and NOT necessarily those of PiM Publications, Inc. or it's associates.*

Since their declaration of the Amiga as an ideal desktop publishing vehicle at the fall developer conference, Commodore has been patiently waiting for desktop publishing software to appear. PageSetter from Gold Disk has appeared as a front runner in this market, with Brown-Wagh's Publisher close behind, though not yet to market.

Commodore has pounced on Gold Disk, and rumors say they will be featured in upcoming Commodore advertisements. Meanwhile, Commodore's PR agency holds a lamp high, searching for more desktop publishing software.

There are rumors of bugs and nasties with Titlecraft, the program that comes with the Genlock. Some users have reported that the program crashes in a big way when trying to load new fonts. The crash leaves the disk drives grinding away on your disks. "It sounds like a chain saw, until you hit the power," according to one unlucky user.

Early users report that both an Amiga with Sidecar and the Amiga 2000 with bridge card take a long time to boot. The boot-up sequences for the Amiga and PC sides happen in sequence, not in parallel, so booting takes twice as long as you might expect.

The Bandito also heard a nasty about the power-up sequence with the Sidecar and the Amiga. Unless they are powered on in the right order - now which one comes first? - there is a good chance the Sidecar might become a Friedcar. And there is still no straight story on why the Sidecar is not yet available in the United States.

People are abuzz about the Amiga 2000 and 500. The public announcement of the 2000 induced a nice rise in Commodore stock prices in late February. This is no doubt linked to the MS-DOS compatibility of the new machine, which somehow excites people on Wall Street, and to more news of Commodore's financial good health. The official announcement of the Amiga 2000 was in early March, worldwide. No word yet on when the Amiga 500 will be announced.

Rumors claim Commodore has started work on a 68020 upgrade card for the 2000 that would increase operational speeds four to five times, and include a megabyte of faster memory, and 68881 numeric coprocessor support. Rumors say it has a primitive memory management chip that brings the machine closer to having virtual memory abilities. A version of Unix is planned for the card. Will this forestall the development of another Amiga? Semi-officially, Commodore people are saying the next Amiga will not appear until 1988.

Other rumors say there will be at least four players in the 68020 upgrade card market for the Amiga 1000 and 2000 - Commodore, Computer Systems Associates, Ronin, and Byte-by-Byte. They hope to differentiate the market on price and performance, and because of tricks you can play with fast and cheap memory, this should be true.

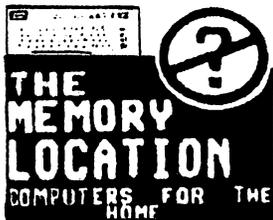
Let us hope that the new super-fast, super-expandable 68020-based Macintosh computers will inspire Commodore to push the next Amiga machine out the door. And if the Bandito hears one more person complain about their Amiga 1000 becoming an 'orphan', she/he/it is going to scream! Be thankful that the new machines are completely software compatible and no faster than your present machine. Remember, it could be worse.

The Amiga 1000 might be upgradeable to be like the 2000. Rumors circulate that an Amiga hardware company is making an expansion box for the Amiga 1000 that has a similar PC bus, as well as the Zorro and coprocessor slots in the Amiga 2000. The only thing it does not have is the video expansion slot. It probably won't be cheap, and you will still have to buy a Commodore bridge card.

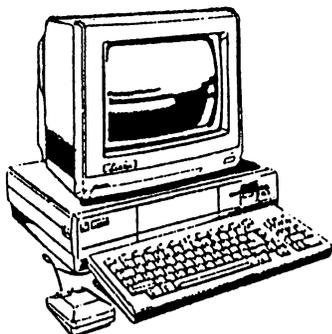
However, if you want your Amiga 1000 to be more like an Amiga 500, you can Superglue your keyboard to the front of your Amiga 1000, (after you turn it upside down, so the expansion port is on the other side.)

Sources say hardware developers are working on adapters to fit Amiga 1000 peripherals to the 500, including memory and Genlock adapters. The expandable nature of the 2000

*continued...*



396 Washington Street  
Wellesley, MA 02181  
(617) 237-6846



- AMIGA OWNERS -  
IMAGINE A STORE BUILT AROUND THE AMIGA!  
IT'S HERE NOW!!

THE MEMORY LOCATION  
396 WASHINGTON STREET (RT. 16)  
WELLESLEY, MA 02181  
617 - 237 - 6846

JUST A FEW DOORS UP FROM THE PLAYHOUSE  
FEATURING THE LATEST AND THE GREATEST FOR AMIGA  
WHAT DO WE HAVE?

A.S.D.G MINI-RACK-B TWO SLOT CARD CAGE +RAM  
SUPER HUEY JUMP DISK EXPERT SYSTEM MEAN18  
CHESSMASTER 2000 ADDISON WESLEY MANUALS  
THE PAWN CRIMSON CROWN TRANSYLVANIA TRINITY  
WINNIE THE POOH CHESSMATE CASIO KEYBOARDS  
FLIGHT SIMULATOR II INSIDE AMIGA GRAPHICS  
ADVANCED BASIC DOS REF. GUIDE TRUE BASIC  
THE EXPLORER LEATHER GODDESSES OF PHOBOS  
MONEY MENTOR ZUMA FONTS DELUXE HELP GRABBIT  
DB-MAN DATAMAT GENESIS II IMPACT GRAPHICS  
LOGISTIX LEADER BOARD DELUXE VIDEO DIABLO  
FINANCIAL PLUS INFO BASE LATTICE C ZORK I  
DELUXE PAINT MASTERTYPE MOUSTERPIECE  
FINANCIAL COOKBOOK BRATTACUS HACKER FORTH  
SEVEN CITIES OF GOLD ONE ON ONE MARAUDER  
TALKING COLORING BOOK ANALYZE! TEXTCRAFT  
AEGIS ANIMATOR ZORK II AEGIS IMAGES LISP  
MONKEY BUSINESS FORTRAN 77 SPELLBREAKER  
AZTEC C SCRIBBLE ZORK III DIGITAL LINK  
RACTOR ARCHON GISMOZ CUSTOM PRINT DRIVERS  
AMIGA DOS MANUAL (BANTAM) KID TALK BBS-PC  
TYCHON UTILITIES PAK-A-DISK MOUSE MATS  
ON-LINE AMIGA HANDBOOK (SUNSHINE) FLOW  
MOUSTERPIECE HALLEY'S PROJECT PASCAL  
GRAPHICRAFT CSI MULTI-FOURTH ARCTIC FOX  
PAR-HOME CABLES MINDSHADOW MUSIC STUDIO  
BORROWED TIME DISCOVERY AMIGA MODEM T&E  
TALKING TRIVIA DIGI-VIEW META-PASCAL  
MODULA II DEVELOPERS + COMMER. SPELLER BEE  
ELEMENTARY AMIGA BASIC BOOK INFOMINDER  
BEGINNERS GUIDE TO AMIGA AMI-PROJECT  
AMIGA CROSS DEVELOPMENT ENVIRONMENT FOR IBM  
MIND FOREVER VOYAGING BUSINESS STATISTICS  
TYPING TUTOR + WORD INVADERS VOLKSMODEM 12  
MIAMIGA LEDGER+FILE EXPERIMENTAL STATISTICS  
MAXIPLAN SALES FORCASTING VIP PRO. MIRROR  
ONE MEG RAM EXPANDER INFOMINDER FISHDISKS  
AMICUS DISKS DYNAMIC-CAD GOLDEN HAWK MIDI  
MIMETICS SOFTWARE, MIDI, DIGITIZER MAXIPLAN  
GOLDEN OLDIES OKIMATE 20 PRINTER  
AMAZING COMPUTING AMIGA WORLD TRANSACTOR  
CANON COLOR INK JET AND DRIVER  
SOFTWARE RENTAL CLUB CONSIGNMENT SALES  
AND MORE !!!

A BETTER QUESTION WOULD BE  
"WHAT DON'T WE HAVE?"  
ONLY WHAT WORKS, SATISFACTION GUARANTEED

has rekindled fires under hardware developers who found Amiga 1000 boxes unfeasible. Others are still balking at the changes in the board size change required in the 2000. Boards are cheaper and easier to produce than boxes, so more vertical-market applications will arise around the 2000.

Meanwhile, another set of rumors claimed the Amiga 500 was not going to be produced, but instead the Amiga 1000 was going to be redesigned and sold for a lower price. A modified and cheaper Amiga 1000 makes some sense, but making a still cheaper Amiga 500 makes more sense.

Developers are beta-testing a streamlined version of AmigaDOS that increases speed and performance for Amiga hard disk systems. No word on how much, or when, but at least it exists!

It is said an arcade coin-operated game company is developing games that use the Amiga chip set inside, if not an actual, revamped Amiga circuitboard. They are doing this to reduce the cost of the game machine itself, as well as the lowering the cost of developing quality game software. The machine would carry an extra amount of memory to eliminate disk access, except at power-up. All they would need to do is license the best titles from the Amiga market, and change the source code to accept quarters, right?

Speaking of great graphics, the authors of the Juggler demo is said to have signed with an Amiga company to produce a commercial version of the program that created this HAM animation. With a top-secret trick, it might produce the animation frames at near-real time speeds. No joke!

Rumors continue about a company nicknamed 'Phoenix, Inc.' This is the pipedream of several former Amiga people, hoping to form a new company after the attrition and shutdown of the Amiga office. Several former Amigans have gone off to Apple Computer, and others to Mitsumi. One rumor-monger claims there is a \$5000 bounty for the person who brings an Amiga person to the Sun company, who make the mini-like but mainframe-power computers of the same name.

•AC•

# BASIC

# Gadgets

## Create Your Own Gadget Functions!

by Bryan Catley

As I'm sure you're aware, gadgets are those little boxes containing a word or two, or some type of symbol, which appear in windows and invite you to click on them to cause some predetermined event to occur. Unfortunately, Amiga Basic contains no built-in functions for drawing or checking gadgets. However, it is a relatively easy task to draw and check your own gadgets, and that's what we're going to look at! Not only will we see how to do it, but we'll also see how to do all this from three subprograms that may be merged into any program which requires them. You will, in effect, create your own gadget functions!

### A Simple Gadget

The simplest form of gadget is just a box with a word in it. For example:

```
CLS:LINE (36,20)-(92,36),1,b:LOCATE 4,6:PRINT "Gadget"
```

From a purely functional point of view, there is nothing wrong with this gadget, but you will probably prefer one that is a little more pleasing to the eye. However, there is an important lesson to be learned from this simple gadget! Choosing the gadget coordinates!

The first thing to do is to decide on the location of the text which will appear within the gadget, and then choose the appropriate coordinates for the box itself. In the example above, the box is drawn four pixels, or half a character wider and higher than the text itself. You may, of course, draw your box any distance from the text you decide, but if it is too big it will look a little odd! The most important thing to remember is that text is, for the most part, drawn at a specific row and column location, and the box should surround the text evenly.

### A Prettier Gadget

Okay, let's see what we can do to make this gadget a little nicer to look at; something we'd like to have in our programs! How about if we make the box solid and then outline it with a contrasting color? While we're at it, let's insert an inner

outline, and place a "shadow" on the right and lower sides! Try the following program to see what it looks like:

```
LINE (36,20)-(92,36),1,bf
LINE (36,20)-(92,36),3,b
LINE (38,22)-(90,34),3,b
LINE (93,21)-(93,37),2
LINE (93,37)-(37,37),2
COLOR 3:LOCATE 4,6:PRINT "Gadget"
```

If more than one gadget is present at the same time, we can even use different colors; how about green for a "Yes" gadget, and red for a "No" gadget?

Please note that the rest of our discussion will be based on gadgets of the "prettier type" described above. Feel free to establish your own design, but it will also become necessary for you to make the appropriate changes through-out the remainder of our discussion!

### The Components of a Gadget

By now it should be pretty obvious that to draw each gadget separately, using the code shown above, would be very cumbersome and a very poor approach. This is the perfect situation in which to use a subroutine. But we're going one step further, and use a subprogram! The two big reasons for this are that subprograms are a lot more independent (very important when they will be used in many programs), and they can be used in a similar fashion to other functions, i.e. "name parameters"; and this makes the program easier to read and less complicated.

Now, what information about each gadget is our subprogram, (let's name it "DrawGdgs"), going to need? Well, for starters, it will need the coordinates of the upper left-hand corner and either the length and height in pixels, or the coordinates of the lower right-hand corner. We'll use the former, it's easier! Also, we're going to need the color of the gadget's background, the color of the border and contents, the color of the "shadow", and finally, the text that is to be inserted in the gadget. That's eight different pieces of information!

*continued...*

The most obvious way of handling this is to pass all eight pieces of information to "DrawGdgts each time we want to draw a gadget. Using our earlier example, we'd have:

```
DrawGdgts 36,20,56,16,1,3,2,"Gadget"
```

While there is nothing particularly bad about this approach, its biggest disadvantage is that each gadget must be drawn individually; we cannot specify a range of gadgets to draw.

This is not terribly important when drawing gadgets (unless you have a lot of them on the screen at once), but it does take on some significance when checking which gadget has been selected. Thus, it becomes important to be able to specify a range of gadgets to be drawn or checked - which is why we called the subprogram "DrawGdgts" rather than "DrawGdgt"!

Needless to say, given this requirement, the information about the gadgets must be stored in an array; and since we have both numeric and string data, we'll actually need two arrays. This automatically means the need to DIMension two arrays, and to provide a series of DATA statements which describe the gadgets.

This might be done as follows:

```
NumGdgts=4:DIM Gdgts (NumGdgts-1,6),GdgtTxt$(NumGdgts-1)
```

The following points are of interest:

*-A variable (NumGdgts) is used to specify the actual number of gadgets that will be used. As we add or delete gadgets to our programs, we only have to change the value of one variable rather than searching through the entire program for every reference to the number of gadgets.*

*-The "Gdgts" array is two-dimensional; one seven position entry for each gadget.*

*-Remember that arrays start at entry zero, so they are dimensioned to the maximum number less one. The associated DATA statements would look something like:*

```
DATA 36, 20, 56, 16, 1, 3, 2, "Gadget"
```

In other words, it contains the same eight pieces of information we've already been discussing! One important point must be made here, just in case it has not become obvious! Since we will be drawing and checking our gadgets in ranges, gadgets that will appear on the screen at the same time, must be grouped together in the DATA statements.

### **Saving the Gadget's Components**

All of this brings us to the first of our three subprograms, creating the arrays from the DATA statements. Let's call this subprogram "BldGdgts". Now, what information is "BldGdgts" going to need to perform its defined function of building the two gadget arrays? Well, it's going to need to know the total number of gadgets, and the names of the two arrays. We'll pass all three pieces of information as parameters; we could use the SHARED statement, but while

we'll remove the need for parameters, we would be forced into using the same names in every program, and that might not be what we really want. So, we'll invoke our "BldGdgts" subprogram with one of two statements types:

```
CALL BldGdgts (NumGdgts,Gdgts(),GdgtTxt$())
```

**OR**

```
BldGdgts NumGdgts,Gdgts(),GdgtTxt$
```

Either form is correct and the choice is strictly a personal one. Note how the arrays are specified. The () is required for Basic to know it is an array being passed and not a variable.

Look at Listing #1 and then type it in. This is our "BldGdgts" subprogram; it is a pretty straight-forward use of the READ statement to extract values from DATA statements and place them in arrays. Note that while the specified parameters must be in the same sequence, they do not need the same names, and that variable names are unique within a subprogram; i.e. "x" in a subprogram is a different variable from "x" in a main program. Since we will eventually be combining the other two subprograms with "BldGdgts", pick an appropriate name to save it as! How about "Gadgets"? And remember to save it with the ",A" option; for example:

```
SAVE "Gadgets",A
```

This will allow the code to be merged in with other programs at a later time. N.B. You must use an immediate command to do this, the Project Save menu item cannot be used, at least not for the first save.

### **Drawing the Gadgets**

All of the array information is now stored in arrays, and the time has come to use "DrawGdgts". To do its job, this subprogram needs to know the starting and ending gadgets to be drawn, and the name of the arrays containing the gadget data. Thus, we might use

```
CALL DrawGdgts (2,5,Gdgts(),GdgtTxt$())
```

to draw gadgets three through six from the arrays; (remember, the first entry in an array is entry zero). Now, "DrawGdgts" itself will set up a loop, and for each gadget to be drawn, it will:

- *Extract the necessary data from the array, (not strictly necessary, but it allows Basic to make the array calculation only once, and it also allows shorter, more usable names to be applied to the values)*
- *Draw the gadget in much the same fashion as we did earlier.*
- *Print the text in the gadget after calculating the starting row and column*

Listing #2 shows the "DrawGdgt" subprogram. It also shows something new! The inner outline, text and shadow are only drawn if the "shadow" color is greater than -1 (or is a valid palette number)! This minor extension allows u to handle another type of gadget very easily.

Up to this point we've been discussing "Yes/No" type gadgets which will "flash" when clicked on, and which are used simply to indicate the user's choices. Another type of gadget is used for data entry purposes. For example, in a requester, a user clicks in a box to indicate some data will be entered. The box reverses its outline and interior colors and all is ready for some data to be entered.

Since this type of gadget will not use a "shadow", we can use this field to indicate the type of gadget by using an illegal palette number. So, if we desire this form of gadget, we set the "shadow" color to -1 and everything will be taken care of automatically!

Back to "DrawGdgt"; if a -1 is detected as the shadow color then only the box and its outline are drawn. The inner outline and the text are ignored.

Now, LOAD "Gadgets" (or whatever you called it) and extend it by adding Listing #2 to the end, and then save it again. The ".A" should not be necessary this time, and the Project Menu save item may also be used. Once saved with the ".A", all future saves (or replaces) will automatically use the ASCII format.

### Which Gadget was Clicked?

Our third subprogram, "GetGdgt" is also our most complicated! So before looking at how it works, let's review how the mouse might be used to check for a simple gadget being clicked. Remember, MOUSE(0) returns a zero value until the left button is pressed when MOUSE(1) returns the x coordinate of the mouse pointer, and MOUSE(2) returns the y coordinate. The following sample program draws a simple gadget, and will not quit until you click in the gadget. Try it!

```

LINE (36,28)-(92,42),1,B'Draw Gadget
LOCATE 5,7:PRINT"Quit"      'Insert Text
Ok=0                        'Set Flag=0
WHILE Ok=0                  'Loop till flag not zero
  WHILE MOUSE(0)=0:WEND     'Wait for click
  x=MOUSE(1):y=MOUSE(2)     'Pick up x & y coordinates
  IF x>36 AND x<92 THEN    'Check x coordinates
    IF y>28 AND Y<42 THEN  'Check y coordinates
      Ok=1                  'Set flag=1 if in gadget
    END IF
  END IF
WEND
LOCATE 1,1:PRINT"You did it!"
END

```

The above is pretty straight forward, and once you understand it, "GetGdgt" will be that much easier to follow. To do its job, "GetGdgt" (Listing #3) needs to know the range of gadgets to be checked, the names of the gadget arrays, and to have a means of letting the caller know which gadget was clicked in. The latter may be handled with a variable that

## 5 Reasons Why You're Ready For MacroModem

1. You love telecom, but not memorization. MacroModem's user-written macro libraries and companion help screens (36 macros per file) store log on procedures, remote system menus and commands, .....
2. You've always wanted to use the mouse after you're connected, too. Write macros that mimic remote system commands and menus then execute them with the mouse or keyboard.
3. You like automation, but not script languages. Our macros use normal commands from MacroModem, remote systems, and AmigaDOS, as well as text and control codes. A multi-windowed MacroEditor is included. No new programming language to learn.
4. You want to do other things while downloading a file. MacroModem is truly multi-tasking, with a NewCLI available anytime, even during file transfers. And MacroModem's error checking won't stop downloads unless you tell it to.
5. Of course MacroModem includes standard telecom software features, too. Teach MacroModem what you want, and it will remember for you.

**MacroModem** - the better way to do telecommunications. \$69.95

Kent Engineering & Design  
P.O. Box 178, Mottville, NY 13119  
(315) 685-8237

is set to the relative number of the gadget clicked in, within the given range. This being the case,

```
CALL GetGdgt (2,5,Gdgt$,GdgtTxt$,gdgt)
```

may be used to determine which gadget, if any, has been clicked. In this example, "gdgt" will return 0-4 depending on whether the click was outside all the gadgets, or was in one of gadgets 3, 4, 5, or 6 respectively. If "gdgt" comes back as zero, we'll almost certainly want to loop back and issue the call again. The following is an example of how "GetGdgt" might actually be used in many situations:

```

gdgt
WHILE gdgt=0
  CALL GetGdgt (2,5,Gdgt$,GdgtTxt$,gdgt)
WEND
ON gdgt GOTO Rtn3,Rtn4,Rtn5,Rtn6

```

The obvious question right now is "Why can't GetGdgt be written to wait for a click in one of the specified gadgets?" The answer, for now, is "Flexibility". We'll discuss a more practical reason later!

Beside the five parameters, "GetGdgt" also shares three variables with the main program. All are conveniences for the main program; two pass back the actual x and y coordinates selected, and the third is simply an indicator that a valid gadget was, indeed, selected. The first two are

*continued...*

useful when the program needs to know where the pointer was in the gadget when it was clicked, while the third can be useful if the program is waiting on more events than just a gadget being selected.

Anyway, once "GetGdgt" is called, it waits for a click, sets variables and picks up the x and y coordinates of the pointer. It then loops through the specified range of gadgets to see if any of them were clicked. If one was, the gadget is drawn in its opposite colors (starting the "flash"), the relative number of the gadget within the specified range is calculated and set, and the loop is terminated. At this point, the subprogram waits for the left button to be released, when it redraws the gadget in its usual colors, (if the gadget has been selected and the shadow color is greater than -1 and thereby ending the flash). Before returning to the main program, the cursor is relocated to its original position. (The need for this last function is not that obvious, but I found it the hard way when trying to combine user input and gadget selection!)

The listing of "GetGdgt" may look a little complicated, but it really isn't. It's just the example we looked at earlier with a few bells and whistles thrown in!

It's now time to add "GetGdgt" to "Gadgets". Load "Gadgets" and extend it with Listing #3 and save the entire thing. You could have saved each of the three subprograms separately if you wished, but since the chance of one being used without the other is very slim, it makes more sense to combine them into one program unit.

### **Let's Try Them Out**

After all that effort we'd better make sure they work correctly! Listing #4 shows a sample program which exercises all three subprograms. Enter it as shown, and then click in the Basic Output Window and enter the immediate command: MERGE "Gadgets" (or whatever you called the subprograms).

The three subprograms should be appended to Listing #4 in your Edit Window. If you receive a "Bad File Mode" error message you probably forgot to use the ",A" option when you originally saved "BldGdgt". No problem! Just save Listing #4, load "Gadgets" again and then resave it with the ",A" option; (use the immediate command:SAVE "Gadgets",A), reload Listing #4 and then issue the MERGE again. Once you have them combined, save the entire program as, say, "GdgtDemo"; (no need to use the ",A" option now).

If you look at Listing #4, it won't be too long before you notice two things that are different from what we have discussed so far. First, there is a gadget with a shadow color of -2, and we are using a completely different method of waiting for the mouse click! The former is easy, it just gives us a third type of gadget without any need to modify the gadget subprograms. This gadget will not change at all when it is clicked in. A useful option when selecting an item from a list to obtain further information in a separate window.

The second is a little more involved. Rather than use the method of checking the mouse described above, we set up a mouse event trapping routine, (ON MOUSE GOSUB GetMouse); turn it on, (MOUSE ON); and then, when it's time to wait for the click, we go to SLEEP until the user clicks the button. At this point, control automatically passes to the "GetMouse" routine which simply invokes "GetGdgt". The RETURN results in control being passed back to the statement immediately following the one that was being executed when the "mouse event" occurred. In our case, this is the WEND following the SLEEP. Now, if "gdgt" is still zero, the program goes right back to SLEEP; if it is non-zero, it responds appropriately. In other words, we are waiting for the click, via the SLEEP statement, outside of the "GetGdgt" routine.

Why is this an important option? Well, assume you have a program which puts up a "Yes/No" requester, and then goes to "GetGdgt" and dutifully waits at the WHILE MOUSE(0) statement. Meanwhile, the user scratches his/her head and decides to use the help menu that you have thoughtfully provided, and at the bottom of the help window, lo and behold, is an "OK" gadget.

The result of this is that when the help routine tries to go to "GetGdgt", the program crashes with an error message informing you that a subprogram cannot be used by two users at the same time! It is, therefore, important to wait outside of the "GetGdgt" subprogram! SLEEP is used because it does not require any CPU cycles (unlike WHILE MOUSE(0)=0:WEND) and this is very important in a multitasking machine such as the Amiga. SLEEP is also useful when you're waiting for one of several events to occur. Just set up the event traps and go to SLEEP; there is no need to establish a loop which is constantly checking on all the events you are waiting on!

All this is not to say that polling (WHILE MOUSE(0)=0:WEND) should never be used. Event traps are just an alternate (but probably better) method, which involves a little more programming, but which offers a number of advantages.

We got side tracked, back to our "GdgtDemo" program! Once you've entered Listing #4, MERGED in the "Gadgets" subprograms, and saved it; RUN it. The screen will show a number of gadgets examples in the lower portion, and will have two "real" gadgets at the top, "More" and "Quit". If you click in any of the example gadgets, you'll receive a message telling you which one you clicked. If you select "More", you'll receive a second screen of example gadgets, with "Repeat" and "Quit" gadgets at the top. As you might expect, "Repeat" returns you to the first set of examples, and "Quit" terminates the program, as it does on the first screen.

Well, you are now an expert at creating and using your own custom gadgets from within Amiga BASIC™! Use them well and spruce up those programs!

•AC•

## Amazing Reviews...

# METASCOPE

The Debugger by  
METADIGM

*"Whatever level of debugging you're interested in, Metascope's easy-to-use interface can make life much easier."*

**Reviewed by**  
**Harriet Maybeck Tolly**

Metascope is a symbolic debugger which allows you to observe and control execution of application programs to assist in the location of bugs. Using a multi-window environment, the programmer can interactively halt program execution, and look at or change the memory that the program is occupying.

First, a brief history of debuggers on the Amiga. Developers who got kits from Commodore-Amiga received two debuggers.

One is called ROM-Wack. ROM-Wack can be entered in the few moments between when a program crashes, and when the machine reboots. It can also be entered at any time by calling the Exec function Debug(). It sends information out the serial port at 9600 baud. To use ROM-Wack you must have another machine or terminal receiving this output from your serial port. You may then look around through memory (using this other machine) to see where things might have gone awry.

The other debugger provided to developers is called Wack. Wack is a superset of ROM-Wack. You can run Wack from the CLI, or enter it by calling Debug(). Output can be directed to the screen, or to the serial port. This is useful for debugging graphics applications, as the display is not disturbed. It is a symbolic debugger much the same as Metascope is. It is very powerful and has impressive command macro capabilities. The major problem with Wack is that most developers did not receive any documentation concerning its operation until the Gamma 1 release in October 1986. If you had used a debugger before, you had a shot at guessing at a few commands, but this didn't get me very far.

Both ROM-Wack and Wack have one major shortcoming. They are command-line oriented - sort of like using 'ed' on memory. You have a prompt, and you enter in a command to get some information. There are no gadgets, no menus. Wack will allow you to open one other window for I/O, so commands you enter don't get confused with program output, but other than that there are no windows for information display.

Which brings us to Metascope. Metascope was given the 1986 Best Utility award by Bruce Webster of Byte magazine. This is a fully Intuition-ized application that lets you look at and change memory at will. All information is displayed via windows, and input is accomplished through requesters, gadgets, and mouse clicks. This Intuition interface, especially multiple windows, allows for very fast retrieval of debugging information.

Windows may display memory (as data or assembly code), symbol definitions, breakpoints you have dynamically defined, current program registers, and hunk definitions. You may open as many windows as your memory can handle. Metascope only requires that at least one window be open at any time. Once you close all windows, Metascope unloads your program and terminates.

The information in windows can always be scrolled through with the use of sliders (proportional gadgets), in the standard Intuition manner.

When you run Metascope the first window you are given is a status window. This shows the contents of your program registers, the program counter, condition code values, and the assembly instruction to which the program counter currently points. There is also a description of the current program state (ie. executing, at breakpoint, etc.).

Any of these values can be changed (excluding the state description). Altering a value in Metascope can be accomplished two ways. First, you can 'open' the location with a double click. At this point you are presented with a 'requester' containing a string gadget that lets you enter a new value. Once you enter the value(s) and hit the DONE gadget (or RETURN), the value replaces the previous value at that location.

An alternate method of changing a value is provided for registers. You first select a value, by single clicking on it. Then double click on, for example, the register name A7. The selected value will automatically replace the old value of register A7.

Metascope also allows expressions to be entered whenever a value or address is required as input. These expressions can contain symbols and register names. For example, if your program defined a symbol called ControlWindow, you could enter the expression (ControlWindow & A3) when asked for a memory value.

If you are using a high-level language such as 'C', you'll want symbols to help you navigate through your application in memory. The phrase 'symbolic debugger' refers to a debugger which makes available to you, the names of functions, variables, and any other named memory you may define. You can then refer to these names, rather than having to juggle numbers of memory locations.

Metascope lets you open a window containing all of the symbols that it was able to read from your file. In true Intuition style, you simply click on the symbol name or its memory location and it becomes selected (highlighted). Having symbol names available at a click of the mouse eliminates having to remember and type in long names. You can even use a symbol in an instruction that you submit for in line assembly (more on that later).

Metascope also allows you to define your own symbols. You select a memory value, give it a name, and then you may use it just as you would use a symbol that was read in from your file. Any value or name in the symbol window can be selected, to be used elsewhere. These values, however, can not be altered.

Unfortunately, choosing the debug switch on the Lattice or Manx C compilers currently only includes global symbols in the file. Local symbols are not available to Metascope. So you will not see variable names you have defined within functions. It is still quite possible to trace through a function. I'll give an example below.

Metascope lets you open multiple windows displaying memory contents. You can open a memory window at any location. If that location is associated with a symbol, you just click on the symbol, and choose windows/memory from the menu. The window will automatically be positioned at that symbol. You may also simply type in a number if you would like to view a memory location that is not associated with a symbol. Memory is displayed as two LONGs across. At any location which corresponds to a symbol, the symbol will be shown above the memory contents.

Memory may be displayed either as data or as code, by a simple menu choice. Memory displayed as data has a (optional) text interpretation displayed to the right. If you open a memory window at a symbol that represents a function, you would most likely want to look at the contents as code. This will be standard 68000 assembly code, and symbols will be substituted whenever possible. If your original code is in assembly language, you will be looking at the actual code. If, however, you are writing in a high level language, you will see a disassembled version of your code.

If you have compiled and linked with Lattice C, Metascope will be able to convert most addresses to symbols where applicable. However, the Manx C compiler and linker default to using an indirect addressing scheme through register A4, for access to data and branches to code. If you want symbols to be displayed by Metascope in disassembled code, you have two options. One is to compile with the +C (large code) and +D (large data) switches. This eliminates the use of A4 for addressing, but produces larger code.

Another way around the problem is to provide Metascope with knowledge of the register being used for the base of address offsetting, A4 in the case of Manx. Once the base is loaded, (i.e. you have executed to \_main), Metascope can calculate the absolute address and locate the symbol. The symbol will then be substituted into your disassembled code. You can enable and disable this feature anytime.

Memory can be displayed by absolute location, by offset or both. The offset can be Hex or decimal.

At first you'll probably spend a lot of time just looking at memory. The real power, however, comes from altering memory values. This takes a strong stomach the first time, but before long you'll find yourself changing pointers and assembly statements with ease. Memory values can be modified the same way as register values. Simply double click on the value to 'open' it. You'll be presented with a requester, into which, you enter the new value. This can be either data or actual code. When you modify a line of code in memory, the new code is reassembled in before you continue execution.

Memory can also be changed via 'editor' type commands. You can search memory for a particular value. Memory can be 'filled' with a value, or a value can be 'copied' into another location.

If you have any doubt that Metascope is updating its display as memory changes, try the following: Open a memory window at the start of where your Window structure resides. This structure has two members called MouseY and MouseX. As you move the mouse pointer around the window, you will see Metascope updating these values at lightning speed.

An important function of any debugger is the ability to set a breakpoint. What this means is that you choose a symbol, or location (frequently a function), and tell the debugger to pause execution of your application when it encounters that symbol in the code. This is accomplished by the debugger replacing your instruction with an illegal one that will be trapped by the processor to the Metascope program, which will in turn halt your application. Once you tell the debugger to continue, it places the correct instruction back into the code.

To set a breakpoint, you select a symbol name to break on, chose command/break/set from the menu, and you've got your breakpoint. If no symbol is selected, Metascope will ask you to enter the location to break on. The currently set

breakpoints can be displayed at any time by opening a breakpoint window with the menu choice open/break. The window will include the number of times the breakpoint has been triggered and program execution halted. Breakpoints can easily be cleared with the menu choice command/break/clear.

Once you have a breakpoint set, you instruct Metascope to run your program until it encounters that point. A breakpoint can be set so that the program will stop just once, or to stop each time it is encountered. You can trace through code, stopping after each instruction, including going down into functions. You can also set current level tracing, which will treat function calls as single statements. All of these choices can be easily made by clicking the mouse on menu choices, or from the keyboard with right-Amiga key combinations.

Using breakpoints to travel through memory is one way to overcome the problem of not having internal symbols. Although you can't set a breakpoint on a symbol within the function, you can break at the beginning of the function, then step through the instructions one at a time. If your code was written in assembly language, you can step through your actual code. If your code was in a high level language, you will need to step through the disassembled code that Metascope will provide. Even if you don't use assembly language regularly, you can usually figure out approximately what the code is doing. It is handy to have a 68000 programming manual to look up instructions in this case. I frequently trace through disassembled code that was written in C and find it quite easy to locate my desired position.

Metascope also allows you to open a Hunk window. Programs on the Amiga are comprised of several 'hunks'. These correspond to sections of data and code. Metascope keeps track of where your hunks are located, so scatter-loading does not cause a problem as you are wading through memory. You can also get offsets within a particular hunk. A hunk window shows you the starting address of the hunk, the size of the hunk, and the first symbol defined in the hunk. The values in the hunk window are selectable just as any memory values are. You could click on a hunk address, then double click on a register name in the status window, and the hunk address will replace the value that was in the register.

A new type of window added to V1.15 is the Frame window. This shows you the history of functions you have called to get to your current location. In addition to the symbol of the function (if it was available), the stack frame pointer and the return address are provided. These addresses can be selected and used elsewhere.

Metascope also provides many ways to manipulate the display windows. These are all available from right-Amiga key combinations as well as the menu. Windows can be frozen, so even as you continue execution, you have a window with the old values to refer back to. These windows can be 'warmed' to return to being updated whenever memory changes, or can be 'updated' to the current values, but still remain frozen.

# WELCOME TO CANADA!

\*Software Publishers  
\*Peripheral Manufacturers  
\*Hardware Developers

**Be Represented by Canadas Premier  
Distributor of Amiga support products**

**PHASE 4 DISTRIBUTORS INC.**  
HEAD OFFICE: 7157 Fisher Road South East  
(403) 252-0911 Calgary, Alberta Canada T2H 0W5

**CAVALRY-TORONTO-VANCOUVER-ST.JOHN'S**

**ATTENTION: CANADIAN DEALERS  
CALL TOLL FREE 1-800-661-8358  
FOR THE LATEST AMIGA/128 UPDATES  
or (403)-258-0844 for our Dealer BBS**

Another handy feature of Metascope is the Log. By turning on the logging function, you can create a file containing a record of the tracing you've done, dumps of memory, either as data or code, or the contents of any window. A sample log file is included at the end of this article.

Now for that age old question. Is size really important? Well, for debuggers it is. Version 1.15 occupies about 98K bytes of memory. This has never presented a problem for me, but a game writer friend of mine insists that he could never fit both Metascope and his application in memory at the same time. The new version of Metascope can access any external memory which you have added on.

What don't I like about Metascope? Very little. Here are some minor complaints.

A few frequently used menu choices have corresponding key combinations for those who don't like being forced to use the mouse. But the majority of commands available from the menu can not be accessed using the keyboard.

Visually, Metascope windows are very plain and all the same. The argument can surely be made that a development tool does not need pretty gadgets. Agreed. The problem I have is that all of the windows are color 1 on color 0. Once you have a bunch of windows open, they start

*continued...*

to all look the same. I think I am spoiled by programs such as TxEd which lets you choose the color of your window. I think that being able to color code a bunch of memory windows would be a big plus. Metascope does let you change the title of any window, but a more visual solution would be better.

The manual provided with Metascope is 35 pages long with no index. For someone who has never used a debugger, explanations will seem scant. The information is there, just not in a tutorial format; more like a quick reference.

Metascope does not provide a way to temporarily allocate memory during the debugging process. Wack does provide this feature.

Overall, I think Metascope is one of the best pieces of software I've bought for my Amiga. It performs as promised. Anyone doing serious development, who can't look at one more 'printf' statement, can benefit from Metascope. It can be used casually, just for browsing through memory. It can also be used mercilessly, changing pointers in memory and changing code on the fly. Whatever level of debugging you're interested in, Metascope's easy to use interface can make life much easier.

---

This is a sample Log file produced by Metascope. This was accomplished by selecting memory values with the mouse, then choosing menu items to output the information to the log file. My experimentation with Metascope was done with Metascope V1.15 and AmigaDOS V1.2, with the Manx Aztec C compiler, beta version V3.30.

This is an example debugging session. I purposely created an executable file, in 'C', with the following error:

```
struct Window *ControlWindow;
:
:
RefreshGadgets(&StrGadget, &ControlWindow, NULL);
```

As you can see, ControlWindow was already a pointer to a Window. The function RefreshGadgets takes the pointer to the window as the second argument, so preceding the variable name with the address specification was incorrect. The following log shows how we could find this error.

1. Load program and open symbol window.
2. Select symbol RefreshGadgets and open memory window. Display as code.
3. Set breakpoint at RefreshGadgets, execute program to that point.
4. Select symbol ControlWindow and open memory window. Value of pointer is 38C58.

5. Looking back at the code in the RefreshGadgets window, we see the statement `MOVEM.L 8(A7),A0-A2`

This tells us that the LONG value at `A7 + 8` will be moved into `A0`, and the next LONG, at `A7 + 16`, will be moved into `A1`. This should be our pointer to a window, `38C58`.

6. Select `A7` and open a memory window. At `A7 + 16` we see the value `35CE2`. We have located the error. The address of `ControlWindow` will be moved into `A1`, instead of the value of `ControlWindow`.

```

/*****/
Program loaded at address 0395E0 (__H0_org).
Execution stopped at address 0395E0 (__H0_org).

Listing of window "Symbol":
03A6E4 _CloseLibrary
03A780 _CloseWindow
035CE2 _ControlWindow
035FD8 _DOSBase
035FCE _Enable_Abort
03A726 _GetMsg
035CDE _GfxBase
035CDA _IntuitionBase

Listing of window "Memory":
 RefreshGadgets
03A798 MOVE.L A2,-(A7)
03A79A MOVEM.L 8(A7),A0-A2
03A7A0 MOVEA.L -$7D54(A4),A6
03A7A4 JSR -$DE(A6)

Continuing execution at address 0395E0 (__H0_org).
Execution stopped at address 03A798 (_RefreshGadgets).

Listing of window "Memory":
_ControlWindow
035CE2 00038C58 "...X"
_NewControlWindow
035CE6 00140014 "...."
035CEA 025D00AC ".].."

/* Author's note:
This memory window was opened at the value of A7. */
Listing of window "Memory":
03EABA 00039732 00035CAE "...2..\"
03EAC2 00035CE2 00000002 "..\...."
03EACA 0008F800 00000100 "....."
03EAD2 40000000 0003EAF0 "@....."
03EADA 00039AD8 00010000 "....."
```

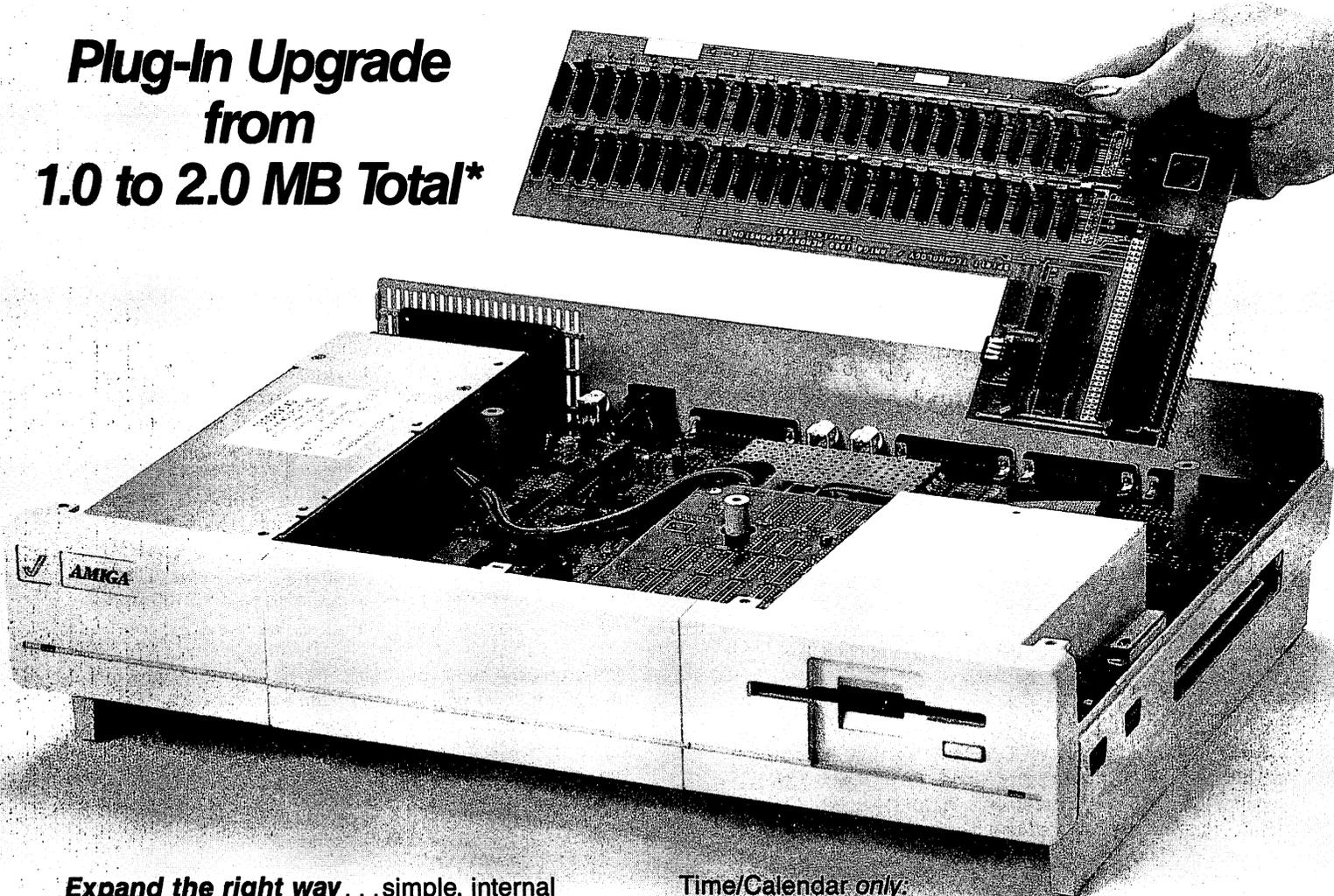
•AC•

# ✓ **AMIGA INTERNAL MEMORY EXPANSION**

## **Plus**

# **TIME/CALENDAR**

**Plug-In Upgrade**  
**from**  
**1.0 to 2.0 MB Total\***



**Expand the right way . . .** simple, internal plug-in mounting leaves your side expansion port free to add other peripherals. Also, the internal Time/Calendar does not use a joystick port.

### **Memory Expansion Features:**

- \* Zero Wait-State
- \* No Cuts or Soldering Required
- \* Full Auto-Configuration
- \* Lithium Battery Back-Up for Time/Calendar

### **ORDERING INFORMATION:**

DRAM Memory *with* Time/Calendar:

#ST-05	0.5 MB	\$349.50 List
#ST-10	1.0 MB	\$499.50 List
#ST-15	1.5 MB	\$599.50 List

\*Memory expansion from 1.0 to 2.0 MB includes AMIGA 1000 512K RAM.

AMIGA is a trademark of Commodore-Amiga, Inc.

*Time/Calendar only:*

#ST-TC	Time/Calendar including	
	Battery Back-Up	\$59.50 List

ASK ABOUT increased speed  
with the new 68010 Processor

VISA and Mastercard Welcome  
**CALL TOLL FREE: 1-800-433-7572**

Factory direct: 1-801-485-4233  
DEALER INQUIRIES INVITED

**SPIRIT™**  
**TECHNOLOGY** 

220 West 2950 South • Salt Lake City, Utah 84115

# The AMICUS Network<sup>TM</sup>

by John Foust

## *The Commodore Show in San Francisco, GENie and MORE!!!*

ONE TRUE MEASURE OF THE SUCCESS of a computer show is how you feel afterwards. Once all the events are summed together, does the whole feel as good as the parts? In this case, the answer is "yes." I left the Commodore Show pleased with the Amiga 1000 marketplace, and with a heady anticipation for the Amiga 500 and 2000.

Surprises highlight any Amiga show. There are always a few new developers who rent booth space, and always a few who could not afford a booth and instead hand out leaflets in the aisles. At the Commodore Show in Brooks Hall in San Francisco, held February 20 to 22, both types of new developers appeared.

While the fall 1986 Commodore Show had a large contingent of Commodore 64 and 128 booths, this show was dominated by the Amiga. The Amiga has moved to the forefront of the Commodore line, in public presence, even if not in dollar sales.

The Commodore Show was called the West Coast Commodore Association (WCCA) Show last fall. It is now run by Computer Swap, a computer show organizer who took over for the WCCA. This seemed to be for the good. The show organization appeared unruffled on the surface, and the WCCA officials I saw looked considerably less harried than the last show, when they were in charge.

### ***Inter/active Softworks***

Calligrapher is a font editor, but no ordinary font editor. I think Calligrapher will be the Deluxe Paint of font editors. Inter/active Softworks programmer Jeff Braun showed off the features of Calligrapher to an enthusiastic crowd. It allows kerning, stretching, italics, underline, overline, remapping of colors within a font, cookie-cutter pattern fills, drop shadows, and more. Characters can become IFF brushes, and vice versa.

While I was there, Kailash Ambwani of Gold Disk stopped by, and told me a few advantages of color fonts. They could be used to produce dithered fonts, which means characters

would have smooth transitions at sharp edges, which would prevent some of the stair-stepped effects that arise when square pixels draw continuous lines.

Inter/active Softworks has developed a standard for color fonts on the Amiga. They have been working with Commodore to make an IFF standard for this system of color fonts, so other programs will be able to use color fonts in the future. Braun showed that the color fonts can be used in Deluxe Paint, after a special program has been run.

Braun is no less than enthusiastic about his product. "It will change the face of the Amiga, because this is the way people see the Amiga - through text - and this changes text to color." Braun claims the program will be available at the end of March. The product is expected to sell for \$100, and a \$5 demo disk is available.

Please note the other coverage of the Commodore Show in the article entitled '*Highlights from the San Francisco Commodore Show*'. Amazing Computing writers Harriet Maybeck Tolly and Steve Hull were also present at the show. The show included lectures by well-known Amiga people. This article goes into depth of other booths at the show, as well.

Gold Disk promised a PostScript version of PageSetter in two weeks, which means it should be available before you read this. They primed the pump telling people about future high-end versions of PageSetter in a lecture at the show. It should have color separation abilities, including color halftoning, and improved text abilities, such as kerning and autohyphenation. It is obvious they are heading towards color desktop publishing, and hope to leapfrog the Macintosh desktop publishing software market on this point.

Infinity Software showed Galileo, an astronomy program. It presents a view of the night sky, and can highlight the constellations. It was developed by Mike Smithwick, who said he works for NASA as a "frontline grunt programmer" of flight simulators. Infinity also showed Grand Slam tennis, a game. They plan to have two productivity tools for June release, as well, but their content is still secret at this time.

## Hard disks

C Ltd. demonstrated a three foot stack of hard disks for the Amiga, from 20 to 80 megabytes of storage. They also promised a SCSI multiuser network system by midsummer. The network will work at SCSI speeds of 1.2 megabits per second. According to C Ltd.'s Ed Lippert, "The goal will be to allow any Amiga on the SCSI bus to interrogate any other Amiga's floppy disk cache. Not to write to it, just to get files from it. If it all works out, we'll allow it to write to RAM: disk, so you can leave messages for other people on the system."

So far, one port on the Amiga has escaped expansion-itis, the keyboard port. C Ltd. has developed Timesaver, a battery-backed clock that fits on the keyboard cable. When the Amiga reboots, the box sends a proper 'date' command into the keyboard buffer, setting the date automatically to the CLI window. Beyond date and time, Timesaver gives keyboard macros in a non-obtrusive way that doesn't use any Amiga memory. The box has 8 K of memory, and can store and edit key sequences at will. Because some Amiga mouse functions can be duplicated with the keyboard, and Timesaver can record keystrokes, some mouse functions can be replayed. This bizarre gadget will sell for about \$70 in a few months.

In a demonstration no doubt staged for my benefit, as well as their own, representatives from Byte-by-Byte visited the C Ltd. booth to ask probing questions about the demos running on the C Ltd. hard disks. I have no gory details to report. I do not mean to imply that things were not completely business-like and friendly between the representatives of both companies.

They questioned each other's benchmarks and demonstrations in a scientific fashion, and made an impassioned plea for an impartial, third-party hard disk comparison. At the proper moment, as if they were in complicity from the start, they all turned to me with raised eyebrows. Fortunately, I was a step ahead of them. Amazing Computing will have a discussion of Amiga hard disks in an upcoming issue, and the review process had begun before this meeting at the Commodore Show.

The event focused my attention to the problems of judging Amiga hard disks. All parties agree that benchmarks can be rigged to apparently increase performance. Hard disks can be partitioned and loaded with only a small number of files. Disks like this appear to run much faster than others, because the slowness is in the Amiga operating system, not the hard disk itself. Depending on the demo chosen, DMA can make a difference, contrary to common speculation. There are strong indications from Commodore Amiga that a version of AmigaDOS with hard disk enhancements is in the early stages of testing, and these hard disk manufacturers confirmed that they are working with it. This software alone means a doubling to quadrupling of some hard disk access.

S. Anthony Studios showed laser printer software, including LaserUpI, which prints 32 color and HAM images on PostScript printers, and LaserUtilities Volume One, an imbedded-dot-command text processor for laser printers.

For Your  
AMIGA

## Expansion Flexibility

Give your AMIGA from 512K to  
6 megabytes... an extra card slot  
and pass thru bus!

### The Advantage™...

#### A 512K/2 megabyte RAM expansion

- Auto configures under 1.1 or 1.2
- No-wait state design
- AMIGA Standard Design ("Zorro")
- User expandable with inexpensive 256x1 dynamic RAM
- RAM disk software for warm boot recovery

Cage II™... Two slot "Zorro" compatible

Cage III™... Three slot box with bus return

Call for additional information and pricing.

**(415) 651-1905**

Products shipped since August.

Prices begin at:

**\$299.**

VISA & Mastercard  
Accepted

**Pacific**   
**Peripherals**  
1080 Hiawatha Court  
Fremont, CA 94539

Central Coast Software demonstrated Disk 2 Disk, a program to read and write Commodore 64 disks. It works with Commodore 128 disks from the 1571 drive, as well as the 64's 1541 drive, using an Amiga 5 1/4 disk drive. No special hardware is needed. According to George Chamberlin of Central Coast, their software for reading and writing IBM PC disks has several advantages over the PC disk utility that came with the AmigaDOS 1.2 upgrade. Dos 2 Dos, their PC-reading program uses wildcards and can now use the internal Amiga drive as the source drive, sending files to the RAM: disk. The new Dos 2 Dos also reads and writes Atari ST disks.

Tracy McSheery of Laser Gamesmanship demonstrated his light pens. They are nearing production, he said. He is also working on 3-D glasses for the Amiga, like those available for the Atari ST. These glasses plug into the mouse port, and produce a strong three-dimensional effect with proper graphics software.

### Better than Polyscope?

Visual Aural Animation showed their Music Analysis System, a small hardware box coupled with real-time animation software. This is the same company that developed the TriClops game, whose demo is so popular on the public domain circuit.

*continued...*

The box accepts music input from a stereo or boom box, for example. The circuits measure the sound in frequency and amplitude, and send the results to the animation software. It should sell for about \$100.

Programmer Dan Egolf described this product as a way to "visualize subtleties in music. It enhances music." I would venture that this is a greater Sixties experience than the Polyscope demo. As the music plays, colorful shapes and waves of dots appear on the screen, moving very fast, almost dancing. In one setup, mountains of triangles grew and regrew as the music changed.

There are at least 52 basic patterns that can be chosen, such as mountains, flying polygons, waves of color, and circle-dancing pentagrams. These can be combined and overlaid, and their speed, color and magnitude adjusted, for thousands of variations total. The setup of patterns can be expressed as a single number, like a telephone number. If you discover an effect that you like, you can return to it by dialing its number.

Given a particular setup number, the music excites the elements of the setup, creating real-time animation. According to Egolf, "Some of them look better than others. Like a Mandelbrot, but real-time. The algorithms were written to be very fast, so they can be real-time."

Crystal Rose Software showed a fast Mandelbrot set display program called Analytic Art. It has a zoom function that will calculate the set for a region, using larger pixels to decrease the calculation time, then switch back to a finer picture when the region you want is outlined. It can also display the Julia sets, dragons, 3-D pairs, attractors, and spherical maps of an area. Programmer Rock Howard also promises a regular newsletter that will talk about the program.

### **Free GENie signup**

The GENie network had a booth, presenting their Commodore and Amiga special-interest groups. The Starship Amiga group on GENie is working on a cooperative hardware project, among the members of the network. The project hopes to produce a kit for a 20 megabyte SCSI hard disk that should cost about \$500. Starship members are coordinating circuit board design, cheap parts sources, and even artwork for the drive logo, called Excaliber.

Amiga telecom junkies should note that GENie will waive the \$18 signup fee until April 30 if you follow this procedure to sign up for GENie. Call (800) 638-8369 at 300 or 1200 baud, and enter 'HHH' when the connection is established. At the 'U#=' prompt, enter 'XJM11890,GENIE' and have your credit card handy. For voice help, call (800) 638-9636. The fee waiver means you don't get the printed user manual, but an online electronic copy is available for downloading. GENie is very low cost, you will be billed for \$5 per hour online, total.

### **User groups**

There were a half-dozen user groups present. This is always nice to see. If there is no Amiga user group in your area, consider becoming a long-distance member, and receive the

newsletter the group offers. The larger user groups produce fine newsletters well worth the small membership dues. One user group was copying public domain disks for free, an interesting sight considering another booth successfully sold the same disks for \$8 each.

Someone asked me if I felt funny about the phenomena of people making money from public domain disks because I organize the AMICUS disk set. Many user groups sell public domain disks to build a small cash reserve, and I had hoped the AMICUS disks could help groups in that way. As for businesses that spring up to sell public domain software, I view them in the same way as people who sold pet rocks. The product is available for free, but the convenience attracts others to the rock sellers.

### **Amiga 2000**

The Amiga 2000 was on the tips of many tongues at the show, no doubt inspired by a lecture on the new machine, given by a Commodore representative. At that time, the 2000 was also featured in the most recent issues of *Compute!*, *AmigaWorld* and *Byte* magazines. There is some ambivalence about the Amiga 2000 in the Amiga community.

This surprised me, at first. At worst, I imagined the 2000 as a neutral machine. While I personally cannot get excited about owning a vanilla, low speed PC clone, I know many are excited about PC compatibility. The long term success of the 2000 will be keyed to the success of expansion peripherals. Think about how long Amiga 1000 owners have waited for hard disks and memory expansion.

Consider the 2000 as having a popular (but non-Amiga) expansion bus as well as an Amiga expansion bus, and the appeal of the 2000 comes through. It just happens that the controller card for this series of cheap peripherals can also function as a standalone computer, and run PC programs in windows in the Amiga environment. Other computer owners should be so lucky.

If you like, you can ignore the PC compatibility altogether, and focus on the Amiga expansion slots. No matter how you slice it, Amiga 1000 peripherals are a pain. I seriously think many consumers decide against Amiga peripherals because installation would mean taking a saw, hammer and chisel to their wonderful wood computer desk. On my desk, I do not have room for Sidecar, PAL Jr., or even another memory card, pass-through or not.

Although the 2000 cards are a different size than the original Amiga Zorro spec, Amiga hardware developers have known about this form factor change since the developer conference this fall. While it caused some consternation and gnashing of teeth, developers have come around to support it out of economic necessity, if not out of friendly support for the new card standard. All the weeping and wailing has taken place over the past six months, and hardware developers have returned to their cash registers.

In the PC community, the appearance of the 2000 is rekindling interest in the Amiga, and Commodore can only benefit from being the subject of optimistic conversation.

## Companies mentioned

**Aegle Development**  
2115 Pico Boulevard  
Santa Monica, CA 90405

**Access Associates**  
491 Aldo Avenue  
Santa Clara, CA 95054-2303  
(408) 727-8520

**Brown-Wagh Publishing**  
16795 Lark Avenue, Suite 210  
Los Gatos, CA 95030  
(800) 451-0900  
(408) 395-3838 (California)

**Byte By Byte**  
992 Capital of Texas Highway N.  
Suite 150  
Austin, TX 78759  
(512) 343-4357

**C Ltd.**  
723 East Skinner  
Wichita, KS 67211  
(316) 267-6321

**Central Coast Software**  
268 Bowie Drive  
Los Osos, CA 93402  
(805) 528-4906

**Computer Swap, Inc.**  
PO Box 18906  
San Jose, CA 95158  
(408) 978-SWAP  
(800) 252-SWAP (in CA)

**Delta Research**  
4054 Wilkie Way  
Palo Alto, CA 94306  
(415) 856-3669

**Ellas Engineering**  
4614 Morris Court East  
Santa Rosa, CA 95405

**GErie**  
**GE Customer Services**  
Department 02B  
401 North Washington Street  
Rockville, MD 20850  
(800) 638-9636

**Gold Disk**  
2179 Dunwin Drive #6  
Mississauga, Ontario  
Canada L5L 1X2  
(416) 828-0913

**In-Vitro Design**  
5638 Allen Avenue #3  
San Jose, CA 95123  
(408) 226-3649

**Infinity Software**  
1331 61st Street Suite F  
Emeryville, CA 94608  
(415) 420-1551

**Interactive Softworks**  
57 Post Street, Suite 811  
San Francisco, CA 94104  
(415) 956-2660

**JDK Images**  
2224 East 86 Street  
Suite 14  
Bloomington, MN 55420  
(612) 854-7793

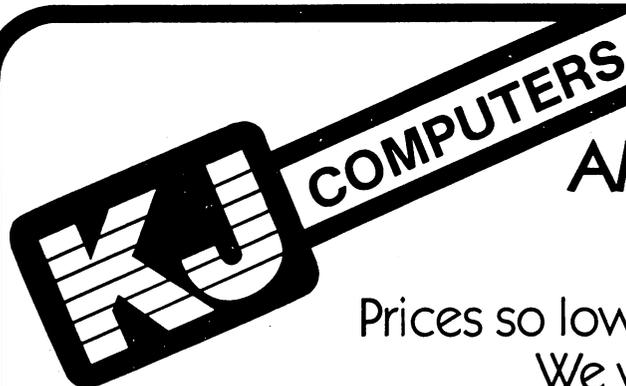
**Laser Gamesmanship**  
PO Box 1223  
Alta Loma, CA 91701  
(415) 891-9968

**Manx Software Systems**  
PO Box 55  
Shrewsbury, NJ 07701  
(201) 542-2121

**S. Anthony Studios**  
889 DeHaro Street  
San Francisco, CA 94107  
(415) 826-6193

**Visual Aural Animation**  
PO Box 4898  
Arcata, CA 95521  
(707) 822-4800

**WordPerfect Corp**  
288 W. Center Street  
Orem, UT 84057  
(801) 225-5000



## AMIGA™ & COMMODORE™ PRODUCTS

Prices so low we will not advertise them...  
We will not be undersold!

Inside CA 1-818/366-5305 • Outside CA 1-800/443-9959

KJ Computers, quite possibly the largest Amiga dealer in the USA, stocks all Amiga and third party Amiga products, as well as most popular peripherals and supplies. KJ is easy to do business with, their staff is knowledgeable, and delivery fast. For all this, and best pricing available, give KJ Computers a call today!



10815 Zelzah Avenue, Granada Hills, California 91344

# The AMICUS & Fred Fish Public Domain Software Library

This software is collected from user groups and electronic bulletin boards around the nation. Each Amicus disk is nearly full, and is fully accessible from the Workbench. If source code is provided for any program, then the executable version is also present. This means that you don't need the C compiler to run these programs. An exception is granted for those programs only of use to people who own a C compiler.

The Fred Fish disk are collected by Mr. Fred Fish, a good and active friend of the Amiga.

Note: Each description line below may include something like 'S-O-E-D', which stands for 'source, object file, executable and documentation'. Any combination of these letters indicates what forms of the program are present. Basic programs are presented entirely in source code format.

## AMICUS Disk 1

### ABasic programs: Graphics

3DSolids 3d solids modeling program w/sample data files  
Blocks draws blocks  
Cubes draws cubes  
Durer draws pictures in the style of Durer  
FScape draws fractal landscapes  
Hidden 3D drawing program, w/ hidden line removal  
JPad simple paint program  
Optical draw several optical illusions  
PaintBox simple paint program  
Shuttle draws the Shuttle in 3d wireframe  
SpaceArt graphics demo  
Speaker speech utility  
Sphere draws spheres  
Spiral draws color spirals  
ThreeDee 3d function plots  
Topography artificial topography  
Wheels draws circle graphics  
Xenos draws fractal planet landscapes

### ABasic programs: Tools

AddressBook simple database program for addresses  
CardFile simple card file database program  
Demo multwindow demo  
KeyCodes shows keycodes for a key you press  
Menu run many ABasic programs from a menu  
MoreColors way to get more colors on the screen at once, using aliasing  
shapes simple color shape designer Speakt speech and narrator demo

### ABasic programs: Games

BrickOut classic computer brick wall game  
Othello also known as 'go'  
Saucer simple shoot-em-up game  
Spelling simple talking spelling game  
ToyBox selectable graphics demo

### ABasic programs: Sounds

Entertainer plays that tune  
HAL9000 pretends it's a real computer  
Police simple police siren sound  
SugarPlum plays "The Dance of the Sugarplum Fairies"

### C programs:

ATerm simple terminal program, S-E  
cc aid to compiling with Lattice C  
decvnt opposite of CONVERT for cross developers  
Dotty source code to the 'dotty window demo'  
echox unix-style filename expansion, partial S, O-D  
fasterfp explains use of fast-floating point math  
FixDate fixes future dates on all files on a disk, S-E  
freedraw simple Workbench drawing program, S-E  
GfxMem graphic memory usage indicator, S-E  
Grep searches for a given string in a file, with documentation  
ham shows off the hold-and-modify method of color generation  
IBM2Amiga fast parallel cable transfers between an IBM and an Amiga  
Mandel Mandelbrot set program, S-E  
moire patterned graphic demo, S-E  
objfx makes Lattice C object file symbols visible to Wack, S-E  
quick quick sort strings routine  
raw example sample window I/O  
setlace turns on interlace mode, S-E

sparks qix-type graphic demo, S-E  
**Other executable programs:**  
SpeechToy speech demonstration  
WhichFont displays all available fonts  
**Texts:**  
68020 describes 68020 speedup board from CSA  
Aliases explains uses of the ASSIGN command  
Bugs known bug list in Lattice C 3.02  
CLICard reference card for AmigaDOS CLI  
CLICommands guide to using the CLI  
Commands shorter guide to AmigaDOS CLI commands  
EdCommands guide to the ED editor  
FileNames AmigaDOS filename wildcard conventions  
HalfBright explains rare graphics chips that can do more colors  
ModemPins description of the serial port pinout  
RAMdisks tips on setting up your RAM: disk  
ROMWack tips on using ROMWack  
Sounds explanation of the Instrument demo sound file format  
Speed refutation of the Amiga's CPU and custom chip speed tips on using Wack  
WackCmds

## AMICUS Disk 2

### C programs:

alb AmigaDOS object library manager, S-E  
ar text file archive program, S-E  
frobj auto-chops executable files  
shell simple CLI shell, S-E  
sq, usq file compression programs, S-E  
YachtC a familiar game, S-E  
Make a simple 'make' programming utility, S-E  
Emacs an early version of the Amiga text editor, S-E-D

### Assembler programs:

bsearch.asm binary search code  
qsort.asm Unix compatible qsort() function, source and C test program  
setjmp.asm setjmp() code for Lattice 3.02  
SVprint Unix system V compatible printf()  
trees.o Unix compatible tree() function, O-D  
(This disk formerly had IFF specification files and examples. Since this spec is constantly updated, the IFF spec files have been moved to their own disk in the AMICUS collection. They are not here.)

### John Draper Amiga Tutorials:

Animate describes animation algorithms  
Gadgets tutorial on gadgets  
Menus learn about Intuition menus

## AMICUS Disk 3

### C programs:

Xref a C cross-reference gen., S-E  
6bitcolor extra-half-bright chip gfx demo, S-E  
Chop truncate (chop) files down to size, S-E  
Cleanup removes strange characters from text files  
CR2LF converts carriage returns to line feeds in Amiga files, S-E  
Error adds compile errors to a C file, S  
Hello window ex. from the RKM, S  
Kermit generic Kermit implementation, flakey, no terminal mode, S-E  
Scales sound demo plays scales, S-E  
SkewB Rubik cube demo in hi-res colors, S-E  
**AmigaBasicProgs(dbr)**  
Automata cellular automata simulation  
CrazyEights card game

Graph function graphing programs  
WitchingHour a game  
**ABasiC programs:**  
Casino games of poker, blackjack, dice, and craps  
Gomoku also known as 'othello'  
Sabotage sort of an adventure game  
**Executable programs:**  
Disassem a 68000 disassembler, E-D  
DpSlide shows a given set of IFF pictures, E-D  
Arrange a text formatting program, E-D  
**Assembler programs:**  
Argoterm a terminal program with speech and Xmodern, S-E

## AMICUS Disk 4 Files from the original Amiga Technical BBS

Note that some of these files are old, and refer to older versions of the operating system. These files came from the Sun system that served as Amiga technical support HQ for most of 1985. These files do not carry a warranty, and are for educational purposes only. Of course, that's not to say they don't work.

Complete and nearly up-to-date C source to 'image.ed', an early version of the Icon Editor. This is a little flaky, but compiles and runs.

An Intuition demo, in full C source, including files: demomenu.c, demomenu2.c, demoreq.c, getascii.c, idemo.c, idemo.guide, idemo.make, idemoall.h, nodos.c, and twrite.c

addmem.c add external memory to the system  
bobtest.c example of BOB use  
consoleIO.c console IO example  
creaport.c create and delete ports  
creastd.c create standard I/O requests  
creatask.c creating task examples  
diskio.c example of track read and write  
dotty.c source to the 'dotty window' demo  
dualplay.c dual playfield example  
flood.c flood fill example  
freemap.c old version of 'freemap'  
gettools.c tools for VSprites and BOBs  
gfxmem.c graphic memory usage indicator  
hello.c window example from RKM  
inputdev.c adding an input handler to the input stream  
joystik.c reading the joystick  
keybd.c direct keyboard reading  
layers.c layers examples  
mousport.c test mouse port  
ownlib.c example of making your own library with Lattice  
paratest.c tests parallel port commands  
seritest.c tests serial port commands  
serisamp.c example of serial port use  
prinintr.c sample printer interface code  
prtbase.h printer device definitions  
regintes.c region test program  
setlace.c source to interlace on/off program  
setparallel.c set the attributes of the parallel port  
SetSerial.c set the attributes (parity, data bits) of the serial port  
singplay.c single playfield example  
speechtoy.c source to narrator and phonetics demo  
timedely.c simple timer demo  
timer.c exec support timer functions  
timstuf.c more exec support timer functions  
WhichFont.c loads and displays all available system fonts

process.l and prtbase.l assembler include files:  
 autorqstr.txt warnings of deadlocks with autorequesters  
 consoleIO.txt copy of the RKM console I/O chapter  
 diskfont.txt warning of disk font loading bug  
 fullfunc.txt list of #defines, macros, functions  
 inputdev.txt preliminary copy of the input device chapter

License information on Workbench distribution license  
 printer pre-release copy of the chapter on printer drivers,  
 from RKM 1.1 v11fd.txt 'diff' of .fd file changes from  
 version 1.0 to 1.1 v28v1.diff 'diff' of include file changes  
 from version 28 to 1.0

#### AMICUS Disk 5 Files from the Amiga Link / Amiga Information Network

Note that some of these files are old, and refer to older versions of the operating system. These files are from Amiga Link. For a time, Commodore supported Amiga Link, aka AIN, for online developer technical support. It was only up and running for several weeks. These files do not carry a warranty, and are for educational purposes only. Of course, that's not to say they don't work.

A demo of Intuition menus called 'menudemo', in C source

whereis.c find a file searching all subdirectories  
 bobtest.c BOB programming example  
 sweep.c sound synthesis example

#### Assembler files:

mydev.asm sample device driver  
 mylib.asm sample library example  
 mylib.l

mydev.l

asmstyp.l

macros.l assembler include files

#### Texts:

amigatricks tips on CLI commands  
 extdisk external disk specification  
 gameport game port spec  
 parallel parallel port spec  
 serial serial port spec  
 v1.1.update list of new features in version 1.1  
 v1.1h.txt 'diff' of include file changes from version 1.0 to 1.1

Files for building your own printer drivers, including dospecial.c, epsndata.c, init.asm, printer.c, printer.link, printtag.asm, render.c, and wait.asm. This disk does contain a number of files describing the IFF specification. These are not the latest and greatest files, but remain here for historical purposes. They include text files and C source examples. The latest IFF spec is elsewhere in this library.

#### AMICUS Disk 6 IFF Pictures

This disk includes the DPSlide program, which can view a given series of IFF pictures, and the 'showpic' program, which can view each file at the click of an icon, and the 'saveibm' program, to turn any screen into an IFF picture. The pictures include a screen from ArticFox, a Degas dancer, the guys at Electronic Arts, a gorilla, horses, King Tut, a lighthouse, a screen from Marble Madness, the Bugs Bunny Martian, a still from an old movie, the Dire Straits moving company, a screen from Pinball Construction Set, a TV newscaster, the PaintCan, a world map, a Porsche, a shuttle mission patch, a tyrannosaurus rex, a planet view, a VISA card, and a ten-speed.

#### AMICUS Disk 7 DigiView HAM demo picture disk

This disk has pictures from the DigiView hold-and-modify video digitizer. It includes the ladies with pencils and killypops, the young girl, the bulldozer, the horse and buggy, the Byte cover, the dictionary page, the robot and Robert. This includes a program to view each picture separately, and all together as separate, slidable screens.

#### AMICUS Disk 8

##### C programs:

Browse view text files on a disk, using menus S-E-D  
 Crunch removes comments and white space from C files, S-E  
 IconExec EXECUTE a series of commands from Workbench S-E  
 PDScreen Dump dumps Rastport of highest screen to printer  
 SetAlternate sets a second image for an icon, when clicked once S-E  
 SetWindow makes windows for a CLI program to run under Workbench S-E  
 SmallClock a small digital clock that sits in a window menu bar  
 Scrimper the screen printer in the fourth Amazing Computing, S-E

#### Amiga Basic Programs:

(Note: Many of these programs are present on AMICUS Disk 1. Several of these were converted to Amiga Basic, and are included here.)

AddressBook a simple address book database  
 Ball draws a ball  
 Cload program to convert Compuserve hex files to binary, S-D  
 Clue the game, Intuition driven  
 ColorArt art drawing program  
 DeluxeDraw the drawing program in the 3rd issue of Amazing Computing, S-D  
 Eliza conversational computer psychologist  
 Othello the game, as known as 'go'  
 RatMaze 3D ratmaze game  
 ROR boggling graphics demo  
 Shuttle draws 3D pictures of the space shuttle  
 Spelling simple spelling program  
 YoYo wierd zero-gravity yo-yo demo, tracks yo-yo to the mouse

#### Executable programs:

3Dcube Modula-2 demo of a rotating cube  
 Alticon sets a second icon image, displayed when the icon is clicked  
 AmigaSpell a slow but simple spelling checker, E-D  
 arc the ARC file compression program, must-have for telecom, E-D  
 Bertrand graphics demo  
 disksalvage a program to rescue trashed disks, E-D  
 KwikCopy a quick but nasty disk copy program: ignores errors, E-D  
 LibDir lists hunks in an object file E-D  
 SaveLBM saves any screen as an IFF picture E-D ??  
 ScreenDump shareware screen dump program, E only  
 StarTerm version 2.0, term program, Xmodem E-D

#### Texts:

LatticeMain tips on fixing \_main.c in Lattice  
 GDDiskDrive make your own 5 1/4 drive  
 GuruMed explains the Guru numbers  
 Lat3.03bugs bug list of Lattice C version 3.03  
 MForgeRev user's view of the MicroForge hard drive  
 PrintSpooler EXECUTE-based print spooling program

#### .BMAP files:

These are the necessary links between Amiga Basic and the system libraries. To take advantage of the Amiga's capabilities in Basic, you need these files. BMAPs are included for 'clist', 'console', 'diskfont', 'exec', 'icon', 'intuition', 'layers', 'mathfp', 'mathleedoubas', 'mathleesingbas', 'mathtrans', 'potgo', 'timer' and 'translator'.

#### AMICUS Disk 9

##### Amiga Basic Programs:

FlightSim simple flight simulator program  
 HuePalette explains Hue, Saturation, and Intensity ex. of doing requesters from Amiga Basic  
 Requester demonstrates scrolling capabilities  
 ScrollDemo sound program  
 Synthesizer draws a map of the world  
 WorldMap

##### Executable programs:

Boing! latest Boing! demo, with selectable speed, E  
 Brush2C converts an IFF brush to C data instructions, initialization code, E  
 Brush2Icon converts IFF brush to an icon, E  
 Dazzle graphics demo, tracks to mouse, E  
 DecGEL assembler program for stopping 68010 errors, S-E-D  
 Klock menu-bar clock and date display, E  
 life the game of life, E  
 TimeSet Intuition-based way to set the time and date,  
 EMEmacs another Emacs, more oriented to word processing, S-E-D  
 MyCLI a CLI shell, works without the Workbench, S-E-D

#### Texts:

FncntrKeys explains how to read function keys from Amiga Basic  
 HackerShn explains how to win the game 'hacker'  
 lst68010 guide to installing a 68010 in your Amiga  
 PrinterTip tips on sending escape sequences to your printer  
 StartupTip tips on setting up your startup-sequence file  
 XfrrmReview list of programs that work with the Transformer

#### Printer Drivers:

Printer drivers for the Canon PJ-1080A, the C Itoh Prowriter, an improved Epson driver that eliminates streaking, the Epson LQ-800, the Gemini Star-10, the NEC 8025A, the Okidata ML-92, the Panasonic KX-P10xx family, and the Smith-Corona D300, with a document describing the installation process.

#### AMICUS Disk 10 Instrument sound demos

This is an icon-driven demo, circulated to many dealers. It includes the sounds of an acoustic guitar, an alarm, a banjo, a bass guitar, a boink, a callopo, a car horn, claves, water drip, electric guitar, a flute, a harp arpeggio, a kickdrum, a marimba, a organ minor chord, people talking, pigs, a pipe organ, a Rhodes piano, a saxophone, a sitar, a snare drum, a steel drum, bells, a vibraphone, a violin, a wailing guitar, a horse whinny, and a whistle.

#### AMICUS Disk 11

##### C programs

dirutil Intuition-based, CLI replacement file manager, S-E  
 cpri shows and adjusts priority of CLI processes, S-E  
 ps shows info about CLI processes, S-E  
 vidtex displays Compuserve RLE pictures, S-E  
 AmigaBasic programs  
 pointered pointer and sprite editor program  
 optimize optimization ex ample from AC article  
 calendar large, animated calendar, diary and date book program  
 amortize loan amortizations  
 brushtoBOB converts small IFF brushes to AmigaBasic BOB OBJECTS  
 grids draw and play waveforms  
 hilbert draws Hilbert curves  
 madlib mad lib story generator  
 mailtalk talking mailing list program  
 meadows3D 3D graphics program, from Amazing Computing™ article  
 mousetrack mouse tracking example in hires mode  
 slot slot machine game  
 tictactoe the game  
 switch pachinko-like game  
 weird makes strange sounds

##### Executable programs

cp unix-like copy command, E  
 cls screen clear, S-E  
 diff unix-like stream editor uses 'diff' output to fix files  
 pm chart recorder performances indicator  
 Assembler programs  
 cls screen clear and CLI arguments example

##### Modula-2

trais moving-worm graphics demo  
 caseconvert converts Modula-2 keywords to uppercase  
 Forth Breshehan circle algorithm example  
 Analyze 12 templates for the spreadsheet AnalyzeI

There are four programs here that read Commodore 64 picture files. They can translate Koala Pad, Doodle, Print Shop and News Room graphics to IFF format. Of course, getting the files from your C-64 to your Amiga is the hard part.

#### AMICUS Disk 12

##### Executable programs

blink 'alink' compatible linker, but faster, E-D  
 clean spins the disk for use with disk cleaners, E-D  
 epsonset sends Epson settings to PAR: from menu, E-D  
 showbig view hi-res pictures in low-res superbitmap, E-D  
 speaktime tell the time, E-D  
 undelete undeletes a file, E-D  
 cnvaplthm converts Apple ][ low, medium and high res pictures to IFF, E-D  
 menued menu editor produces C code for menus, E-D  
 quick quick disk-to-disk nubble copier, E-D  
 quickEA copies Electronic Arts disks, removes protection, E-D  
 txd 1.3 demo of text editor from Microsmiths, E-D  
 C programs  
 spin3 rotating blocks graphics demo, S-E-D  
 popcli start a new CLI at the press of a button, Sidekick, S-E-D  
 like  
 vsprite VSprite example code from Commodore, S-E-D  
 AmigaBBS Amiga Basic bulletin board program, S-D

**Assembler programs**  
 star10 makes star fields like Star Trek Intro, S-E-D  
 Pictures  
 Mount Mandelbrot  
 3D view of Mandelbrot set  
 Star Destroyer hi-res Star Wars starship  
 Robot robot arm grabbing a cylinder  
 Texts  
 vendors list of Amiga vendors, names, addresses  
 cardco fixes to early Cardco memory boards  
 cinclude cross-reference to C include files, who  
 includes what  
 mindwalker clues to playing the game well  
 slideshow make your own slideshows from the  
 Kaleidoscope disk

#### AMICUS Disk 13

Amiga Basic programs  
 Routines from Carolyn Scheppner of CBM Tech Support, to read and display IFF pictures from Amiga Basic. With documentation. Also included is a program to do screen prints in Amiga Basic, and the newest BMAP files, with a corrected ConvertFD program. With example pictures, and the SaveLBM screen capture program.

Routines to load and play FutureSound and IFF sound files from Amiga Basic, by John Foust for Applied Visions. With documentation and C and assembler source for writing your own libraries, and interfacing C to assembler in libraries. With example sound.

#### Executable programs

gravity Sci Amer Jan 86 gravitation graphic simulation, S-E-D

#### Texts

MIDI make your own MIDI instrument interface, with documentation and a hi-res schematic picture.

#### AMICUS Disk 14

Several programs from Amazing Computing Issues:

#### Tools

Dan Kary's C structure index program, S-E-D

#### Amiga Basic programs

BMAP Reader by Tim Jones  
 IFFBrush2BOB by Mike Swinger  
 AutoRequester example  
 DOSHelper Windowed help system for CLI commands, S-E-D  
 PETrans translates PET ASCII files to ASCII files, S-E-D  
 C Squared Graphics program from Scientific American, Sept 86, S-E-D  
 crif adds or removes carriage returns from files, S-E-D  
 dpdecode decrypts Deluxe Paint, removes copy protection, E-D  
 queryWB asks Yes or No from the user, returns exit code, S-E  
 vc VisiCalc type spreadsheet, no mouse control, E-D  
 view views text files with window and slider gadget, E-D

Oing, Sproing, yaBoing, Zoing are sprite-based Boing! style demos, S-E-D

CLIClock, sClock, wClock are window border clocks, S-E-D

#### Texts

An article on long-persistence phosphor monitors, tips on making brushes of odd shapes in Deluxe Paint, and recommendations on icon interfaces from Commodore-Amiga.

#### AMICUS 15

##### The C programs include:

'pr' a file printing utility, which can print files in the background, and with line numbers and control character filtering.  
 'mri' displays a chart of the blocks allocated on a disk.  
 'Ask' questions an 'execute' file, returns an error code to control the execution in that batch file an enhanced version of AmigaDOS 'status' command.  
 'Dissolve' random-dot dissolve demo displays IFF picture slowly, dot by dot, in a random fashion.  
 'PopCLI2' invoke new CLI window at the press of a key.  
**The executable programs include:**  
 'Form' file formatting program through the printer driver to select print styles  
 'DiskCat' catalogs disks, maintains, sorts, merges lists of disk files  
 'PSound' SunRize Industries' sampled sound editor & recorder

'Iconmaker' makes icons for most programs  
 'Fractals' draws great fractal seascapes and mountainscapes.  
 '3D Breakout' 3D glasses, create breakout in a new dimension  
 'AmigaMonitor' displays lists of open files, memory use, tasks, devices and ports in use.  
 'Cosmoroids' version of the 'asteroids' for the Amiga.  
 'Sizzlers' high resolution graphics demo written in Modula 2.  
**Texts:**  
 'ansi.txt' explains escape sequences the CON: device responds to.  
 'FKey' Includes template for making paper to sit in the tray at the top of the Amiga keyboard.  
 'Spawn' programmer's document from Commodore Amiga, describes ways to use the Amiga's multitasking capabilities in your own programs.

#### AmigaBasic programs:

'Grids' draw sound waveforms, and hear them played.  
 'Light' a version of the Tron light-cycle video game.  
 'MigaSof' a game of solitaire.  
 'Stats' program to calculate batting averages  
 'Money' try to grab all the bags of money that you can."

AMICUS 15 also includes two beautiful IFF pictures, of the enemy walkers from the Ice planet in Star Wars, and a picture of a cheetah.

#### AMICUS 16

'juggler' demo by Eric Graham, a robot juggler bouncing three mirrored balls, with sound effects. Twenty-four frames of HAM animation are flipped quickly to produce this image. You control the speed of the juggling. The author's documentation hints that this program might someday be available as a product.

#### IFF pictures

parodies of the covers of Amiga World and Amazing Computing magazines.

#### C programs:

'InputHandler' example of making an input handler.  
 'FileZap3' binary file editing program  
 'ShowPrint' displays IFF picture, and prints it.  
 'Gen' program indexes and retrieves C structures and variables declared in the Amiga include file system.

#### Executable Programs:

'FixHunk2' repairs an executable program file for expanded memory  
 'ms2smus' converts Music Studio files to IFF standard 'SMUS' format. I have heard this program might have a few bugs, especially in regards to very long songs, but it works in most cases.  
 'Missile' Amiga version of the 'Missile Command' video game.

This disk also contains several files of scenarios for Amiga Flight Simulator II. By putting one of these seven files on a blank disk, and inserting it in the drive after performing a special command in this game, a number of interesting locations are preset into the Flight Simulator program. For example, one scenario places your plane on Alcatraz, while another puts you in Central Park.

### Fred Fish Public Domain Software

#### Fred Fish Disk 1:

amigademo Graphical benchmark for comparing amigas.  
 amigaterm simple communications program with Xmodem  
 balls simulation of the "kinetic things" with balls on strings  
 colorful Shows off use of hold-and-modify mode.  
 dhrystone Dhrystone benchmark program.  
 dotty Source to the "dotty window" demo on the Workbench disk.  
 freedraw A small "paint" type program with lines, boxes, etc.  
 gad John Draper's Gadget tutorial program  
 gtxmem Graphical memory usage display program  
 halfbrite demonstrates "Extra-Half-Brite" mode, if you have it  
 hello simple window demo

latfip accessing the Motorola Fast Floating Point library from C  
 palette Sample program for designing color palettes.  
 trackdisk Demonstrates use of the trackdisk driver.  
 requester John Draper's requester tutorial and example program.  
 speech Sample speech demo program. Stripped down "speechtoy".  
 speechtoy Another speech demo program.

#### Fred Fish Disk 2:

alib Object module librarian.  
 cc Unix-like frontend for Lattice C compiler.  
 dbug Macro based C debugging package.  
 Machine Independent.  
 make Subset of Unix make command.  
 make2 Another make subset command.  
 microemacs Small version of emacs editor, with macros, no extensions  
 portar Portable file archiver.  
 xrf DECUS C cross reference utility.

#### Fred Fish Disk 3:

gothic Gothic font banner printer.  
 roff A "roff" type text formatter.  
 ff A very fast text formatter  
 forth A highly portable forth implementation. Lots of goodies.  
 xllap Xllap 1.4, not working correctly.

#### Fred Fish Disk 4:

banner Prints horizontal banner  
 bgrep A Boyer-Moore grep-like utility  
 bison GNU Unix replacement 'yacc', not working.  
 bm Another Boyer-Moore grep-like utility  
 grep DECUS grep  
 kermit simple portable Kermit with no connect mode.  
 MyCLI Replacement CLI for the Amiga. Version 1.0  
 mandel A Mandelbrot set program, by Robert French and RJ Mical

#### Fred Fish Disk 5:

oons Console device demo program with supporting macro routines.  
 freemap Creates a visual diagram of free memory  
 input.dev sample input handler, traps key or mouse events  
 joystick Shows how to set up the gameport device as a joystick.  
 keyboard demonstrates direct communications with the keyboard.  
 layers Shows use of the layers library  
 mandelbrot IFF Mandelbrot program  
 mouse hooks up mouse to right joystick port  
 one.window console window demo  
 parallel Demonstrates access to the parallel port.  
 printer opening and using the printer, does a screen dump, not working  
 print.support Printer support routines, not working.  
 proctest sample process creation code, not working  
 region demos split drawing regions  
 samplefont sample font with info on creating your own  
 serial Demos the serial port  
 singlePlayfield Creates 320 x 200 playfield  
 speechtoy latest version of cute speech demo  
 speech.demo simplified version of speechtoy, with IO requests

#### text.demo

displays available fonts  
 timer demos timer.device use  
 trackdisk demos trakodisk driver

#### Fred Fish Disk 6:

compress like Unix compress, a file squeezer  
 dadc analog clock impersonator  
 microemacs upgraded version of microemacs from disk 2  
 mult removes multiple occurring lines in files  
 scales demos using sound and audio functions  
 setparallel Allows changing parallel port parameters  
 setserial Allows changing serial port parameters.  
 sortc quicksort based sort program, in C  
 stripc Strips comments and extra whitespace from C source

#### Fred Fish Disk 7:

This disk contains the executables of the game Hack, version 1.0.1.

#### Fred Fish Disk 8:

This disk contains the C source to Hack on disk 7.

#### Fred Fish Disk 9:

molre Draws molre patterns in black and white  
 MVP-FORTH Mountain View Press Forth, version 1.00.03A. A shareware version of FORTH from Fantasia Systems.



<b>Fred Fish Disk 32</b>		Vttest	VT-100 emulation test program. Requires a Unix system.	Lines	Line drawing demo program
Address	Extended address book written in AmigaBasic.	<b>Fred Fish Disk 36</b>		SetFont	Changes font used in a CLI window
Calendar	Calendar/diary program written in AmigaBasic.	Acp	Unix-like 'cp' copy program	Vt100	Version 2.3 of the VT-100 terminal program.
DosPlus1	First volume of CLI oriented tools for developers.	Clock	Updated version of clock on disk 15.	<b>FF 42</b>	
DosPlus2	Second volume of CLI oriented tools for developers.	Csh	Manx 'csh'-like CLI, history, variables, etc.	This disk contains an Amiga version of MicroGNUEmacs.	
Executables only:		DietAid	Diet planning aid organizes recipes, calories	<b>FF 43</b>	
MacView	Views MacPaint pictures in Amiga low or high res, no sample pictures, by Scott Evernden.	Echo	Improved 'echo' command with color, cursor addressing	BasicBoing	AmigaBasic program demos page flipping of a 3D cube
Puzzle	Simulation of puzzle with moving square tiles.	FixHunk	Fixes programs to let them run in external memory.	Bbm	Demo copy of B.E.S.T. Business Management System.
ShowHAM	View HAM pictures from CLI.	Fm	Maps the sectors a file uses on the disk.	BbsList	A list of Amiga Bulletin Board Systems
Solitaire	ABASIC games of Canfield and Klondike, from David Addison.	KickBench	Docs, program to make a single disk that works like a Kickstart and Workbench.	Cc	C compiler frontends for Manx and Lattice C
Spin3	Graphics demo of spinning cubes, double-buffered example.	Lex	Computes F <sub>g</sub> , F <sub>iesch</sub> , and Kincaid readability of text files.	Copper	A hardware copper list disassembler
Sword	Sword of Fallen Angel text adventure game written in Amiga Basic.	TunnelVision	David Addison ABASIC 3D maze perspective game.	InstIFF	Converts Instruments demo sounds to IFF sampled sounds
Trails	Leaves a trail behind mouse, in Modula-2	Vc	Visicalc-like spreadsheet calculator program.	PopColours	Adjust RGB colors of any screen
<b>Fred Fish Disk 33</b>		Vt100	Version 2.2 of Dave Wecker's telecom program	SpriteClock	Simple clock is displayed on a sprite above all screens
3dstars	3d version of the "stars" program below.	YaBoing	Oing! style game program shows sprite collision detects	ST Emulator	Non-serious Atari ST emulator
Bigmap	Low-level graphics example scrolls bitmap with ScrollVPort.	<b>Fred Fish Disk 37</b>		WBrn	Lets Workbench programs be run from the CLI
Dbuf.gels	Double-buffered animation example for BOBs and VSprites.	This disk is a port of Timothy Budd's Little Smalltalk system, done by Bill Kinnersley at Washington State University.		Wild	Two Unix shell style wildcard matching routines
DiskMapper	Displays sector allocation of floppy disks.	<b>Fred Fish Disk 38</b>		<b>FF 44</b>	
MemView	View memory in real time, move with joystick.	CSquared	Sep 88 Sci American, Circle Squared algorithm	Icons	Miscellaneous icons
Oing	Bouncing balls demo	FixObj	Strips garbage off Xmodem transferred object files	NewIFF	New IFF material from CBM for sampled voice and music files
Sproing	Oing, with sound effects.	Handler	AmigaDOS handler (device) example from C-A	RayTracePics	The famous ray-tracing pictures, from FF #39, now converted to IFF HAM format for "much" faster viewing.
ScreenDump	Dumps highest screen or window to the printer.	Hp-10c	Mimics a HP-10C calculator, written in Modula-2	ViewILBM	Displays normal and HAM ILBM files
Sdb	Simple database program from a DECUS tape.	IFFEncode	Saves the screen as an IFF file	<b>FF 45</b>	
Stars	Star field demo, like Star Trek.	IffDump	Dumps info about an IFF file	Clue	Clue board game
TermPlus	Terminal program with capture, library, function keys, Xmodem, CIS-B protocols.	Jsh	BDS C-like CLI shell	Make	Another 'make', with more features
Vt100	Version 2.0 of Dave Wecker's VT-100 emulator, with scripts and function keys.	NewStat	STATUS-like program, shows priority, processes	Pictures	Miscellaneous pictures
<b>Fred Fish Disk 34</b>		Reversi	Game of Reversi, version 6.1	Update	Updates an older disk with newer files from another disk
Alint	Support files for Gimpel's 'lint' syntax checker	UUdecode	Translate binary files to text, Unix-like programs	WhereIs	Searches a disk for files of given name
Blink	PD 'alink' compatible linker, faster, better.	Vdraw	Drawing program, version 1.14	<b>FF 46</b>	
Browser	Updated to FF 18 'browser', in Manx, with scroll bars, bug fixes.	VoiceFiler	DX MIDI synthesizer voice filer program	Asm	Shareware 68010 macro assembler, ROM Kernel Manual compatible
Btree	b-tree data structure examples	Window	Example of creating a DOS window on a custom screen	CheckModem	'execute' file program detects presence of modem
Btree2	Another version of 'btree'	<b>Fred Fish Disk 39</b>		Egad	Gadget editor from the Programmers Network
Calendar	Appointment calendar with alarm.	AnsIEcho	'echo', 'touch', 'list', 'cis' written in assembler.	Jive	Transforms a file from English to Jive.
Less	File viewer, searching, position by percent, line number.	Display	Displays HAM Images from a ray-tracing program, with example pictures.	MyLib	A binary only copy of Matt's alternate runtime library. Author: Matt Dillon
NewFonts	Set of 28 new Amiga fonts from Bill Fischer	Driver	Example device driver source, acts like RAM: disk	ProffMacros	Subset Berkeley 'ms' and 'mm' macros for 'proff'
Pr	Background print utility, style options, wildcards.	Xlisp	XLisp 1.7, executable only	ValSpeak	Transforms a file from English to Valley Speak.
Requester	Deluxe Paint-type file requester, with sample.	<b>Fred Fish Disk 40</b>		<b>FF 47</b>	
<b>Fred Fish Fred Fish Disk 35</b>		Ahost	Terminal emulator with Xmodem, Kermit and CIS B protocols, function keys, scripts, RLE graphics and conference mode.	3D-Arm	Simulation of a robotic arm, very good graphics, teaching tool, including C source.
ASendPacket	C example of making asynchronous I/O calls to a DOS handler, written by C-A	AmigaMonitor	Dynamically displays the machine state, such as open files, active tasks, resources, device states, interrupts, libraries, ports, etc.	Juggler	Eric Graham's stunning HAM animation of a robot juggler
ConsoleWindow	C example of getting the Intuition pointer a CON: or RAW: window, for 1.2, by C-A.	Arc	Popular file compression system, the standard for transiting files	VT-100	Version 2.4 of Dave Wecker's terminal emulator, with Xmodem and Kermit file transfer protocols
DirUtil	Walk the directory tree, do CLI operations from menus	AreaCode	Program that decodes area codes into state and locality.	<b>FF 48</b>	
DirUtil2	Another variant of DirUtil.	Blink	'alink' replacement linker, version 6.5	Bru	Alpha version of a hard disk file archiver
FileRequester	Lattice C file requester module, with demo driver, from Charlie Heath.	Cosmo	An 'asteriods' clone.	Comm	Version 1.30 of a terminal emulator with phone directories
MacView	Views MacPaint pictures in Amiga low or high res, with sample pictures, by Scott Evernden.	Dg210	Data General D-210 Terminal emulator	Csh	Version 2.04 of Matt Dillon's Unix 'csh'-like CLI replacement, including Lattice and Manx C source
Plop	Simple IFF reader program	DirUtil	Windowed DOS interface program, version 1.4	Diskperf	Disk benchmark program for Unix and Amiga
PopCLI	Sidkick-style program invokes a new CLI, with automatic screen blanking.	DOSHelper	Windowed AmigaDOS CLI help program	Du	Computes disk storage of a file or directory
QuickCopy	Devenport disk copiers duplicate copy-protected disks.	PagePrint	Prints text files with headers, page breaks, line numbers	MemWatch	Program to watch for programs that trash low memory. It attempts to repair the damage, and puts up a requester to inform you of the damage. From the Software Distillery.
ScrollP1	Dual playfield example, from C-A, shows 400 x 300 x 2 bit plane playfield on a 320 x 200 x 2 plane deep playfield.	PopCLI	Starts a new CLI with a single keystroke, from any program, With a screen-saver feature. Version 2, with source.	Profiler	A realtime execution profiler for Manx C programs. Includes C source.
SendPacket	General purpose subroutine to send AmigaDos packets.	SpriteEd	Sprite Editor edits two sprites at a time	<b>FF 49</b>	
SpriteMaker	Sprite editor, can save work as C data structure. Shareware by Ray Larson.	X-Spell	Spelling checker allows edits to files	Cycloids	Update of electronic spirograph from disk 27
Tracker	Converts any disk into files, for electronic transmission. Preserves entire file structure. Shareware by Brad Wilson.	<b>FF 41</b>		DirUtil	Enhanced version of DirUtil from disk 35
TriClops 3-D	space invasion game, formerly commercial, now public domain. From Geodesic Publications.	AmigaVenture	Create your own text adventure programs in AmigaBasic.	MultiDef	Scans a set of object modules and libraries searching for multiply defined symbols
Tsize	Print total size of all files in subdirectories.	Csh	Version 2.03 of Dillon's Csh-like shell. Executable only	MyUpdate	Disk update utility with options for stripping comments from C header files, and interactive verification of the updating process
Unifdef	C preprocessor to remove given #if'd sections of a file, leaving the rest alone. By Dave Yost.	Dbug	Macro based C debugging package, update to FF #2	Plot	Computes and displays 3 dimensional functions in hires
		DualPlayField	example from CBM, update to Intuition manual	Polygon	Moiré type pattern generator with color
		GetFile	Heath's file requester, with source		
		LatXref	Cross reference of Lattice 3.10 header files		

QMouse pressed.	cycling Queries whether a mouse button is pressed. This can give a return code that can customize a startup-sequence based on whether a mouse button was pressed.
Touch	Example of setting the datestamp on a file, using a new technique from Commodore-Amiga
Trees program FF 50	More extensive version of the trees on Disk 31
Asm	Version 1.1 of a shareware 68000 macro assembler, compatible with the Metacomco assembler. This includes an example module and more Motorola mnemonics.
startup BreakOut DiskZap	A brick breakout game, uses 3-D glasses Version 1.1 of a program to edit disks and binary files
FirstSilicon and Missile sound, PerfectSound Sizzlers	A smart CLI replacement with full editing recall of previous commands A Missile Command-type game, with in assembler Sound editor for a low-cost sound digitizer Graphics demos
UnixArc machines, Wombat	Version of 'arc' for Unix System V In C Version 3.01 of Dave Walker's terminal emulator
FF 51	
Bison 4	GNU for Unix 'yacc', working update to disk version
Compress	Update to the file compression program on Disk 6
Cos AmigaBasic DiSsed	"Wheel of Fortune"-type game in Unix-like 'diff' and 'sed' for finding the differences between two files, and then recreating the other, given one file, and the list of differences.
Sq, Usq	Portable versions of the CP/M squeeze and unsqueeze
FF 52	
Assign	Replacement for AmigaDOS 'assign' command in C
Fractal Poly, HAMPoly polygons MxGads	Makes random fractal terrains Workbench-type demos for making in lores and HAM Example of mutual exclusion gadgets with GadgetText
Tek4010	Tektronix 4010 terminal emulator
VDraw	Versions 1.16 and 1.19 of a Deluxe Paint-like drawing program
FF 53	
Animations	Demo animations with player program for Aegis Animator
ARCre	Creates rename scripts for files with long names, so they can be easily 'arc'ed and un'arc'ed.
ARP	Preliminary AmigaDOS replacements for 'break', 'cd', 'chmod', 'echo', 'filenote' and 'makedir'
Compiler	Not fully ported to the Amiga, this is a 68000 C compiler. It will produce simple assembly language output, but needs a lot of work.
Spreadsheet	Update with source of the 'vc' spreadsheet on disk 36
TarSplit	Port of program to split Unix 'tar' archives
UUencode	Utilities to encode and decode binary files for ASCII transmission, expanding them by 35 percent

To Be Continued.....

**In Conclusion**

To the best of our knowledge, the materials in this library are freely distributable. This means they were either publicly posted and placed in the Public Domain by their Author, or they have restrictions published in their files to which we have adhered. If you become aware of any violation of the author's wishes, please contact us by mail.

•AC•

# Need AMIGA Software?



## Try The Public Domain.....

Amazing Computing™ has vowed, from our beginning, to amass the largest selection of Public Domain Software in the Amiga Community, and with the help of John Foust and Fred Fish, we see a great selection of software for both beginners and advanced users.

These Public Domain software pieces are presented by a world of authors who discovered something fun or interesting on the Amiga and then placed their discoveries in the Public Domain for all to enjoy. You are encouraged to copy and share these disks and programs with your friends, customers and fellow user group members!

The disk are very affordable!

Amazing Computing™ subscribers.....\$6.00 per disk.  
Non subscribers.....\$7.00 per disk

This is extremely reasonable for disks with almost 800K of information and programs. If you agree, please send check or money order to:

PiM Publications Inc.  
P.O. Box 869  
Fall River, MA 02722

All Checks must be in US funds drawn on a US Bank

Please allow 4 to 6 weeks for delivery

Amazing Computing™: Your resource to the Commodore Amiga

## DIGITIZED INSTRUMENTS FOR INSTANT MUSIC OWNERS

Here is what you have been waiting for! This disk is full of digital samples of **real** instruments that can be played by your instant music® program at HI FI quality.

- Write & play songs that sound real
- Use in existing songs or use to write your own
- I.F.F. compatible
- Over 30 instruments
- Classical, Contemporary & Rock instruments
- Can be used by 1 drive systems
- Instruments can be copied to other song disks
- Includes guitars (4 types), drums, organs, trumpets, flute, violin, etc.
- Compatible with Deluxe Music

**\$20** POSTPAID

(Connecticut Residents Add \$1.50 Sales Tax)

### Actionware

1039 Farmington Ave.  
West Hartford, CT 06107  
(203) 233-0151

*Instant Music and Deluxe Music are a trademark of Electronic Arts*

## We have those FUNNY CONNECTORS for the back of the Amiga™!

C				S
	The ones for the DISK CABLE	(DB23P)	\$3.00	
	The ones for the RGB MONITOR CABLE	(DB23S)	\$3.00	
A	The ones for the PARALLEL (PRINTER) CABLE	(DB25S)	\$3.00	W
	CENTRONICS MALE/FEMALE			
	The ones for the SERIAL (MODEM) CABLE	(DB25P)	\$3.00	
	CONNECTOR COVERS and SHELLS		\$1.75	
B	We also have the 34 PIN EDGE CONNECTOR for those making the 5 1/4 Floppy Disk Interface			I
	<b>Ready to Use</b>			T
L	<b>2 DRIVE CABLE w/ POWER SUPPLY</b>		<b>\$59.00</b>	
	Cables for PRINTERS and MONITORS		\$13.00	
	3ft. Sony RGB			
E	SWITCHES	AB	\$30.00	C
		ABC	\$35.00	
S				H

## BCD

Jim Black  
P.O. Box #1224  
Bartlesville, OK 74005

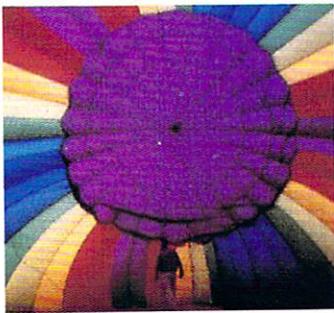
MAIL ORDERS please include \$2.00 Postage & Handling  
PHONE ORDERS (918)-336-1784 (COD only)

## Index of Advertisers

Actionware	96
Adept Software	62
ASDG	70
Akron Systems	38
Alohafonts	22
Alphanetics Apparatus	53
Ameristar Technologies	23
Aminetics	39
Applied Visions	B2
BCD	96
Byte by Byte	CIV
CLtd.	A1
Cardinal Software	26,27
Century Systems	44
Creative Solutions	24
Digi Pix	21
Emusoft Corporation	28
Expansion Technologies	61
Felsina Software	47
Greenthumb Software	45
Hilton Android Corporation	40
Impulse	48,49
Jagware	A2
Jefferson Enterprises	42
Lattice	5
Kent Engineering & Design	79
KJ Computers	89
Magic Circle Software	36
Memory Location, The	76
Meridian Software	33
Metadigm, Inc.	2
Michigan Software	20
Micro Systems Software	66
Microbotics	7
MicroSmiths, Inc.	14
Mimetics	69
New Tek	B1
Pacific Peripherals	87
Phase Four Software Distributors	83
PiM Publications, Inc.	64,65,74
Polyglot Software	19
Progressive Peripherals	C1,CIII
R & S Data Systems	58
SKE Software Company, Inc.	72
Software Factory, The	42
Spirit Technology	85
T's Me	18
T&L Enterprises	30
The Other Guys	63
TDI Software Inc.	41
Technisoft	13
Westcom Industries	51

### Support the Amiga™ & Amazing Computing™, Write!

Your thoughts, experiences, and programs are needed by others. For an Author's guide, write to: Author's Guide, PiM Publications, Inc., P.O. Box 869, Fall River, MA. 02722.



Actual unretouched photos

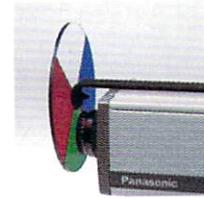
# DIGI VIEW

## brings the world into your Amiga!™



With **Digi-View** and a video camera, your **Amiga** can see! Faces, logos, artwork . . . anything you can imagine! Simply point your camera and click the mouse. In seconds, whatever the camera sees is painlessly transformed into a computer image that can be printed, stored on disk, or transferred to other programs. Imagine how quickly and easily you can generate stunning video art and animation when you start with high quality digitized photographs or artwork.

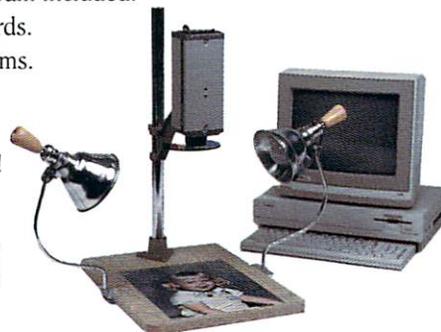
Sophisticated software included with **Digi-View** makes it easy to produce dazzling, broadcast-quality color images. Intuitive, on-screen controls are as easy to use as the knobs on your T.V. set. **Digi-View** can capture images in several modes, including 320x200 pixels with up to 4096 colors on screen ("hold-and-modify" mode), and the incredibly detailed 640x400 high resolution mode.



*The key to Digi-View's incredible color resolution is this color separation filter which attaches to your black-and-white or color video camera.\**

- IFF disk format works with Digi-Paint™, DeluxePaint™, DeluxeVideo™, DeluxePrint, Aegis Images™, Aegis Animator, and more!
- Saves time! No more hours of freehand drawing and redrawing.
- Send photos over the telephone with your modem and terminal software.
- Capture images for scientific image processing or pattern recognition.
- Spice up business graphics — slide show program included.
- Incorporate photos in posters and greeting cards.
- Use **Digi-View** pictures in your BASIC programs.
- Catalog images with IFF database programs.
- Make red/blue 3D photos.
- A powerful tool for commercial graphic artists!

Panasonic WV-1410 video camera w/lens . . . . . \$280  
CS-1L Copy stand w/lights . . . . . \$ 75



### Only \$199.95

includes video digitizer module, color separation filter, software and manual.

Orders Only (800) 358-3079 ext. 342  
Customer Service (913) 354-9332

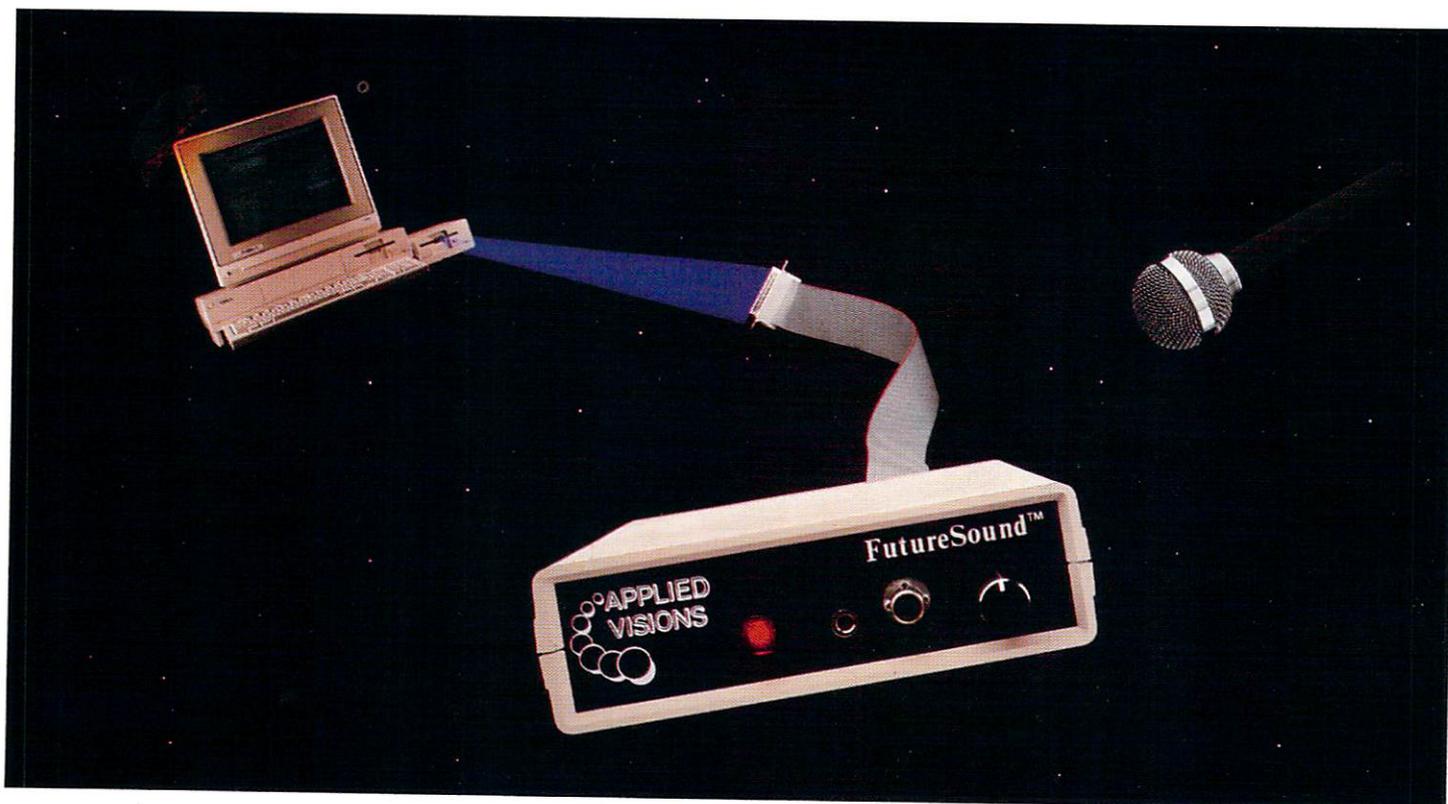
# NewTek

I N C O R P O R A T E D

701 Jackson • Suite B3 • Topeka, KS • 66603

Amiga is a trademark of Commodore-Amiga, Inc. Digi-View and Digi-Paint are trademarks of NewTek, Inc. DeluxePaint, DeluxeVideo, and DeluxePrint are trademarks of Electronic Arts, Inc. Aegis Images and Aegis Animator are trademarks of Aegis Development, Inc.

\* Digi-View software version 2.0 (or newer) required to use color camera. For maximum resolution use monochrome camera with 2.1 interlace. High-res color modes require 1 Meg expansion RAM.



## “Open the pod bay doors, HAL...”

### Programmers cast their vote!

Right now, leading software developers are hard at work on the next generation of Amiga® products. To add the spectacular sound effects we've all come to expect from Amiga software, they are overwhelmingly choosing one sound recording package... FutureSound. As one developer put it, "FutureSound should be standard equipment for the Amiga."

### FutureSound the clear winner...

Why has FutureSound become the clear choice for digital sound sampling on the Amiga? The reason is obvious: a hardware design that has left nothing out. FutureSound includes two input sources, each with its own amplifier, one for a microphone and one for direct recording; input volume control; high speed 8-bit parallel interface, complete with an additional printer port; extra filters that take care of everything from background hiss to interference from

the monitor; and of course, a microphone so that you can begin recording immediately.

### What about software?

FutureSound transforms your Amiga into a powerful, multi-track recording studio. Of course, this innovative software package provides you with all the basic recording features you expect. But with FutureSound, this is just the beginning. A forty-page manual will guide you through such features as variable sampling rates, visual editing, mixing, special effects generation, and more. A major software publisher is soon to release a simulation with an engine roar that will rattle your teeth. This incredible reverberation effect was designed with FutureSound's software.



**Question: What can a 300 pound space creature do with these sounds?**

**Answer: Anything he wants.**

Since FutureSound is IFF compatible (actually three separate formats are supported) your sounds can be used by most Amiga sound applications. With FutureSound and Deluxe Video Construction Set from Electronic Arts, your video creations can use the voice of Mr. Spock, your mother-in-law, or a disturbed super computer.

Programming support is also provided. Whether you're a "C" programming wiz or a Sunday afternoon BASIC hacker, all the routines you need are on the non-copy protected diskette.

Your Amiga dealer should have FutureSound in stock. If not, just give us a call and for \$175 (VISA, MasterCard or COD) we'll send one right out to you. Ahead warp factor one!

Applied Visions, Inc., Suite 2200, One Kendall Square  
Cambridge, MA 02139 (617) 494-5417

Amiga is a registered trademark of Commodore-Amiga, Inc.  
Deluxe Video Construction Set is a trademark of Electronic Arts, Inc.

# CLImate

## The Indispensable Utility Program For The Amiga

- Simple mouse driven operation by-passes the keyboard.
- Batch file operations for copy, delete, move, and print.
- "Showpic" button for easy viewing of picture file.
- Batch printing of files with full control of print format.
- Display files in ASCII/HEX with pause/continue options.
- Format disks with control of format options.
- Compatible with 3½" and 5¼" drives!
- Full support of up to 3 external disk drives, 2 hard drives, and RAM disk.

Requires 512K Ram and Kickstart plus Workbench 1.2

By-pass the difficult Command Line Interface using the AMIGA mouse. CLImate is a time saving program you will never want to be without.

As you will see, CLImate display useful disk and file information which is helpful when batch copying and transferring files.

Easily create and rename directories and files at the touch of a button. With CLImate, a click of a mouse button is almost all you ever need to do.

With SHOWPIC, you can view your IFF/HAM pictures you've created with most popular paint programs and digitizers. Even color cycling is supported.

Disk formatting becomes the simple task of clicking the mouse and entering a disk name. Full control of formatting options is provided.

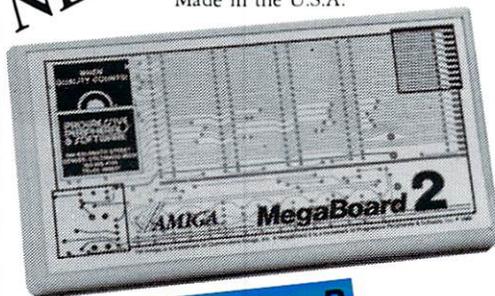
CLImate's print option gives you full formatting control when you print files. Useful headers such as time, date, file name and page/line are options available.

**ONLY!**  
**\$3995**

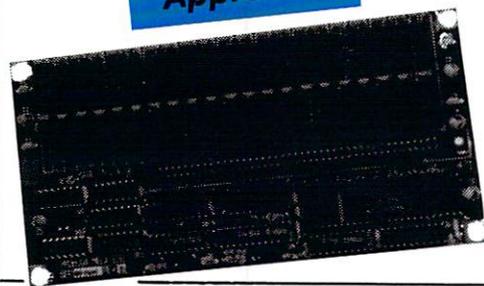
Suggested U.S. Retail Price  
Totally Unprotected

# NEW! MEGA BOARD 2

Memory Expansion for the AMIGA™  
Made in the U.S.A.



**FCC Class B  
Approved!**



### MegaBoard 2 Features:

- 2 Megabytes of fast RAM
- Fully Populated
- No Wait States
- Auto Configure with 1.2 Version Operating System
- Small 7½" x 4" x 1⅛" Footprint
- Fully Amiga Compatible
- Plugs into expansion bus on the side of the Amiga...Ready to go in just minutes!
- Allows better use of memory oriented software, and RAM disk for fast copying and handling.
- Meets all FCC class A & B requirements
- One of the fastest RAMs available!
- Clean, Professional Unit

**NOW!**  
**\$59995**  
U.S. Suggested Retail

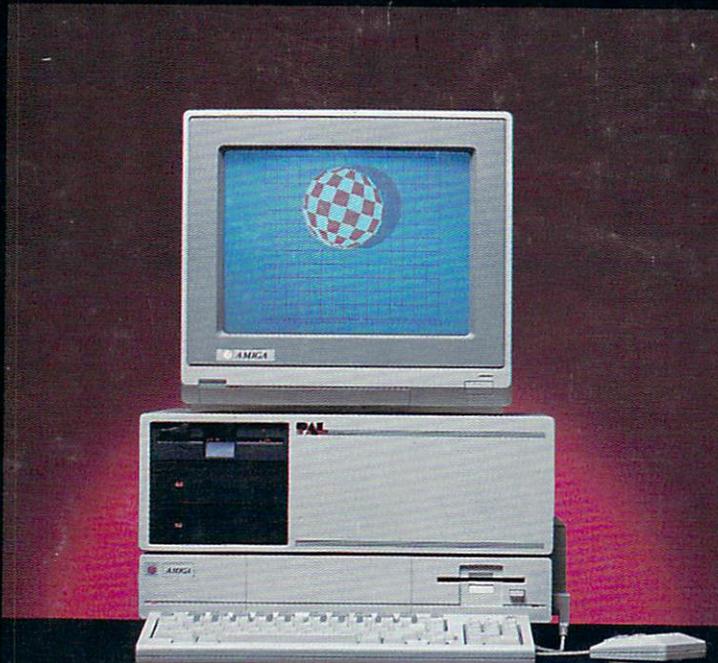


by Progressive Peripherals & Software, Inc.

# UNLEASH THE AWESOME POWER OF THE AMIGA!™

## WITH PAL SYSTEMS

- Supports Three Half Height Devices
  - Hard Disks
  - Tape Backup
  - CD ROM
- Five DMA Expansion Slots
- Battery Backed Clock/Calendar
- Whisper Fan
- Auto-Configure
- 200 Watt Power Supply
- DMA Hard Disk Controller (ST506/412)
  - Optional additional SCSI
- 100% Zorro Compatible
- 1 to 9.5 Megabytes of Fast RAM



## WITH PAL Jr

- One Megabyte of Fast RAM
  - DMA Hard Disk Controller
  - 20 Megabyte Hard Disk
  - Auto-Configure
  - DMA SCSI Pass-through for further expansion
- Suggested retail price only \$1495.



## WITH INFOMINDER

The Information Manager Hierarchical Database that allows you to organize and display text and graphical files. e.g. Real Estate Listings, Personnel Files, Digitized X-Rays, Geographical Maps, etc. Fully supports multi-tasking. Fast access by menu or outline. INFOMINDER will revolutionize the way you store and access both textual and graphical information. Get INFOMINDER today at the special introductory price of only \$89.95.



## WITH TIC

The TIC provides your Amiga with a tiny battery backed clock/calendar that conveniently plugs into the second joystick port. The TIC's 3-year battery will maintain time even if temporarily removed from the Amiga. Change the Amiga's internal time simply by moving the displayed clock's hands with the mouse. Set your Amiga's time once and for all: It's about time for TIC. Suggested retail price only \$59.95.



**BYTE by BYTE**  
CORPORATION

Arboretum Plaza II  
9442 Capital of Texas Highway  
Suite 150  
Austin, TX 78759  
(512) 343-4357