Premier Issue

DISK INSIDE

COMPUTE!'s Atari ST

DISK & MAGAZINE

$12.95
October 1986
Issue 1
Vol. 1, No. 1
$17.95 Canada
ISSN 0888-8442
02617

Features and Columns

3 **The Editor's Page** ..... *Robert Lock*
5 **Readers' Feedback** ..... *The Editors & the Readers*
10 **ST News & Notes** ..... *Arlan R. Levitan & the Editors*
18 **Sam Tramiel Speaks Out** ..... *Tom R. Halfhill & Selby Bateman*
25 **The Comeback of Atari** ..... *Tom R. Halfhill*
72 **Programming in C: Why C?** ..... *Sheldon Leemon*
76 **New Products** ..... *The Editors*

Reviews

28 ***The Pawn*** ..... *Arthur Leyenberger*
28 ***Personal Pascal*** ..... *James W. Maki*
32 ***Megamax C*** ..... *George Miller*
34 ***Thunder!*** ..... *Tom R. Halfhill*
38 ***Major Motion* and *Time Bandit*** ..... *Andy Eddy*
40 ***Sundog: Frozen Legacy*** ..... *Selby Bateman*

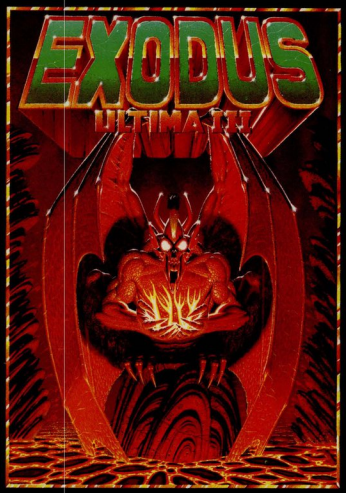
The Disk

42 **AstroPanic!** ..... *Charles Brannon*
44 **AstroPanic!: Annotated Listing** ..... *Charles Brannon*
57 **Short-Term** ..... *Robert M. Birmingham*
60 **Encryptor** ..... *Douglas Wheeler*
62 **Crash Analyzer** ..... *George Miller*
66 **BASIC Animatics** ..... *Robert G. Geiger*
68 **Word Count: A Writer's Accessory** ..... *Tony Roberts*
70 **Atari Art: The Azure Dragon** ..... *Lee Noel, Jr.*

10
0 71486 02617 4

Special Bonus Complete Atari User Group List Included On Disk

"A LIVING TAPESTRY . . ."



NOW AVAILABLE ON ATARI ST

"The world of Ultima III can only be compared to a living tapestry — complex and beautiful . . . This is the best fantasy game in computing. Indeed, it is one of the best fantasy worlds in which to live. Lord British is a veritable JRR Tolkien of the keyboard." — Popular Mechanics

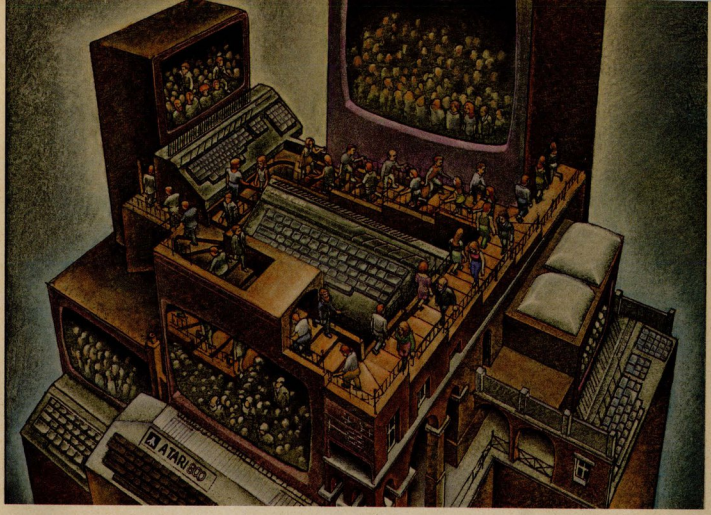
"Exodus: Ultima III, with a superior plot to match its superior gaming system, is a great game. It upgrades the market; in several ways it sets new standards for fantasy gaming state of the art." — Softline

"Exodus: Ultima III is Lord British's magnum opus — so far. It's fun and exciting to play and constantly intriguing. And the ending is marvelously unexpected and not a bit disappointing — except that it is the ending, and as with a good book, you'll probably wish there were more." — Softalk

Available on: Apple, Atari, Com 64, IBM, Macintosh

ORIGIN SYSTEMS INC. 340 HARVEY ROAD, MANCHESTER, NH 03103-3317 (603) 644-3360

Apple, Atari, Com64, and IBM are trademarks of Apple Inc., Atari Inc., Commodore Business Machines, and IBM, respectively.
Ultima and Lord British are trademarks of Richard Garriott. Copyright 1984 by Origin Systems, Inc.



USE THE BRAINS YOUR ATARI WASN'T BORN WITH.

Right at your fingertips in CompuServe's Atari® Forums.

Our Atari Forums involve thousands of Atari users worldwide. These forums show you just how easy and how much fun it is to get the most from your Atari computer.

The **Atari 8-bit Forum** provides the chance to talk with Atari enthusiasts on all levels. You'll learn all kinds of useful information from all kinds of interesting people. It's the leading national Atari Forum.

Devoted exclusively to users of the ST-series, the **Atari 16-bit Forum** provides programs, textfiles, assistance, product reviews, transcripts of online conferences and more.

The **Atari Developers Forum** is the authorized distribution area for all updates to Atari's registered software developer's kit for both the 8-bit and 16-bit Atari Computers. Members may access conferencing channels, data libraries, free programs and software.

Easy access to free software.

- *Download first-rate, non-commercial user-supported software and utility programs.*
- *Take advantage of CompuServe's inexpensive weeknight and weekend rates (when forums are most active, and standard online charges are just 10¢ per minute).*
- *Go online in most major metropolitan areas with a local phone call.*
- *Receive a* ***$25.00 Introductory Usage Credit*** *when you purchase your CompuServe Subscription Kit.*

Information you simply can't find anywhere else.

Use the Forum Message Board to exchange mail with fellow members. Join ongoing, real-time discussions in a Forum Conference with Atari luminaries like Sam Tramiel and Bill Wilkinson. Scan Forum Data Libraries for free software, documentation and contributions from Atari enthusiasts.

Enjoy other useful services too, like electronic editions of your favorite magazines, newsletters and articles, including ANTIC, Family Computing, OMNI Online, and Electronic Gamer.™

All you need is your Atari computer and a modem . . . or almost any other personal computer.

To buy your CompuServe Subscription Kit, see your nearest computer dealer. Suggested retail price is $39.95. To receive our free brochure, or to order direct, call 800-848-8199 (in Ohio, call 614-457-0802). If you're already a CompuServe subscriber, just type GO ATARI at any ! prompt and see what you've been missing.

CompuServe®

Information Services, P.O. Box 20212
5000 Arlington Centre Blvd., Columbus, Ohio 43220

800-848-8199

In Ohio, call 614-457-0802
An H&R Block Company

BATTERIES INCLUDED

Atari ST Software

Integral Solutions

D·E·G·A·S

DESIGN & ENTERTAINMENT GRAPHIC ARTS SYSTEM

by Tom Hudson

The artistic standard for the ST! Beautiful graphics program for business and pleasure. All the key drawing/painting functions, text integration, and graphic design tools! Available: Now!

TiME LiNK

by Softechnics

Scheduling & Time-keeping tool for home and business. Your day, week, month, year at a glance. Many incredible uses! Available: Now!

THUNDER!™

by Mark Skapinker

I see this unique real time spelling checker desk accessory within **any** *ST GEM application. 50,000 word real time spelling checker. Abbreviations function completes a word when you enter the first letters. Amazing speed. Available: Now!*

HomePak

by Russ Wetmore, Dan Moore and Steve Ahlstrom

ST version of InfoWorld's Best Buy of the Year Award! 3 integrated programs on one disk. Telecommunications. Word-processor. Information manager. The easy answer to three key software needs. Available: 3rd Quarter 1986.

BTS THE SPREADSHEET

by Alan Porter, Martin Reznick, Stephen Couchman and Chris Bailey

Two in one! *Sophisticated full featured spreadsheet program. All the key Math, Stats, Logical and Financial functions. 1000 x 1000 worksheet.* **Plus** *Desk accessory version on same disk! Available: 3rd Quarter 1986.*

The Consultant

by Scott Northmore

THE ultimate relational data base. Easy to learn. Unique new features. Power and sophistication you can use right away. For business or personal use. Available: 4th Quarter 1986.

THE ISGUR Portfolio SYSTEM

by Lee Isgur and Mark Skapinker

Investment management program designed for private investors and professionals. On-line portfolio updating. Analytical functions for more profitable decisions. A PC Magazine editor's choice! Available: Now!

I·S TALK

by Stephen Couchman

Full-scale telecommunications program. Easy to use and virtually error-proof. Includes 50,000 word spelling checker and three levels of macros. Available: Now!

D·E·G·A·S Elite

DESIGN & ENTERTAINMENT GRAPHIC ARTS SYSTEM

by Tom Hudson

Second-generation graphics program creates presentation-quality visuals. Full range of design/color functions. Multiple workscreens, new fonts, many other enhancements. The deluxe Degas Elite is totally compatible with all other Degas files! Integrate your Degas Elite pictures with PaperClip Elite text files. Available: Now!

PaperClip Elite

by Dan Moore and Steve Ahlstrom

Next generation word-processor. All the high-productivity features plus a real-time spell checker, idea processing, integrated text/graphics, much more! Available: 3rd Quarter 1986.

B/GRAPH! Elite™

by Alan Page and Robert Wilson

Serious graphics/charting and statistics desk package. Pie charts, 2 and 3 dimensional bar charts, area graphs, much more. Change designs without re-entering data. Make beautiful presentations. Available: 4th Quarter 1986.

I·S TIME AND BILLING

by Roy Miller

Office management program for professionals. Functions include Daily Records, Automatic Billing, Accounts Receivable, Billing Breakdowns and more! Available: 4th Quarter 1986.

I·S TALK, TIME & BILLING, PAPERCLIP ELITE, CONSULTANT, DEGAS ELITE, DEGAS, PORTFOLIO, BTS SPREADSHEET, HOMEPAK, THUNDER AND B/GRAPH! ARE ALL FILE COMPATIBLE, OF COURSE!

BATTERIES INCLUDED

WRITE TO US FOR FULL COLOUR CATALOG OF OUR PRODUCTS FOR COMMODORE, ATARI, APPLE AND IBM SYSTEMS. FOR TECHNICAL SUPPORT OR PRODUCT INFORMATION PLEASE PHONE (416) 881-9816. 30 MURAL STREET, RICHMOND HILL, ONTARIO L4B 1B5 CANADA. (416) 881-9941. TELEX: 06-986-266. 17875 SKY PARK NORTH, SUITE P, IRVINE, CALIFORNIA, USA 92714. (416) 881-9816. Telex: 509-139.

1986 BATTERIES INCLUDED. APPLE, ATARI, COMMODORE AND IBM ARE REGISTERED TRADEMARKS RESPECTIVELY OF APPLE COMPUTERS INC., ATARI INC., COMMODORE BUSINESS MACHINES INC., AND IBM BUSINESS MACHINES INC.

I·S Integral Solutions

Publisher	James A. Casella
Founder/Editor In Chief	Robert C. Lock
Senior Editor	Richard Mansfield
Managing Editor	Kathleen Martinek
Executive Editor	Selby Bateman
Editor	Tom R. Halfhill
Assistant Editor	Todd Heimarck
Production Director	Tony Roberts
Editor, COMPUTE! & COMPUTE!'s GAZETTE	Lance Elko
Technical Editor	Ottis R. Cowper
Assistant Technical Editors	George Miller, Dale McBane
Program Editor	Charles Brannon
Assistant Editor, COMPUTE!	Philip Nelson
Assistant Features Editor	Kathy Yakal
Programming Supervisor	Patrick Parrish
Editorial Programmers	Tim Victor, Kevin Mykytyn, Tim Midkiff
Copy Editor	Tammie Taylor
Submissions Reviewer	Mark Tuttle
Programming Assistants	David Florance, David Hensley
Executive Assistant	Debi Nash
Administrative Assistants	Julia Fleming, Iris Brooks, Mary Hunt, Sybil Agee
COMPUTE!'s Book Division Editor	Stephen Levy
Assistant Editors	Gregg Keizer, Ann Davies
Director, Book Sales & Marketing	Steve Voyatzis
Production Manager	Irma Swain
Art & Design Director	Janice R. Fary
Assistant Editor, Art & Design	Lee Noel
Mechanical Art Supervisor	De Potter
Artists	Debbie Bray, Dabney Ketrow
Typesetting	Terry Cash, Carole Dunton
Illustrator	Harry Blair
Director of Advertising Sales	Peter Johnsmeyer
Associate Advertising Director	Bernard J. Theobald, Jr.
Production Coordinator	Kathleen Hanlon
Promotion Assistant	Caroline Dark
Customer Service Manager	Diane Longo
Dealer Sales Supervisor	Orchid Tamayo
Individual Order Supervisor	Cassandra Green
Receptionist	Anita Armfield
Warehouse Manager	John Williams

James A. Casella, President, COMPUTE!
Richard J. Marino, Vice President, Advertising Sales
Christopher M. Savine, Director, Finance & Planning

Editorial offices:	324 West Wendover Avenue Suite 200 Greensboro, NC 27408 USA
Corporate offices:	825 7th Avenue New York, NY 10019 212-265-8360
Customer Service:	800-346-6767 (In NY 212-887-8525)
Hours:	9:30 A.M.–4:30 P.M. Monday–Friday

Advertising Sales Representatives

New England & Mid Atlantic	Bernard Theobald Andrew Eisenberg (212) 315-1665
Midwest & Southwest	Jerry Thompson 312-726-6047 (Chicago) 713-731-2605 (Texas) 303-595-9299 (Colorado) 415-348-8222 (California)
West, Northwest & British Columbia	Phoebe Thompson 408-356-4994
Southeast & International	Harry Blair 919-275-9809

Call Peter Johnsmeyer, *Director of Advertising Sales or* Bernard J. Theobald, Jr., *Associate Advertising Director (212) 887-8491*

Send all advertising materials to:

Kathleen Hanlon
324 West Wendover Ave., Suite 200
Greensboro, NC 27408

COMPUTE! Publications, Inc.
Part of ABC Consumer Magazines, Inc. One of the ABC Publishing Companies
ABC Publishing, President, Robert G. Burton
1330 Avenue of the Americas, New York, New York 10019

The Editor's View

Premier Issue

When the nearly bankrupt Atari division of Warner Communications spun off Atari Corporation to Jack Tramiel, late of Commodore, the deal was openly derided by some on Wall Street. Now, after a mere 18 or so months, anxious speculators openly ponder the question of when Atari will go public.

At COMPUTE! Publications, we do not take lightly the decision to openly commit our journalistic strengths to a market segment. We must be convinced of the viability of a product, or better, of a family of products. We have made such a decision with the new Atari and with this magazine are evincing such a commitment. We are beholden to no manufacturer and maintain one of the strictest standards of "arm's length" publishing in the industry.

If you are a longtime reader of COMPUTE! Publications, this will not be news to you. If you are a new reader, you'll rapidly reach this conclusion. We are a magazine of readers, of users, of active, committed participants in the application of personal computers in the home, in education, in business, in recreation, in all areas where personal computers are used. We encourage you to submit articles, hints, feedback, and other help and suggestions. We welcome your support and participation in COMPUTE!'s Atari ST Disk & Magazine.

A Major First

No magazine in the history of the publishing industry has ever gone into national newsstand distribution as an integral magazine/disk product. We have purposefully made this product accessible and easy to study at the newsstand. Our disk is unprotected, for your ease of backup. We must rely on you to make this massive experiment in distribution work. The magazine requires the disk, and vice versa. We have avoided plastic wrapping and the problems it causes. We expect you'll respond fairly to what we feel is an excellent value. Thank you, and enjoy this first issue.

Robert C. Lock
Editor in Chief

COMPUTE!'s Atari ST Disk and Magazine is published bimonthly by COMPUTE! Publications, Inc., 825 7th Ave., New York, NY 10019 USA. Phone: (212) 265-8360. Editorial Offices are located at 324 West Wendover Avenue, Greensboro, NC 27408. Domestic Subscriptions: 6 issues, $59.95. POSTMASTER: Send address changes to: **COMPUTE!'s Atari ST Disk and Magazine**, P.O. Box 10775, Des Moines, IA 50347-0775. Second class application pending. Entire contents copyright ©1986 by COMPUTE! Publications, Inc. All rights reserved. ISSN 0888-8442.

The COMPUTE! subscriber list is made available to carefully screened organizations with a product or service which may be of interest to our readers. If you prefer not to receive such mailings, please send an exact copy of your subscription label to: COMPUTE! P.O. Box 10955, Des Moines, IA 50950. Include a note indicating your preference to receive only your subscription.

Authors of manuscripts warrant that all materials submitted to COMPUTE! are original materials with full ownership rights resident in said authors. By submitting articles to COMPUTE!, authors acknowledge that such materials, upon acceptance for publication, become the exclusive property of COMPUTE! Publications, Inc. No portion of this magazine may be reproduced in any form without written permission from the publisher. Entire contents copyright © 1986 COMPUTE! Publications, Inc. Rights to programs developed and submitted by authors are explained in our author contract. Unsolicited materials not accepted for publication in COMPUTE! will be returned if author provides a self-addressed, stamped envelope. Programs (on tape or disk) must accompany each submission. Printed listings are optional, but helpful. Articles should be furnished as typed copy (upper- and lowercase, please) with double spacing. Each page of your article should bear the title of the article, date and name of the author. COMPUTE! assumes no liability for errors in articles or advertisements. Opinions expressed by authors are not necessarily those of COMPUTE!

Another Great Simulation from Sid Meier – Author of F-15 Strike Eagle

Now he takes you from the cold, thin air and limitless space of F-15 Strike Eagle down into the dark depths of the Pacific Ocean inside an American World War II submarine for a realistic, action-filled simulation —

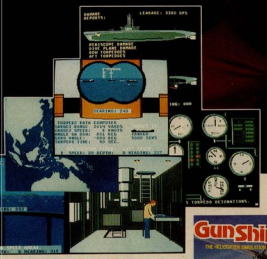
SILENT SERVICE

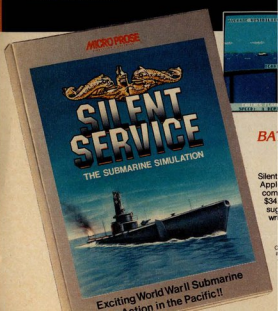
ATARI 520 ST NOW SHIPPING!!

Thrill to the initial sighting of the enemy's strike force in your periscope as their ships come into your range. But watch out — the enemy's escorts have just sighted you. You're the **hunter** — but suddenly — you've become the **hunted!**

As Commander, you must sink their ships and keep your submarine from being destroyed — if you can. Will you select a quiet patrol sector in the Marianas Islands or choose the dangerous waters off the coast of Japan? Is a submerged daylight periscope attack best or do you charge in on the surface at night using only radar bearings to guide you? Do you fire a spread of your precious torpedoes or can you close the range and pick off the enemy with a single torpedo shot? These decisions and many more are yours to make as you take your place among the elite ranks of the **SILENT SERVICE!**

It's **exciting** — and it's **fun.** It's another great Micro Prose simulation — and it's called **SILENT SERVICE.** Look for it now on your dealer's shelves.





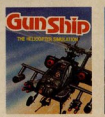
FIVE AUTHENTIC BATTLE STATION SCREENS

Silent Service is available for Commodore 64/128, Apple II Family, Atari XL/XE, IBM PC/PC Jr. computers for a suggested retail of only $34.95. Available soon for Macintosh for a suggested retail of only $39.95. Call or write for more information or MC/VISA orders.

COMMODORE, APPLE, ATARI, IBM, and MACINTOSH are registered trademarks of Commodore Business Inc., Apple Computer Inc., Atari Inc., and International Business Machines Inc., respectively.

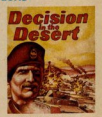
Photo courtesy Baltimore Maritime Museum

Try These Other Real Life Simulations



Daring Attack Helicopter Action in the AH-64 Apache!

Thrilling Decathalon of aerobatic events in your own personal jet!



You are in command — North Africa 1940-1942

MICRO PROSE
SIMULATION • SOFTWARE

120 LAKEFRONT DRIVE • HUNT VALLEY, MD 21030 • (301) 667-1151

Readers' Feedback

Do you have an ST-related question or problem? Have you discovered something that could help other ST users? We want to hear from you. Write to ST Feedback, COMPUTE!'s Atari ST Disk & Magazine, P.O. Box 5406, Greensboro, NC 27403.

Editor's Note: For this premiere issue of COMPUTE!'s Atari ST Disk & Magazine, we have filled this section with questions that might be raised by a wide range of ST users. In future issues, we'll publish actual letters from readers.

Changing GEM Directories

I know that most application programs let you save and load files of some sort—text files for word processors, screens for drawing programs, data files for spreadsheets and databases, and so on. When I drop down the File menu and select Load, a small window with a menu of disk files appears. I understand how to load the file I want by clicking once on the filename and then clicking the OK button, or by rapidly double-clicking on the filename. But sometimes the file I want to load is on a different disk. How can I make the menu display the files on the other disk?

The small window with its menu of disk files is called a GEM item selector. *It's a built-in feature of GEM (Graphics Environment Manager), which is the user-interface portion of the ST's operating system. Any application program—such as a word processor, drawing program, or spreadsheet—can make the GEM item selector appear by calling a certain GEM routine. The idea behind this is to standardize the way application programs load their data files. Once you learn how to use the item selector, you can quickly transfer the knowledge to any new program that supports GEM. In fact, many application programs use the item selector for* saving *data files, too. Figure 1 shows what a typical GEM item selector looks like when it first appears.*

This particular item selector is from 1ST Word, *but all selectors are basically the same. There are four important parts: The central window shows the disk files you can select from; the OK and Cancel buttons let you load a file that has been selected, or abort the loading operation and close the item selector; the short dotted line above the buttons is the Selection field, which shows the file currently selected; and the longer dotted line at the top of the item selector is the Directory field, which shows the* pathname. *It's the pathname that you're interested in changing.*

The pathname tells the item selector which directory of files to display. The selector below has defaulted to the root directory *(main directory) of floppy disk A. There are three ways to load one of the files currently displayed: click on the filename to highlight it, and then click on the OK button; double-click on the filename; or click on the Selection line to make a cursor appear, type in the filename and extender, and then press Return or click the OK button.*

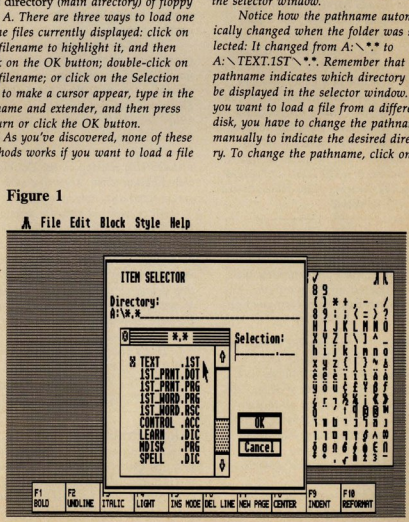
As you've discovered, none of these methods works if you want to load a file which is not displayed by the item selector. The file might be in a folder (subdirectory) on the disk, or it might be on another disk.

If the file you want is in a folder on the same disk, the solution is simple. Folders are denoted in the item selector by a diamond-shaped bullet. For instance, the first item in the selector below, TEXT.1ST, is a folder. To display the files in a folder, simply click on the folder's name. A new selector window opens and shows the files contained in that folder (Figure 2).

Now you can load one of these files by any of the three methods described above. To close the folder and return to the previous directory, click on the small button in the upper left corner of the selector window.

Notice how the pathname automatically changed when the folder was selected: It changed from A:.* to A:\TEXT.1ST*.*. Remember that the pathname indicates which directory will be displayed in the selector window. If you want to load a file from a different disk, you have to change the pathname manually to indicate the desired directory. To change the pathname, click on*

Figure 1



the Directory dotted line to make a cursor appear. You can then edit the pathname by using the keyboard. The cursor-left, cursor-right, Backspace, and Delete keys all work as expected. In addition, the Esc key erases the entire pathname so you can start from scratch. To view a directory on floppy disk B, you change the pathname to B:.*, then click inside the selector window. If you have a two-drive system, the root directory of the disk in your second drive will be displayed instantly. If you have a one-drive system, you'll be prompted to eject disk A and insert disk B.*

The asterisks in the pathname are called wildcards. *In this usage, they indicate that all files in the desired directory should be displayed. If you want, you can display only files you're interested in by replacing one or both of the wildcards with a more specific identifier. For instance, the filenames of all of your* 1ST Word *text files might end with a certain three-character extender, such as .DOC. To display only these files in the selector window, you could change the pathname to B:*.DOC. Try experimenting with different pathnames to get a feel for how they work. (You can't cause any harm by entering an invalid pathname; at worst, you'll simply get a blank selector window.)*

If you happen to know in advance that the file you want to load is both on another disk and in a folder, you can change the pathname to display the contents of that folder. Use the backslash key (\), below the Return key, to separate the name of the folder from the drive identifier and filename identifier. For instance, B:\TEXT.1ST.* would display the contents of the folder named TEXT.1ST on disk B.*

Double Confusion

My friend has a 1040ST and I have a 520ST. Why can't my computer read disks formatted on his computer? His machine can read disks formatted on mine.

The disk drive built into the 1040ST is a double-sided drive: It stores data on both sides of the 3½-inch floppy disk, achieving a total capacity of about 720K. But the disk drive hooked up to your 520ST is probably a single-sided drive: It stores data on only one side of the disk, for a capacity of about 360K. A double-sided drive can read a single-sided disk, but not vice versa.

There are two ways you can make your disks interchangeable. Of course, one solution is to buy a double-sided drive for your 520ST. The double-sided drive's model number is SF314, while the single-sided drive is labeled SF354. Otherwise, they appear identical. (Note, however, that drives released early this summer have a larger, 1040ST-style ejection button.)

A more economical solution is to format disks on the double-sided drive for single-sided use. When you select the Format option from the GEM desktop's File menu, a dialog box lets you choose either density. *If you click on the button labeled* Single Sided, *the disk will be formatted for use on either type of drive. (Incidentally, clicking on the* Double Sided *button when formatting with a single-sided drive has no effect.)*

All commercial software disks are formatted as single-sided disks. If the program requires more files than can fit on a single-sided disk, a second disk is included in the package. The reason for this is that no software publisher wants to neglect any potential customers, and the vast majority of ST drives now in use are single-sided.

That's also why the magazine disk which accompanies every issue of COMPUTE!'s Atari ST Disk & Magazine *is a single-sided disk.*

Printer Problems

When I send a file to my printer, the first line prints normally, but when the printhead reaches the right margin, it hangs and prints all remaining characters in the file at the end of the line. What's wrong?

It sounds as if your printer isn't set up correctly. Check your printer manual for instructions on setting the printer's DIP switches, which are usually found inside or underneath the printer. Your goal is to set the switch so the printer generates a linefeed character (ASCII code 10) after a carriage return (ASCII 13). Some computers don't require this, but the ST does.

Alphabet Soup

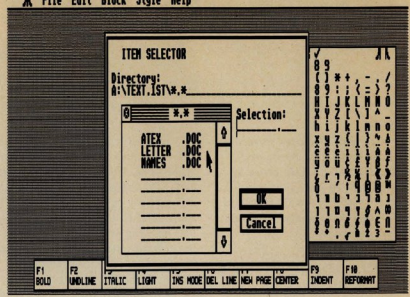
I keep hearing about TOS, GEM, GEMDOS, VDI, BIOS, XBIOS, AES. What do these terms mean, and how are they related to the ST?

These are all parts of the ST's system software—the operating system that controls the function of the computer and the desktop user interface. They are mainly of interest to programmers.

TOS stands for Tramiel Operating System. It's the overall term for the ST's system software; everything else falls beneath the TOS umbrella. The main components of TOS are CP/M–68K and GEM.

CP/M–68K is the underlying operating system within TOS. Supplied by Digital Research, it's an advanced version of the popular CP/M (Control Program/Microcomputers) operating system for the eight-bit Z80 microprocessor. The 68K in CP/M–68K refers to the 16/32–bit 68000 microprocessor, which is the heart of the Atari ST. By running a special program, it's possible to exit

Figure 2



the ST's desktop environment and enter CP/M–68K commands at a CP/M–style prompt, but there's rarely any need to do so.

GEM is the Graphics Environment Manager. This is the desktop user interface which resembles those found on the Macintosh and Commodore Amiga. GEM, in turn, is made up of other parts which handle the major graphics routines of the ST.

VDI (Virtual Device Interface) is a part of GEM which provides low-level graphics display and mouse input routines. This subroutine library includes primitive drawing operations as well as some routines for display management.

AES (Application Environment Services) performs higher-level graphics and data-management operations for maintaining the GEM desktop. AES routines are built on top of GEM VDI and GEMDOS, making it easier for programs to perform mouse and window operations.

GEMDOS (GEM Disk Operating System) performs character-oriented file and device input/output (I/O).

BIOS (Basic Input/Output System) is a low-level group of routines which perform machine-specific tasks governing I/O.

XBIOS (eXtended Basic Input/Output System) provides additional machine-specific operations.

For more detailed information, you'll have to refer to some programming books for the ST, such as COMPUTE!'s ST Programmer's Guide *(COMPUTE! Books). There's a great deal to learn if you want to write sophisticated programs on the ST.*

Assembler Listings

I'm using AS68.PRG from Digital Research to assemble machine language programs. Is there any way I can make a hardcopy listing of a program as the object code is being assembled?

The easiest way to create a listing is to write a listing file to disk. All you need to do is add the parameter −P and specify a filename for output. You can do this either from a batch file, or by installing AS68 as TOS Takes Parameters *(.TTP) and executing it from the desktop.*

In either case, enter this line to assemble and write a listing file to disk:

AS68 −P −L −U sourcefile.s > listingfile.L

The listing file created is an ASCII file which may be printed out using a word processor, such as 1ST Word *or* ST Writer, *or by double-clicking on the file from the GEM desktop, then selecting the Print option.*

Here's a list of some other parameters for use with the AS68 assembler.

AS68 Assembler Parameters

Parameter	*Effect*
−F b:	*Allows you to specify on which drive work files will be created. b: is the character code of the drive to be used. If you skip this parameter, the assembler defaults to the active drive.*
−I	*Initializes the assembler. This parameter is not necessary for the ST. The initialization process creates the file AS68SYMB.DAT.*
−P	*Outputs a listing. Usually the list appears on the screen, but it may be diverted to an .L file for printing, as explained above.*
−S b:	*Allows you to declare which drive contains the AS68SYM-.DAT file. b: is the character code of the drive to be used. Skipping this parameter defaults to the active drive.*
−U	*All undefined labels will be handled as global values and can be used in linking other programs.*
−L	*Sets up all constants as long words.*
−N	*Prevents the assembler from automatically converting the JSR instruction into the BSR instruction.*
−T	*Allows the assembler to accept 68010 opcodes.*

Custom Icons

I'd like to replace the icons on the GEM desktop with customized icons of my own design. How can I replace them?

There's no way to replace the icons on the GEM desktop. If you write your own application program, however, you can customize the icons. The Resource Construction Set included in Atari's Developer System Kit has an icon editor, and several commercial icon editors are available. Using custom icons in your own application program isn't very difficult, but it does require interfacing with GEM.

Auto Folders

The floppy boot disk for my hard disk drive requires a folder named AUTO, which must contain the boot file for the hard disk. How does an AUTO folder work?

When an Atari ST system is switched on (booted), it looks on the disk inserted in drive A for a folder named AUTO. If it finds one, it executes any programs found inside the folder. These programs are known as COMMAND.PRG files.

To create your own AUTO folder, place the disk you use to boot your ST in the drive and select the New Folder option from the File menu. Enter AUTO as the name of the folder. Then move the program you want to automatically run into the AUTO folder. The filename of the program must have a .PRG extender.

If you have more than one program in the AUTO folder, the programs will be executed in the order in which they were placed in the folder. That is, the first program placed in the folder will be executed first, the second program placed in the folder will be executed second, and so on.

There are only two caveats: A program that has GEM features (drop-down menus, windows, and so on) cannot be auto-starting, since GEM itself has not been activated at this stage in the boot process; and the AUTO folder is not completely dependable when using the current version of the operating system. We've noticed that sometimes a program in an AUTO folder won't execute, especially if there are several programs in the folder. Perhaps this will be corrected in a future revision.

IBM Compatibility

I'd like to run IBM PC programs on my ST. I've heard that there will be an IBM emulator, but how well will it work?

That remains to be seen. Atari acknowledges that the MS-DOS emulator it is preparing to release is not 100 percent PC-compatible, but says it will run most of the popular IBM software. Keep in mind that even the so-called IBM clones are not completely IBM-compatible.

At the Spring Computer Dealer Exposition (COMDEX) in Atlanta last May, we saw a prototype of Atari's MS-DOS emulator running Microsoft's Multiplan. *Over a period of four days, however, this was the only IBM program we saw running on the emulator, suggesting that perhaps it wasn't quite perfected. For more information, see the interview with Atari President Sam Tramiel elsewhere in this issue.*

A couple of independent companies have also been working on MS-DOS emulators for the ST, but they have yet to be exhibited in public.

As an interesting sidelight, Atari is also reportedly working on an ST emu-

lator for the PC. This plug-in board would allow a PC to run most ST software when using GEM, which originally was developed for the PC.

Composite Video Output

I've been trying to find out how I can hook up a composite video monitor to my ST. Can you help?

Only late-model 520STs—not 1040STs or early 520STs—support composite video. If your 520ST has an RF (radio frequency) output for TV hookup, then it's also capable of driving a composite video monitor.

The 520ST Owner's Manual *supplied with early 520STs shows a pinout map of the monitor jack on page 75. According to this pinout, pin 2 is reserved. Page 83 of the* 1040ST Owner's Manual *has a similar pinout map that shows pin 2 is for composite video. But this page also shows that the 1040ST has a TV output which is in error. Atari originally planned to include a TV output in the 1040ST, but changed plans at the last minute.*

If you bought a 520ST during the spring or later, it should be equipped with a TV output and should also support composite video on pin 2. Hooking up a composite monitor is simply a matter of connecting pin 2 and the ground (pin 13) to the corresponding inputs on the monitor. Unfortunately, the 13-pin DIN connector used by Atari for the video jack is nonstandard and virtually impossible to find. You could probably rig up some alligator clips to the pins, but the chance of shorting out the signal would be great. Some cable manufacturers have located sources for the 13-pin connectors and should be selling composite video cables by the time you read this. Check with your local Atari dealer.

If you have a 1040ST or an early 520ST and want composite output, you'll have to wait until someone designs an external converter to change the RGB output to composite video. We know of some companies working on this.

Modifying The Desktop

My boot disk has a file called DESKTOP.INF which apparently contains information about the GEM desktop. Can you interpret the information in this file?

DESKTOP.INF is created when you choose the Save Desktop selection from the Options menu. It contains all the information needed to recreate the current state of your GEM desktop—the resolution mode, where the icons are to be placed on the screen, the icon captions, and the size and position of disk windows. The figure below shows the contents of a typical DESKTOP.INF file.

```
#a000000
#b001100
#c77700070007000700552005555222
 07705570750577055041123O6
#d
#E 9B 03
#W 00 00 0C 01 1D 16 08 A: \*.*@
#W 00 00 28 01 1F 17 00 @
#W 00 00 0E 09 2A 0B 00 @
#W 00 00 0F 0A 2A 0B 00 @
#M 00 02 00 FF A FLOPPY DISK@ @
#M 00 03 00 FF B FLOPPY DISK@ @
#T 00 07 02 FF TRASH CAN@ @
#F FF 04 @ *.*@
#D FF 01 @ *.*@
#G 03 FF *.PRG@ @
#F 03 04 *.TOS@ @
#P 03 04 *.TTP@ @
```

Each character in this file gives information about your desktop. A good way to learn more about DESKTOP.INF is to make changes to the file by loading it into a word processor that can save files in ASCII format. But be sure you experiment with an expendable copy of your boot disk, not the original. Any change will affect what you see on the desktop and, to a certain extent, how your ST functions. Use caution, since some changes might not yield the results you expect.

Some of the more interesting parameters to change in DESKTOP.INF are the first two lines, which begin with #a and #b. These digits control the keyboard repeat rate, keyboard repeat delay, key click, and the bell. Of course, all of these parameters can be more easily (and safely) changed with the Control Panel.

The line which begins with #c contains the 16 color palette values. Even the high-resolution monochrome and medium-resolution color modes save 16 values in DESKTOP.INF. The numbers are stored in octal (base eight) format, with three numbers for each palette. Why three numbers? They are the red-green-blue (RGB) values. They, too, can be easily changed with the Control Panel.

The line which begins with #E is interesting because it controls how disk folders are displayed: as text or icons, and sorted by name, date, size, or type. These values can be changed by dropping down the View menu on the desktop.

The lines which begin with #W control the windows. The first two sets of numbers, 00 00, are for the horizontal and vertical sliders. The zero values indicate the sliders are not active. The next two values, 0C and 01, are the x *and* y *position in character cell units. 1D and 16 are the width and height in character cell units. The remaining information governs the path for the directory.*

The next grouping is for the disk icons. The first two numbers are instructions on where the icon should appear. Next is a numeral for the icon type. (A file-drawer icon is indicated by 00. Other possible values are 01 for a document icon, 02 for a trash can icon, 03 for a program icon, and 04 for a folder icon.) The FF value doesn't seem to change anything. Finally, these lines contain the disk icon captions.

The line which begins with #T describes the trash can icon. The parameters are the same as for the disk icon.

Next is information for displaying folders, documents, GEM, TOS, and TOS Takes Parameters files in directories.

To modify parameters not changeable from the Control Panel, use a text editor such as Mince *or* EMACS *or a word processor such as* ST Writer *or* 1ST WORD. *If you're using a word processor, set the left and top margins to zero, and be sure to save the file in ASCII format under the filename DESKTOP.INF. With* 1ST Word, *turn off WP Mode before saving.*

To install a revised desktop, turn off your ST, insert the disk with the new DESKTOP.INF file in drive A, then turn the computer back on.

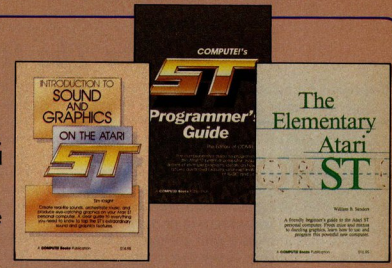
If you're using a hard drive, disconnect the drive while experimenting with DESKTOP.INF. We know of one case where changing values in the DESKTOP.INF file is suspected of causing a hard disk failure with loss of data.

ST

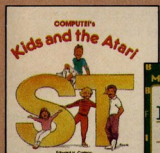
COMPUTE!'s Atari ST Disk & Magazine Toll Free Subscription Order Line
1-800-247-5470
In IA 1-800-532-1272

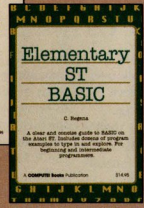
COMPUTE! Books' ATARI ST Collection

COMPUTE! Books offers you a superior line of titles for the new Atari ST. Packed full of useful utilities, exciting games, in-depth tutorials, and valuable applications, these clearly written books bring you fully tested information and entertainment for the whole family.



Look for these COMPUTE! books at your local book or computer store.





You can order directly from COMPUTE! by calling 800-346-6767 (in NY call 212-887-8525) or by sending your payment to COMPUTE! Books, P.O. Box 5038, F.D.R. Station, New York, NY 10150.

Please include $2.00 per book shipping and handling for U.S. and surface mail or $5.00 for airmail. North Carolina residents add 4.5 percent sales tax.

Please allow 4–6 weeks for delivery from receipt of order.

COMPUTE! books are available in the U.K., Europe, the Middle East, and Africa from Holt Saunders, Ltd., 1 St. Anne's Road, Eastbourne, East Sussex BN21 3UN, England, and in Canada from McGraw-Hill, Ryerson Ltd., 330 Progress Ave., Scarborough, Ontario, Canada M1P 2Z5.

COMPUTE! Publications, Inc. abc
Part of ABC Consumer Magazines, Inc.
One of the ABC Publishing Companies
825 7th Avenue, 6th Floor, New York, NY 10019
Publishers of COMPUTE!, COMPUTE!'s Gazette, COMPUTE!'s Gazette Disk, COMPUTE! Books, and COMPUTE!'s Apple Applications.

COMPUTE!'s First Book of the Atari ST
Edited
A valuable collection of ready-to-type-in-and-use applications, games, and utilities. Graphics utilities like "ST Doodler," games like "Switchbox" and "Tug-a-War," and educational programs like "Hickory Dickory Dock" turn your Atari ST into everything from a business graphics machine to a powerful teaching tool. Tutorials show you how to add power to ST BASIC and how to add excitement to your own creations with sound effects. *A disk is available for $15.95 which includes all the programs in the book, 203BDSK.*
(September release)
$16.95 ISBN 0-87455-020-3

The Elementary Atari ST
William B. Sanders, 272 pages
A friendly, easy-to-use guide to the Atari ST, this book takes you through connecting your computer, loading programs, creating graphics and music, and writing your own programs.
$16.95 ISBN 0-87455-024-6

Elementary ST BASIC
C. Regena, 208 pages
A tutorial and reference guide to the ST's impressive graphics, animation, and sound with complete descriptions of ST BASIC's commands, syntax, and organization. *A disk is also available for $15.95 which includes programs from the book, 343BDSK.*
$14.95 ISBN 0-87455-034-3

COMPUTE!'s Kids and the Atari ST
Edward H. Carlson, 238 pages
Easy-to-understand instructor notes, lessons, assignments, and lively illustrations help both kids and adults painlessly learn to program on the Atari ST. The latest in the bestselling series by this author.
$14.95 ISBN 0-87455-038-6

COMPUTE!'s ST Programmer's Guide
Editors of COMPUTE!, 356 pages
A comprehensive reference guide to the Atari ST, this book explores in detail Logo and BASIC, the advanced features of the ST such as GEM and TOS, and every aspect of programming from concepts to actual program writing.
$16.95 ISBN 0-87455-023-8

Introduction to Sound and Graphics on the Atari ST
Tim Knight, 197 pages
Thorough descriptions of the Atari ST's color graphics and sound abilities, plus all the information needed to create a complete sound and graphics system.
$14.95 ISBN 0-87455-035-1

STs On The Half Shell

Overheard during dinner at Pacific Fresh, a Sunnyvale seafood eatery just a stone's throw from Atari headquarters on Borregas Avenue:

Atari's next computer will attempt to penetrate Fortune 500 companies in a big way. To be announced and shown in November at the Winter Computer Dealer Exposition (COMDEX) in Las Vegas, the Atari GST (the actual name varies depending on the day of the week) will be driven by a 32-bit 68020 microprocessor with a whopping four megabytes of memory. A two-meg version of the GST may be shown in the U.S., but will probably be marketed only in Europe.

The GST's operating system will be a System V–compatible Unix, although a hardware switch will allow the system to be booted in ST mode to run ST software. A 20- or 30-megabyte hard disk drive will be standard, along with an MS-DOS emulator and built-in networking capabilities.

Even more interesting is the GST's monochrome-only screen display, which will be vertically oriented and boast a resolution of 1024 × 1024 pixels for true workstation quality. It's said to include most of the functionality of AT&T's BLIT (Bell Labs Intelligent Terminal).

At this point, Atari hopes to start shipping the GST by the end of the second quarter in 1987. T-NET networking software is planned to be available in the third quarter.

The price of this package should send severe shock waves throughout the industry: Watch for a sticker price around $1,995 to $2,495, plus a network-capable laser printer for about $1,000.

The Computer For The ReST Of Us

What has Apple Computer seeing red these days? A little plug-in cartridge with a dash of software that can turn an Atari ST system into a Macintosh. It wowed onlookers at the West Coast Computer Faire in San Francisco last spring by running Mac software even faster than the Mac.

Called the M-Cartridge, the device is the brainchild of David Small, whose name is already well-known to many eight-bit Atari users. Small and his wife Sandy headed up the "Outpost Atari" column in the now-defunct *Creative Computing* magazine for several years. Small was also the father of the LE Development System, a nifty—albeit somewhat cranky—high-speed disk interface for eight-bit Atari software developers.

We reached Small at his Rocky Mountain digs to talk over old times and his new project.

"I started working on M-Cartridge around March of 1985," he says. "It just seemed like a good idea at the time. It eventually turned into an obsession. I spent hours poring over the [operating system] code in my Mac's ROMs [read only memory chips]. The toughest part was trying to sort out all of the nasty little tricks that the Mac developers had used to fit their code into the limited 64K ROM space of the Mac."

It was a lot of work, but Small has his own methods for boosting personal productivity. "I'm a firm believer in the 'no gain without pain' theory. Most of my late-night development was done to the strains of the *Neil Young—Live* album."

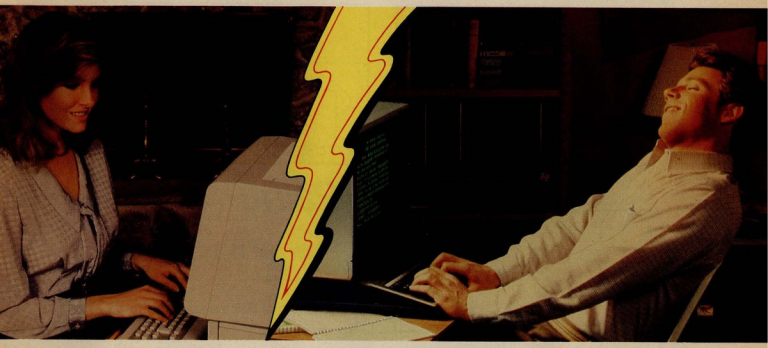
Small's M-Cartridge made its public debut at the Computer Faire in early May. "The first visitor at our booth was Apple," he says. The Apple personnel were apparently shocked and somewhat incredulous. "They wanted to know where we got the ROMs for the demo unit." Apple backed off somewhat after learning that the ROMs were genuine Apple parts that had been legally purchased from a dealer.

At first, Small planned to call his invention the MacCartridge. Those plans changed when representatives of Apple's legal department informed Small that *Mac* is a proprietary Apple trademark. (Of course, there's a plethora of *MacThis* and *MacThat* products currently being sold by other companies.)

To get around these legal problems, Small hopes to sell the M-Cartridge as a cartridge shell with two empty ROM sockets. It will be up to the purchaser to acquire and install a pair of Macintosh ROMs to make the cartridge fully functional. The M-Cartridge will, however, come with Small's special emulator software that

Arlan R. Levitan & The Editors

Become A "Plinker"... And Fall In *LOVE* With Your Modem



Meeting new friends is what People/Link is all about; whether it's via live PartyLine chatting or in one of the popular special interest clubs. There are clubs for computer buffs, hobbyists, dating, sports, religion, lifestyles, and more. And, of course, they all have complete upload and download support. Our **private** mail system even allows "power" users to send binary or ASCII files of up to 192K.

	REGISTRATION FEE	HOURLY NON-PRIME TIME RATES*		
		300 baud	1200 baud	2400 baud
People/Link	**$10.00**	**$ 4.25**	**$ 4.95**	**$ 7.95**
Genie	$18.00	$ 5.00	$ 5.00	$15.00
Compuserve	$39.95	$ 6.25	$12.75	$19.25
The Source	$49.95	$ 8.40	$10.80	$12.00
Delphi	$49.95	$ 7.20	$ 7.20	$ 7.20

Rates in effect June 19, 1986. Rates subject to change.
*Non-prime rates in effect 6pm - 7am local time & all day Sat - Sun.

You may call now to sign-up or to receive more information:

VIA MODEM
(800) 826-8855
(312) 822-9712 (ILLINOIS RESIDENTS)

VIA VOICE LINE
(800) 524-0100
(312) 870-5200

AMERICAN PEOPLE LINK

Send Me FREE Information Packet

Name _____

Address _____

City _____

State _____ Zip _____

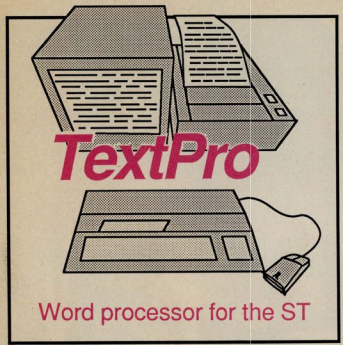
MAIL TO: American PEOPLE/LINK
Arlington Ridge Office Center
3215 N. Frontage Road - Suite 1505
Arlington Heights, Illinois 60004

USE YOUR VISA, MASTERCARD OR AMERICAN EXPRESS CARD.

GEnie is a trademark of General Electric Company. CompuServe is a trademark of CompuServe Incorporated, a subsidiary of H&R Block Company. The Source is a trademark of Source Telecomputing Corporation, a subsidiary of the Reader's Digest Association. Delphi is a trademark of General Videotex Corporation.

DYNAMITE ST SOFTWARE

that won't blow your budget



TextPro is a professional quality wordprocessor for the ST. In addition to standard options found in most other word processors, **TextPro** features full-screen editing via mouse or keyboard, multi-column output, automatic indexing and table of contents generation, sideways printing (to Epson printers), up to 30 user definable function keys, mode for editing C source programs and flexible printer driver installation. It is designed with fast entry of text in mind. The advanced **TextPro** user can substitute shortcut keyboard commands for the drop-down menu commands. **$49.95**

PaintPro is a friendly, yet powerful design and painting package for drawing graphic and artistic pictures. This GEM-based package supports up to **three** active windows—cut & paste from one window to another. Complete toolkit of functions: free-form sketching, lines, circles, ellipses, boxes, fill, copy, move, spray, zoom, undo, help and extensive text capabilities. Import "foreign" pictures for enhancement using **Paintpro**'s double-sized picture format and send hardcopy to most popular dot-matrix printers. **PaintPro** works with either monochrome or color systems. **$49.95**

Other sensational ST software available:

ST Forth/MT - Powerful, full-featured, multi-tasking Forth. 32-bit implementation based on Forth83 standard. Full screen editor, monitor, macro assembler. 1500+ word library. **$49.95**

ST Text Designer - Ideal package for page layout. Accepts prepared text files from TextPro, other ASCII wordprocessors. Perform block operations—copy, move, columns. **$49.95**

ST AssemPro - Professional developer's package. Editor, interactive two- pass assembler with error locator, online help, monitor- debugger, disassembler, 68020 simulator. **$59.95**

ST FilePro - Simple-to-use, versatile data manager. Help screens; tailorable display - multiple fonts; RAM disk support; Multiple files; 64,000+ records. Use with TextPro. **$49.95**

PCBoard Designer - Interactive, computer-aided design package for pc boards layouts. Auto-routing, pinout & component list, Two-sided boards. 45° & 90° traces. **$395.00**

Call **now** for the name of your nearest dealer. Or order directly with your MC, VISA, or Amex card. Add $4.00 per order for postage and handling. Foreign add $10.00 per item. **Other software and books coming soon.** Call or write for your **free** catalog. Dealer inquiries welcome-1400 dealers nationwide.

Abacus Software

Phone (616) 241-5510 - Telex 709-101

P.O. Box 7219 Dept.S9 Grand Rapids MI 49510

makes the ST appear like a Macintosh to the Mac ROMs. According to Small, the M-Cartridge will be marketed by Data Pacific and will be available by this September for $50–$100.

The M-Cartridge does have a number of restrictions, however. First, the emulator software requires almost 512K of memory, which means a megabyte is a must. A one-meg ST system with the M-Cartridge behaves like a 512K Mac.

A second restriction is that the M-Cartridge works only with monochrome ST systems. (The Mac, of course, has no color graphics.)

Another problem is that an ST equipped with an M-Cartridge still can't read or write Macintosh-format disks. Any Mac software you want to run must be transferred to an ST-format disk—either by modem or null modem cable. This means that copy-protected Mac software can't be transferred to the ST without breaking the copy protection.

But perhaps the biggest problem will be where to get a set of Macintosh ROMs to plug into the M-Cartridge's empty sockets. Although Small was able to legally acquire a pair of ROMs from a local Apple dealer, most ST owners will have difficulty locating a set. Since becoming aware of the M-Cartridge, Apple has reportedly reminded its dealers that spare ROMs are intended for replacing chips that fail in real Macintoshes.

In the meantime, Apple has released a new version of the Macintosh ROMs for the Macintosh Plus. These ROMs contain 128K of operating system code instead of 64K. Will there be a new version of the M-Cartridge to support the 128K ROMs?

"No. I'm not a masochist," says Small wearily. "I've spent almost a year and a half learning the innards of 64K of code. I think proving it could be done is enough for now."

Good Posture Counts

Think your new double-sided ST disk drives are the last word in storage? Think again. Toshiba of Japan has shipped samples of a new four-megabyte 3½-inch drive to computer manufacturers. The new drives achieve their high storage density by using a special barium ferrite medium and a magnetization technique called *vertical recording*. The ferrite particles actually stand straight up on the surface of the medium, yielding the capacity for five times more magnetic flux reversals per linear inch of medium. Present processes lay the particles flat in an end-to-end manner.

Déjà Vu Department

Two years ago, eight-bit Atari users who wanted the best display for their dollar often gritted their teeth and opted for the Commodore 1701 composite color monitor. But then, Commodore (still under the management of Jack Tramiel) changed suppliers and replaced the 1701 with the 1702, a look-alike which many people nevertheless believed to be inferior. Now, in 1986, is history repeating itself?

Observant Atari users this summer noticed a change in the appearance of the SC1224 color monitor for the ST. Industry sources report that the switch from original supplier Panasonic (Japan) to Goldstar (Korea) was made to reduce Atari's costs of bringing ST systems to market. It seems that the declining yen/dollar exchange rate has been cutting into Atari's already-low margins with its Far East manufacturers.

According to sharp-eyed ST users, in side-by-side comparisons, the new model plays second fiddle to the old in terms of sharpness and color intensity.

Since the model numbers haven't been changed, how can the average person tell the two monitors apart? The new Goldstar unit sports the same one-color case styling as the SM124 monochrome monitor. The older version has a two-tone case.

The official line from Atari? The two versions are "substantially similar" in quality.

Meanwhile, some monochrome monitor owners are unhappy with the Cinemascope aspect ratio of their black-and-white displays. They've taken to tinkering with the potentiometer which controls the vertical height adjustment inside the SM124 in order to fill more of the screen's vertical space. Usually this also requires a change in both the focus and width potentiometers. Unfortunately, this requires the tinkerer to maneuver the adjusting tool uncomfortably close to some particularly nasty high voltages. Sounds like a job which should be left to skilled electronics repair people.

MIDI Products Making Waves

The ST is the first personal computer with a built-in MIDI (Musical Instrument Digital Interface) port. This port saves musicians from spending $50–$100 for a separate MIDI interface box. However, until recently, there was not much music software on the market for the ST.

Now that situation is changing. Hybrid Arts (Los Angeles) came to the National Association of Music Merchants Show in Chicago recently with three products that can turn the ST into a powerful music studio.

The first product, *DX-Droid*, is a $245 librarian/editor program that links the ST via MIDI cables to a Yamaha DX- or TX-series keyboard synthesizer. The *DX-Droid* librarian lets you store banks of 32 musical instrument sounds (voices) in the ST's main memory or on disk. Each instrument stored in memory can be played on the Yamaha keyboard simply by pushing a button. And the *DX-Droid* editor lets you display pictures of each instrument's sound wave and modify the wave to create a new sound.

DX-Droid has a certain amount of intelligence, too. If you turn it loose, it automatically fills up your disk with new instrument sounds it creates itself. Most of these sounds are random and not especially musical. But many turn out to be similar to real musical instruments or quality synthetic sounds and are thereby quite appealing. *DX-Droid* creates new sounds using many different methods, including random sound generation, averaging two instrument sounds together (a piano and a saxophone, for instance), merging voices via a glide function, and randomly distorting a real instrument sound.

The second Hybrid Arts ST product is *EZ-Track ST*. For $65, it turns the computer into a 20-track MIDI digital tape recorder. *EZ-Track* lets you record music on up to 64 MIDI keyboards and other MIDI-connected devices, including drum machines, guitars, harmonicas, harps, or whatever. You can combine separate tracks into a single track, copy sounds from one track to another, record in real-time or step time, vary your song's tempo from one-half beat per minute up to 480 beats per minute, and control all of your instruments from a single master keyboard. The program works with all MIDI equipment, regardless of the manufacturer. Commands are entered with the mouse or keyboard.

Professional musicians who demand even more performance can purchase the *MIDITrack ST* for $375 or the *MIDITrack Pro ST* for $575. These programs let you edit sound waves by playing with a picture of the wave on the ST screen. They let you create your own movie soundtracks by synchronizing your music and sound effects with a videotape via a SMPTE (Society of Motion Picture and Television Engineers) clock. On a 1040ST, you can record up to 60 instrument sounds and up to 100,000 notes in just one song. According to Frank Foster, chief executive officer of Hybrid Arts, "You can do a whole movie score and not run out of memory."

A third ST music product is the *ADAP SoundRack* from Hybrid Arts and Nilford Laboratories. (ADAP stands for Analog to Digital Audio Processor.) The *SoundRack* is expensive (from $1,995 to $15,000), but, according to Foster, it performs "better than audio processors that cost over $150,000." When plugged into an ST, it lets you sample sounds for 20–80 seconds, then store those sounds in the computer. You can store up to 64 samples in memory for instant playback on an electronic keyboard hooked up to the ST.

Once a sound is in memory, you can use the *SoundRack* to look at a graphic representation of the sound on the screen. Then you can cut pieces out of the sound wave and paste them on different parts of the wave. You can also modify and create new sound waves by using digital effects processing (including delay, echo, reverb, and so forth).

The Ear

News, rumors, and gossip heard around the ST community.

Mum's The Word

At this writing, Atari is expected to soon announce a **major software deal** with a big-name software company—one of the biggest, in fact. If the deal goes through, it should **turn a lot of heads** and gain new respect for Atari and the ST. Even the **Macintosh and IBM people** will be impressed. Sorry, but we promised not to reveal any more about this one.

Death Of A Salesman

When the 520ST first started appearing in Toys "R" Us stores around the country, **Atari Chairman Jack Tramiel** and an entourage of other Atari execs dropped in unannounced at an outlet near Sunnyvale and posed as customers to see how their computer was being received. But the surprise visit **turned sour** when an unknowing salesman tried to **steer Tramiel away** from the 520ST toward a Commodore 128. Jack was not amused.

Beta Blockers

Software developers use special codewords for prerelease versions of their programs: An alpha copy is the earliest version that functions at all, and a beta copy is a nearly finished version that needs further testing and debugging. Customarily, software developers distribute beta copies to beta testers—trusted people outside the company who test the program and file reports of any bugs they find. But recently some ST software developers have quietly started

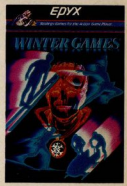
WINTER GAMES.™
THE QUEST FOR THE GOLD CONTINUES...

You've captured the gold in Summer Games® and Summer Games II™. Now it's on to the *Winter Games*! And what an incredible setting—a completely realistic winter wonderland featuring *seven* action-packed events.

At the Ski Jump you control your form in mid-air, knees straight, leaning forward. Hot Dog Aerials challenges your courage *and* your sense of humor. In Figure Skating you leap into Double and Triple Lutz jumps—wow the crowd with a perfect Camel into a Sit Spin. It's timing and style that counts. Free Skating lets you choreograph your own routines. In Speed Skating it's you against a fellow speed demon—the fastest human beings on level earth! And the Bobsled—still faster as you fly around hairpin turns, leaning hard to stay in the tube. Finally the Biathlon, the ultimate challenge to your endurance in cross-country skiing and marksmanship.

All of this fun and excitement is easy to learn and play. You control the action with the joystick, animating your player for style and rhythm. You choose the country you want to represent. Listen to its national anthem. Then it's practice, training and learning a winning strategy for each event. Now the *Opening Ceremony* and the competition begins—against your friends or the computer. Will you be the one who takes the gold at the *Awards Ceremony*? Will your name be etched amongst the *World Record* holders?

The quest for the gold continues... And it's all here—the strategy, the challenge, the competition, and *pageantry* of Winter Games!



Available for Atari ST.

EPYX® COMPUTER SOFTWARE

1043 Kiel Ct., Sunnyvale, CA 94089

Strategy Games for the Action-Game Player®

moving all beta testing in-house. Why? Because **some unscrupulous beta testers** have been making copies of the programs for friends, and the undebugged versions are filtering through the user community and causing **all kinds of problems.** The in-house beta testing often takes longer, but apparently the feeling among these software developers is **beta late than never.**

Unable To Cable

One of the largest manufacturers and suppliers of computer cables—a Dayton, Ohio–based outfit called Cables-To-Go—is **stymied** by the Atari ST. Since the ST has IBM PC–compatible modem and printer ports, those cables are commonly available. But a few months ago, people started asking Cables-To-Go for a video cable to connect an Atari ST to a Sony KV-1311CR. The KV-1311CR is a do-everything TV/monitor that offers remote control, analog RGB, digital RGB, composite video, and 105 cable-ready channels for about $600. Very popular among Amiga users, the KV-1311CR would work with the ST, too, with the right cable. Cables-To-Go figured it would be **no problem** to whip one up in a week or so.

Guess again. First they had trouble locating a source for the **weird 13-pin DIN connectors** that Atari uses for video output on the ST. Then, when they finally found a supplier for the connectors, the prototype cables didn't work as expected. Oddly, the ST's video and audio signals are both incompatible with the monitor.

At this point, **Sony itself stepped in** to lend Cables-To-Go a hand. Sony engineers bought a 1040ST and have been **struggling for weeks** to make a cable that works with their monitor, yet is economical to manufacture. Apparently the cable will have to include some **extra circuitry** to avoid requiring modifications to the monitor. Sony is anxious to perfect the design because it plans to order **at least 1,000** of the cables from Cables-To-Go. Why so many? Because Sony wants to distribute the cables to its nationwide sales force, which will then try to **entice Atari dealers** into carrying the KV-1311CR as an alternative to the Atari SC1224 color monitor.

New ST BASIC?

There's definitely going to be a new and improved version of ST BASIC. The only question is **when and how much.** The latest rumor is that we can expect something this fall.

The new version is being written by **Metacomco,** the British company that wrote the original BASIC that Digital Research and Atari eventually turned into ST BASIC. (Metacomco also wrote AmigaDOS and Amiga ABasiC, the language that was quickly replaced by Microsoft's Amiga BASIC soon after the Amiga was released.) Supposedly, the new ST BASIC will have greater support for GEM and VDI calls, but **may not be completely compatible** with the current ST BASIC. This shouldn't be a problem, though, because relatively few programs have been written in ST BASIC anyway.

Ahl's Well That Ends Well

Atari has contracted with **David Ahl** and **Betsy Staples** to put together its in-house magazine, the *Atari Explorer*. Ahl founded *Creative Computing* in the mid-1970s and remained the magazine's editor until it folded in 1985. Staples is his long-time associate. Both will remain based in New Jersey and edit the *Atari Explorer* from there. They're taking over the bulk of the *Explorer* workload from Atarian **Neil Harris**, who used to help edit Commodore's in-house magazines until he followed Jack Tramiel to the new Atari in 1985.

Quick Blit

Atari has said that current 520STs and 1040STs can be **upgraded** with the long-awaited **blitter chip** when it becomes available later this year. (The chip is designed to speed up screen graphics and must be installed inside the computer.) But a highly credible source at Atari is **skeptical.** Not only are there no empty sockets for the blitter in current machines, he points out; there also **isn't any room** to squeeze in a daughterboard. We hope our source is wrong on this one, but he comments, "Remember all those VIC and 64 upgrades that Commodore owners got?"

The Volkscomputer

You've probably heard that about half of the ST's installed base is in Europe. But did you know that for the past year, more STs have been sold in **West Germany alone** than in the entire United States? One reason is that the ST was introduced in Europe a few months before it first appeared in the U.S., because Atari had to wait for FCC approval. Another reason is that a significant portion of Commodore's traditionally strong dealer network in Europe **switched to Atari** when Jack Tramiel took over.

On The Whims Of Eagles?

Just what is **Electronic Data Systems** (General Motors' data processing division) doing with the four Atari 1040STs they bought this summer? If you've got a handle on **what EDS is up to** with their STs, drop us a line.

Whisper To The Ear

Got something you want to get off your chest? ***The Ear wants to hear.*** *Mail missives to The Ear, c/o COMPUTE!'s Atari ST Disk & Magazine, P.O. Box 5406, Greensboro, NC 27403. Or send electronic mail to CompuServe ID 70675,463, Delphi ARLANL, or The Source TCT987. All sources treated confidentially.* **ST**

COMPUTE!'s Atari ST DISK & MAGAZINE

BIG BONUS! 7 ISSUES FOR THE PRICE OF 6

Say "YES" now to COMPUTE!'s Atari ST Disk & Magazine, and get your first issue—FREE.*

If you're one of the tens of thousands who've recently purchased an Atari ST, *here's great news!*

Introducing COMPUTE!'s Atari ST Disk & Magazine—an exciting new publication *devoted exclusively* to the special needs and interests of Atari ST users like you. And this special FREE offer is a great way to get acquainted.

Every other month, COMPUTE!'s Atari ST Disk & Magazine brings you exciting new action-packed programs already on disk! Each is ready to run. Just load the disk and you're ready to go.

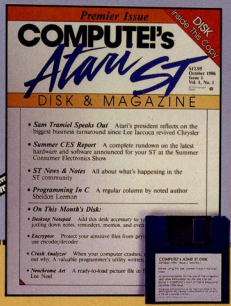
You can depend on getting at least five new programs in each issue—high-quality educational, home finance, utility, and game programs you and the entire family will use, enjoy, and profit from all year long.

*With your paid subscription.

NEW!

Fact is, the fantastic programs alone are worth much, much more than the low COMPUTE!'s Atari ST Disk & Magazine subscription price. But exciting new programs are just the very beginning of what you'll get as a COMPUTE!'s Atari ST Disk & Magazine subscriber.

Has a hot new piece of software suddenly hit the market? Turn to the magazine section and find out fast—and first. Our up-to-the-minute software reviews are timely and complete. And you can depend on each issue to deliver the latest inside word on everything from interfaces to languages...programming to disk drives. If it has anything to do with your Atari ST, you'll find out right here in the pages of COMPUTE!'s Atari ST Disk & Magazine...the definitive guide to your new Atari ST computer.



So subscribe today and take advantage of our special FREE SOFTWARE AND MAGAZINE offer. Sign up now and we'll give you your first issue of the disk and magazine—absolutely free.* It's yours without charge as our way of saying "Welcome Aboard."

But hurry. A deal this special can't last forever. So return the attached reply card now!

Mail coupon today or call toll-free

1-800-247-5470

In Iowa

1-800-532-1272

COMPUTE!'s Atari ST Disk & Magazine

A **COMPUTE!** Publication

CHARTER SUBSCRIPTION OFFER

☐ **YES!** Sign me up for this no-risk offer. Send me my first issue of the disk and magazine FREE of charge—as my special "Welcome Aboard" bonus. Then enter my charter subscription for six MORE issues (a full year's subscription) at the special introductory price of just $59.95. I save more than $17 off the newsstand price.

Mail today to:
COMPUTE!'s Atari ST Disk & Magazine
P.O. Box 10775
Des Moines, IA 50340

☐ Payment Enclosed ☐ Bill My VISA/MasterCard

Credit Card #_____ Exp. Date_____

Signature_____

Name_____

Address_____

City_____ State_____ Zip_____

Outside U.S.A., please add $6 (U.S.) per year for postage.

Sam Tramiel Speaks Out

Tom R. Halfhill, Editor
Selby Bateman, Features Editor

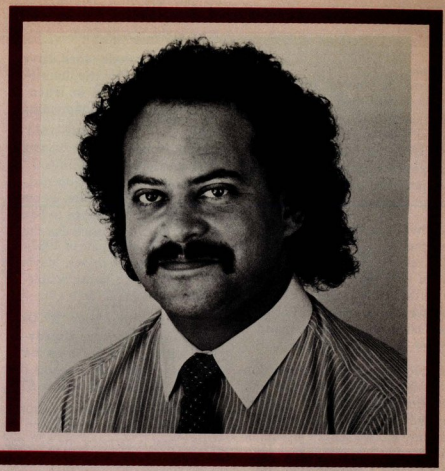
In the summer of 1984, Sam Tramiel followed his father—Commodore founder Jack Tramiel—to Sunnyvale, California to become president of the new Atari Corporation. The senior Tramiel had just purchased Atari from Warner Communications in a last-ditch attempt to rescue the beleaguered company from impending bankruptcy. Another Tramiel son, Leonard, became vice president for software development. After two years, this management team has not only turned Atari around, but has also helped to inject new excitement into the industry with its ST line of personal computers. This interview was conducted on June 3 in Atari's booth at the Summer Consumer Electronics Show in Chicago.

Q: Tell us what you consider to be some of the key factors to the success of Atari over the past year—the things which have put you in the position you are in now with the ST.

Tramiel: OK. As you know, we took over Atari in July 1984. We knew that Atari was in a mess, but we had no idea of the extent that mess was. It took us until almost the end of 1985 to get the mess turned around. So by the end of 1985, we had turned the company around, we had reorganized, and we almost set the company up into a number of different divisions. One group of guys was cleaning up the old problems—cleaning up the accounts, getting everything reorganized, getting the overheads down, getting everything in place. Then we had a group of engineers, as you know, who started working on the new products. They started even before we took over Atari. They started really in June, conceptualizing how the ST should be designed. My brother Leonard [vice president for software development], Shiraz Shivji [vice president for research and development], John Feagans [director of software development]—we all sat down and said, "OK, the ST will do the following things." We met with DRI [Digital Research Inc.] to discuss what would be the best interface for our operating system, and everyone agreed wholeheartedly which way we should go. We made a very aggressive engineering plan, and at the January [Consumer Electronics] show of 1985, bang! We had the prototypes.

It was amazing how four custom chips at one time came out, and every single one of them worked. We were just over the moon. Four custom gate-arrays in January came out. Two days before the show, Shiraz was saying, "I hope they work." And they all worked.

The operating system took us a while to finish off. That was finished in April–May of 1985, and



we started shipping to Europe. And then we got the FCC [Federal Communications Commission] approvals by July, and started shipping in the U.S.

So, separating the company from the old and the new was the key. Not that the old problems slowed down the new developments.

Q: Have you been surprised at the extent of the interest—how quickly it's come, the software, the users? It seems as if the extent of the interest has been even greater than you anticipated.

Tramiel: Well, to be very frank, we anticipated the interest to be very high. We felt the market had not gotten a new machine at the right price since the Commodore 64 came out [in late 1982]. There was nothing that was really that exciting. The Macintosh—yes, a very nice machine, conceptually beautiful—but overpriced for the average man. And the ST was to get that new technology at the price for the average man. That's been our whole philosophy.

Q: Why has the ST sold so well in Europe? The European market accounts for over 50 percent of the STs.

Tramiel: I'm not sure why. We have ideas why. We have a very good team over there of men, old friends from our old company who know the marketplace very well. And I think the European dealers who are carrying the ST know value. They go into an office, let's say, in Germany or Switzerland, and they show the buyer in the office, "Why buy 10 IBM PCs when you can buy 20 STs and achieve the same solution?" It's a much better machine for a lower price. And they go for it.

Q: Do you anticipate increasing a move toward parity with U.S. sales?

Tramiel: That's our plan. We're pushing the U.S. very hard this year. We want to get the U.S. up and going. We have a promotion this summer which should do well on the ST: That's a monochrome system complete with the floppy drive, monitor, and CPU [central processing unit] for under $599, for the 520ST. That should get the marketplace rolling bigger in the U.S.

Q: What is your estimate of the installed base now, here and in Europe, and what do you think it will be by the end of 1986?

Tramiel: We don't publish country-by-country numbers. We never have and we don't plan to. But on the ST worldwide, it's now over 200,000 machines. And we hope that before the end of 1986 to be around 500,000 machines. That's our target, and that's a conservative target.

Q: Who do you think is buying the ST so far?

Tramiel: For our market study we had some focus groups that we studied. Most of them are

second-time computer owners. They own a Commodore 64, an Atari 800, or an Apple II, and they want to upgrade to the new technology. That, I think, has been our first buyers. I think as people realize that it's such an easy machine to use, then of course first-time buyers will get involved with it also. And that's our whole thrust this year: ease of use, power, and for a very low, low price.

Q: There seems to have been, a few months ago, a real scarcity of machines in the U.S. We wondered why there was this scarcity at that time, and were some units being diverted to Europe because of the strong demand there?

Tramiel: We plan our production according to what we think our requirements are. In the U.S. and around the world, we're very careful who we sell the machine to. The retailers are all shaking out—who's a good retailer and who's a bad retailer? So we're very careful who we sell to and how many we will sell to them. We will not stuff machines into the stores. We don't want them to be over-inventoried. So they have a lean inventory, and they get the machines and sell them right away. That's why there's a scarcity. We're giving a little bit more now to each dealer so they have more inventory.

Q: You've established some pretty complete requirements for dealers who want to carry the 1040ST. Is that part of your new philosophy?

Tramiel: Yes. We're trying to build up a very strong dealer base—a sophisticated dealer base. They have to be a servicing dealer who knows how to fix an ST. We know that after the 1040 there will be a two-meg machine, a four-meg machine, Unix-based 68020s next year, and you can't sell that through Toys "R" Us. You have to have sophisticated dealers. So we're trying to build up that base now, starting with the 1040. Slowly, but very strong. It's a good, strong base.

Q: Will the single-sided drives be superseded by the double-sided drives, and if so, when do you anticipate that happening?

Tramiel: Yes. The single-sided drive now has a price advantage over the double-sided drive. I would assume, as it happened with the 5¼-inch drives, as time goes on they will merge to be almost the same in price. For the year of 1986 and part of 1987, there will be two drives. Maybe in late 1987 it'll go to the double-sided drives only.

Q: We know that Toys "R" Us and Target stores are already in line with you in the mass market. Any other major chains that you anticipate carrying the ST?

Tramiel: There are some major regional chains. Lechmere's on the East Coast is carrying the 520 and doing very, very well with it. Federated on the West Coast is carrying it and doing well with it. Also in Texas. We're also being careful which mass merchandisers will carry it. I had a meeting today with Zayre. The question is, can a Zayre support a 520? They're not sure and we're not sure. So instead of going full blast, we're going to take a small marketplace and study it.

Q: Do you think that's key to the approach of avoiding the kind of problems that happened two or three years ago, when home computers were put into stores and customers couldn't get adequate support or even just basic guidance?

Tramiel: I think that the mass merchandiser has its place, and so does the specialty store. The mass merchandiser allows a machine to be sold at a very low price, and a lot of people who already know how to use a computer can just go there and buy it at that price. The support that the specialist dealer affords is important to people who want the support. Even the guy who buys the low-priced machine from the mass merchandiser can go to the specialist for second-time buying: for software support, peripheral support, and getting all the backup he needs. So there's a symbiotic relationship between the two of them.

Q: You and other people at Atari have already talked about the different generations of STs. What's the next step, and how do you see the progression in the future?

Tramiel: First there's the ST line, which will be going up in RAM [random access memory] as RAM technology and affordability increases. One-meg RAM should be big in 1986, and therefore we will be introducing a two-meg and a four-meg machine later in 1986—probably the late quarter in '86. After that we'll be having enhanced graphics on the ST. Internally we call it the EST—E for Enhanced. It will be software-compatible with the ST, an upward growth path, and just have basically higher-resolution monochrome and color screens, aimed more at the workstation markets. It will probably be more expensive than the ST today, but aimed for the higher end of the personal computer market. And then we'll go into the 68020 market, which will have Unix-based software, and your STs will be able to be terminals to the 32-bit workstations. So the ST line will be going up and up and up in that way.

Q: You mentioned two- and four-megabyte STs. Will there be any way possible to upgrade a 520ST or 1040ST to that much memory?

Tramiel: No. We at Atari have no plans for that. Some entrepreneurs outside, I'm sure, will do some kind of boards to let people do that. People outside are already upgrading 520s to one meg, and they'll figure something out.

Q: Will the 68020 Unix-based machines have expansion slots and so forth?

Tramiel: Yes. Absolutely. We're designing two new housings for the ST and EST machines which will be more like the conventional IBM-looking PCs—a detachable keyboard, a bigger box in the back—for a different part of the marketplace.

Also, before I forget, there's a blit [bit-block transfer] chip that Shiraz mentioned in one of his interviews. The blit chip is almost finished. We're making another pass at it. When the blit is ready, your 520 and your 1040 will be upgradable with the blitter. We're working that out. There will be no problem for both STs to take the blit.

Q: How will that work as far as installation?

Tramiel: That will be a dealer upgrade. You'll go to your dealer and he'll plug it into your machine.

Q: How about the status of the Amy sound chip?

Tramiel: When we took over Atari, Atari had spent $3 million so far on the Amy chip. We did, I think, four passes on the Amy. Couldn't get it to work. It's an extremely complicated chip, but when it works, it will be phenomenal. So we now have an outside company working on it with us, and NCR is the fab[rication] house, and we hope the next pass will work. We just don't know. When it does work, then bang! We'll be starting to use it. It's going to be a dynamite product when it comes out.

Q: It will be upgradable to previous machines?

Tramiel: We definitely will make an ST with it inside, later on, but on the present ST it will be a cartridge that will plug into all the existing STs.

Q: What about the eight-bit Atari computers? Will it plug into those?

Tramiel: We have to discuss it. Perhaps. I don't see why it couldn't work. But on the STs for sure, it will plug into existing STs.

Q: When do you foresee CD-ROM [Compact Disc–Read Only Memory] and CD-I [Compact Disc–Interactive] players coming out for the ST, and will they be from Atari?

Tramiel: CD-ROM, as you know, we showed it here one year ago [at the 1985 Summer Consumer Electronics Show]. Now the software is ready; the interface is ready; everything is ready to go from a technology point of view. But we don't want to sell it for a high price. We think it's better if we sell it for under $500. We're negotiating with our friends in Asia to get the price down to that point. With the recent yen revaluation, it becomes more and more difficult. We think that if a CD-audio player retails for about $200 today, you can't expect the public to pay $700 or $800 for a CD-ROM. It just doesn't make sense.

Now on the CD-I, the CD-ROM that we have is a subset of the CD-I product. When CD-I comes out, we will be into that as well. But the CD-I requires a brand-new TV standard, and we think that will be at least three years away. CD-I requires a whole new technology which is just starting to be worked on. Sony and Philips are the main pushers behind this, and they haven't even finalized the CD-I specs yet. But it will be coming. The whole CD-ROM thing I find extremely exciting. When you see it work, it just blows your mind. It's amazing, the speed of getting this data right at your fingertips.

Q: We've heard rumors that the 520ST could be selling for as low as $200 by Christmas. What do you think about that?

Tramiel: Well, there could be certain promotions in a way that dealers and retailers could package the product, the system. With our summer special on a monochrome system, our suggested retail is $599 for that system, for this summer. The retailer could, perhaps, take the 520 and say, "Buy this at $199, and then buy the floppy drive and the monochrome monitor at regular retail—that's $200 each—and it's $599." So it depends how they want to package the whole ball game.

Q: What would you say is the future status of the eight-bit Atari line?

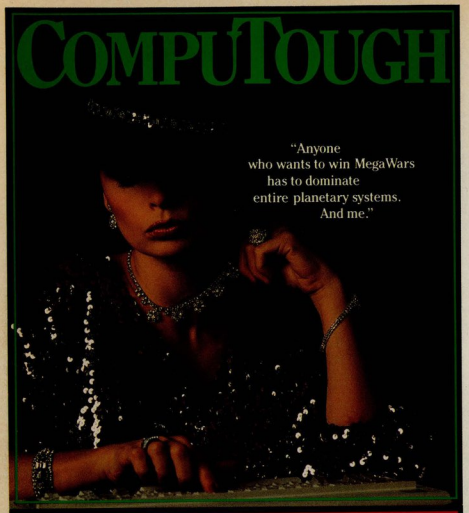
Tramiel: From this show, we're very excited. Our old company raised their price on the [Commodore] 64. They're recommending a retail now of $199. We feel the eight-bit market is very alive; it's a budget, entry-priced computer. From our meetings that we've had with the major retailers here, our 65XE is getting back into the market now. Everyone is very excited about it. It's $99; you get a lot of software with it—I think it's four titles—and it's a strong machine. Technically, it's just as good as a Commodore 64, if not better. With these price variances now, we feel the eight-bit line will come back to life.

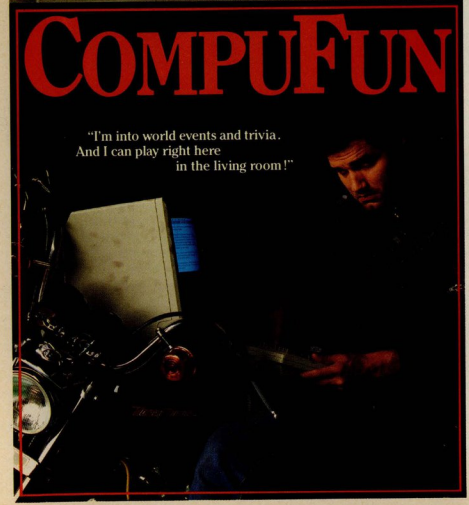
Q: Do you see Atari ever getting into major promotions in educational institutions and schools, as Apple has done?

Tramiel: Yes. We are pushing that very hard. We just did a deal two days ago for the East Coast. We hired a man who is an educational expert who is going after that. And Sig Hartmann [president, Atari Software] just closed a deal with a company called Arrakis [Montreal], and they're doing 17 titles—educational—for the ST. A company called Computer Curriculum Corp. [Palo Alto, California] is carrying the ST—the ST is their terminal for their products.

Q: How satisfied are you so far with third-party software support for the ST?

Tramiel: In the beginning we were not so satisfied because they were slow, but things are now coming out at a very good clip. And we have now, worldwide, about 1,500 software developers,





and every week more and more stuff comes out.

Q: What do you feel is your weakest category of software support? If you were going to advise a developer to work in a certain area, what would you advise?

Tramiel: Right now I think that we're weak in spreadsheets—we don't have enough good spreadsheets—and also the high-end word-processing market. But really, the machine is so strong, in the monochrome mode it's a wonderful word processor with fantastic resolution. So that's why I think we should be getting more support on that.

Q: What kind of future do you see the ST having in desktop publishing?

Tramiel: My brother Leonard and Shiraz are working on our answer to the [Apple] Laserwriter. We're working with a number of Japanese hardware firms, and we're now working internally on the software. The desktop publishing business is very big. Apple pioneered it, which we thank them for, and we want to get into it in a big way. Hippopotamus [Software, Los Gatos, California], as you probably know, has a driver for the Canon laser printer, the Hewlett-Packard laser printer, and it's just a great market to go after. We're definitely going into it.

Q: So Atari is making its own laser printer rather than depending upon a third-party company to do it?

Tramiel: Correct. We're definitely doing our own hardware/software combination. Again, we think that's the best way to get the price to the right point.

Q: Do you ever foresee yourself pushing the ST in the business market head-to-head against IBM, as Apple tried to do with the Macintosh?

Tramiel: Yesterday there was a conference here at CES talking about home office products. I was

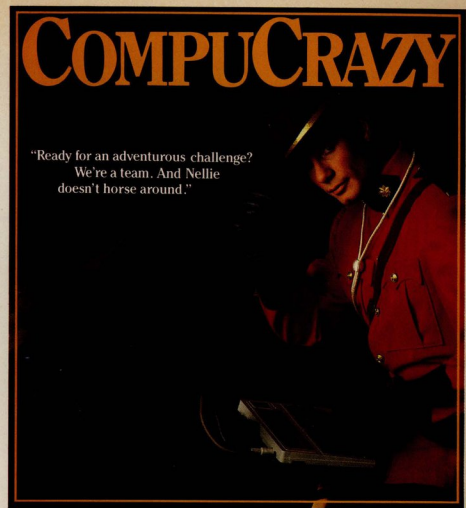
asked, "What do you think about IBM, and how are you going to fight against IBM?" While we were at Commodore, and now that we're at Atari, we're not going to fight against IBM. We pass. We're not going to do it against them. They can smother you. So we're not going after the Fortune 500; we're going after the small businessman—the man who himself is paying the money out of his pocket to buy the computer for his store or his office. IBM goes after the corporate buyer, the buyer who wants to be protected, who wants to have his life very easy, and who's not spending his own money. They can have that market. But we're definitely going after the lower end of the office/business market.

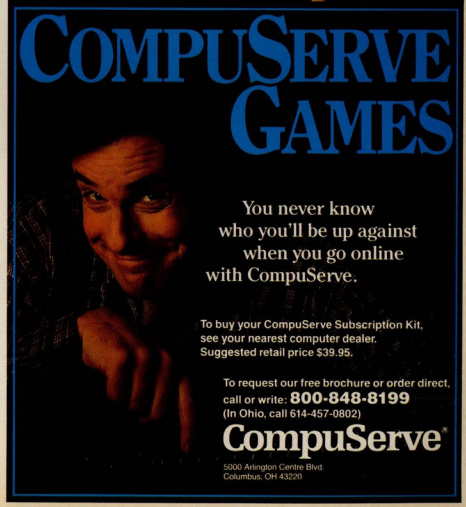
Q: What can you tell us about your IBM emulator, and how does that fit into this picture?

Tramiel: Many people feel uncomfortable—they have to use IBM software. They have the IBM machines in their office at work, and when they come home they want to be able to use that software. So to answer that problem, we're not going to make the ST into an IBM PC, but with the emulator you'll be able to use most of the heavy IBM software. There's a common Big Ten list which we got—the Lotuses, Ashton-Tate products, the Microsoft products. We will run all that software. It will be working in the most popular monochrome and color modes. There will not be slots, but it will be a good emulator. It will run at IBM speeds; it will not be a slow emulator, because it's being done in hardware.

Q: What kinds of specs will it have in terms of built-in memory and so forth?

Tramiel: 512K of RAM inside; a socket, I think, for the 8087 math processor; and the 512K RAM, when not being used with the emulator, can be a RAM disk for the ST. So you can have an ST running with a 20-meg hard disk, a





one-meg or 500K floppy, and a 512K RAM disk—a phenomenal array of outside memory.

Q: What kind of price are you looking at for the IBM emulator?

Tramiel: It depends on whether it will be with or without a drive, but $200 to $300 is the price range right now we're talking about. We're making it to give the user a bridge to the IBM software world, not a pure 100-percent emulation. But it will run a lot of the software.

Q: And you think it will be available at retail this fall?

Tramiel: Yes. We hope it will be finished by August—let's say September—and start shipping around that time.

Q: Of course, you know that the Amiga also will have its Sidecar, an IBM emulator. What do you think has been Commodore's biggest problem since Jack Tramiel left Commodore, especially in terms of the Amiga?

Tramiel: Well, we think there's a very simple answer. The new Commodore is like the old Atari, and the new Atari is like the old Commodore. Our philosophy is that if a product costs us $X, we make a fair profit on it and charge that to the consumer. Commodore and other companies look at the market and say, "Ah! This product can get $X thousands of dollars because that's the perceived value, so that's what we'll sell it at." We want to be more fair. We offer the technology at the affordable price. And the Amiga is just too high-priced.

Q: If you were selling the Amiga, what would its price be?

Tramiel: Around the price of an ST.

Q: With the increased sales and cash flow you've got now, do you think that Atari is out of the woods financially?

Tramiel: Absolutely. We have been out of the woods for quite a while now. In 1985 we were profitable in our ongoing business, but there was a lot of old stuff to clean up. In 1986, we're just plain profitable—old and new, we're nicely profitable. We were very pleased with our first-quarter results.

Q: Do you foresee a time when Atari might go public?

Tramiel: There might be a time. We have no firm plans at the present time. We're 100-percent internally financed, positive cash flow. We're very happy; very happy.

Q: When we were out at Atari about a year ago, we noticed it was sparse in terms of employees. Have you started adding more people, expanding your research and development teams?

Tramiel: Well, we don't think we're sparse. That's the way a company should be run. We believe in having a low overhead and running things very cleanly, not having layers of management. Where we are investing our money is, yes, in R & D, and now also, more in marketing. We've got to get the ST in front of the public's eyes, and our eight-bit line as well. I don't know if you've seen our videogame advertising—the TV ads for that. All three categories we carry—the videogames, the eight-bits, and the ST—are all doing very well. We will be advertising all three lines very heavily this year. So more money will go into that, and the R & D section, yes. That's always been our philosophy, to spend a lot of money in R & D, so that we'll be building up as time moves along.

Q: Will the 68020 Unix-based system be the TT that we've heard about?

Tramiel: That's our internal code. We might call it the TT, for Thirty-two Thirty-two. We haven't decided on a name yet, but that's the machine we're talking about, the TT.

Q: We've heard that you're trying to get a 1024 × 1024 screen resolution, but that there are no monitors inexpensive enough to make that possible.

Tramiel: I know. We have spoken to our monitor manufacturers, and they are now figuring out ways to bring that price down. Our Unix-based machine will have very good resolution. Our EST will have the same good resolution, and the prices are coming down on the monitors right now.

Q: Resolution greater than what the ST has now?

Tramiel: Yes. Workstation-like resolution. It will be like Daisy [a high-end computer-aided design workstation] kind of quality—very high resolution.

Q: Where do you see consumer computing going five years from now?

Tramiel: Shiraz could answer you better, but in our opinion, from the microprocessor point of view, the 68020, we think, is just about it. After that will be 64-bit micros, but that's a while away. The 68020 32-bit processor is incredibly fast—a lot of data going through. And with RAM going further and further ahead, there will be incredibly good software which will be very, very, very, very easy to use. It will be almost like an appliance. You won't have to think how to make the machine work; it'll just work. You'll push a button and things will happen for you. The future computers will be so simple to use, it'll be amazing. Also, five years from now, there'll probably be a lot more voice interaction—voice recognition, voice coming at you. Things like that will be happening.

Q: And where do you see Atari five years from now?

Tramiel: Into that. Our plans are to be the biggest PC company in the world. At Commodore we achieved that, and we'll do it again at Atari. That is our target. **ST**

The Comeback of Atari

Tom R. Halfhill, Editor

A revitalized Atari Corporation is rapidly gaining the confidence of product developers and industry observers, and it's starting to show in the flood of new products being released. Here's an analysis of Atari's status as perceived in the summer of 1986.

By this summer there was little doubt: Not only is the Atari Corporation back on its financial feet, but the Atari ST is here to stay. Just a year after its introduction, the ST is rapidly attracting support from hardware and software developers both here and abroad and is accumulating a full-fledged library of quality programs.

These impressions were reinforced when the new Atari made its strongest appearances ever at two crucial industry trade shows in May and June: the Spring Computer Dealer Exposition (COMDEX) and the Summer Consumer Electronics Show (CES). Both shows are primarily intended to help manufacturers attract new dealers. To demonstrate the growing support for its computers, Atari subdivided its large exhibit areas into small booths which were rented to dozens of independent hardware and software developers. Since many of these developers are relatively small companies, they had a rare chance to demonstrate their products at a pair of shows where booth space normally costs thousands of dollars for a few square feet.

It resembled a frantic flea market. From morning to evening, the developers met with thousands of potential new dealers and curious onlookers. It's apparent that at the one-year mark, the Atari ST has a greater volume and wider variety of software than was available for any other computer after a similar period.

What's on the way? A veritable flood. The new products span virtually every category of hardware and software: home, education, small business, and entertainment. Word processors, spreadsheets, desk accessories, utilities, database managers, programming languages, drawing programs, animation creators, music editors, games, simulations, educational programs, sound digitizers, video digitizers, MIDI controllers, sequencers, operating system emulators—you name it; you'll see it.

Atari's corporate survival—a matter of grave speculation just a year ago—is no longer the central issue. Already, the company is setting its sights on widening the U.S. market for the ST and, in 1987, introducing an even more powerful series of 32-bit computers. The new Atari may be the biggest corporate turnaround story since Lee Iaccoca revived Chrysler.

Atari's announcements in the first half of 1986 reveal the various directions in which the company is moving. The top priority seems to be a steady stream of enhancements to the ST line. Another priority is the maintenance of the eight-bit line as an alternative for those who want to get into personal computing at the lowest possible cost. Meanwhile, Atari engineers are working on the next generation of Atari computers.

The introduction of the one-megabyte 1040ST in early 1986 caught many observers by surprise. The half-megabyte 520ST was already considered a very powerful machine, and it was just gaining a foothold when the 1040ST was announced. By mid-1986, Atari was already talking about two-megabyte and four-megabyte STs by the end of the year.

Another early 1986 development was the switch from a RAM-based to a ROM-based operating system. The first 520STs required users to load the TOS operating system into RAM whenever the computer was switched on. Besides wasting time, it also consumed about half of the machine's available memory. When TOS was squeezed into ROM, more than 200K of RAM was freed up for application programs. In addition to building TOS into ROM on late-model 520STs and 1040STs, Atari sold the ROM chips at nominal cost to owners of early-model 520STs.

Other enhancements planned for 1986 include a blitter chip—which will speed up the ST by accelerating the screen display hardware—and the Amy sound chip. Amy is an unfinished project of the old Atari that has caused the new Atari a lot of headaches, but which will be worth the trouble if perfected. Those who have heard Amy, claim it can simulate musical instruments like a professional-quality digital synthesizer. If the engineers can get the bugs out, Amy will be built into newer

STs and sold as a plug-in cartridge for older STs.

In effect, all of these upgrades represent Atari's efforts to continue design of the ST even after it has left the labs. The ST was put together from scratch in the incredibly short period of about six months after Jack Tramiel bought the remnants of Atari from Warner Communications in July 1984. Six months isn't much time to design a computer, and Atari had to leave some things out. When the ST was introduced, it was immediately compared to the Macintosh and Commodore Amiga—computers which had benefitted from years of development time. The blitter chip and Amy are clearly aimed at the Amiga, which boasts superior graphics and sound.

According to both official and unofficial sources, the next generation of Atari computers will attempt to go a step beyond. Watch for a machine with a 32-bit 68020 microprocessor, two to four megabytes of memory as a standard feature, built-in hard and floppy disk drives, monochrome screen resolution of 1024 × 1024 pixels, multitasking, and compatibility with both ST and MS-DOS software.

Perhaps the most unexpected trend—at least for those familiar with Atari's history—is the emphasis on small business/productivity software for the ST. When the first-generation Atari home computers hit the market in the late 1970s and early 1980s, most of the programs released for them were games. The top-notch word processors and productivity programs didn't come until later. As a result, the eight-bit Ataris were dismissed by many people as fancy game machines, despite their overall technical excellence. But it's just the opposite with the ST series. So far there are only a handful of arcade-style games, yet scores of more serious programs.

One reason may be that the Atari ST delivers a lot of powerful hardware for the money—more than many personal computers now installed on office desktops. Software developers hope the power and low prices will lure some small-business customers to the ST.

The hardware half of the equation is there. At a time when many industry observers are excited by the prospect of $500 IBM PC–compatibles with 128K or 256K of RAM, Atari has been offering a $599 summer promotion for a much faster, 16/32-bit computer with 512K of memory; a fast, compact, 360K floppy disk drive; and a monochrome monitor that has the sharpest, highest-resolution screen in the industry. For $999, Atari is selling another version of the computer with a full megabyte of memory, a 720K floppy drive, and the same monochrome monitor. Furthermore, both the 520ST and 1040ST come equipped with features that are usually extra-cost options on other machines: RS-232 serial port, parallel printer port, monochrome video output, RGB color graphics output with 512 colors, hard disk interface, and even a pair of Musical Instrument Digital Interface (MIDI) ports.

With its desktop-style user interface and provision for memory-resident desk accessories, the ST compares favorably to a PC-AT running such software as *Microsoft Windows* and *Sidekick*. It's no wonder that developers are anxious to attract business people and so-called "power users."

From the start, few people have doubted the capabilities of the ST's hardware. The more important questions, however, have been Atari's financial stability and the availability of useful software.

The stability issue is raised less often now. Although Atari is hardly cash-rich, it doesn't appear to be drowning in debts, either. There are even rumors that Atari may go public with a stock offering next year. Under the leadership of Jack Tramiel and his sons Sam and Leonard, the new Atari is a very lean company that drives hard bargains with its Far East suppliers and keeps a close eye on expenses. One employee noted with a chuckle that Atari recently switched from two-color embossed business cards to single-color plain cards to save on printing costs.

One way Atari does not make money is by padding its retail prices with fat margins. Atari's "Power without the price" strategy is a major factor in the ST's success. Unfortunately, this strategy also hurts Atari's retail distribution, since it makes Atari computers less profitable for dealers than other brands.

Atari is trying to widen its distribution by releasing the 520ST to discount stores and other mass merchants, but most of these retailers backed away from computers after the 1982–83 home computer boom went bust. Furthermore, as even Atari admits, these stores aren't the ideal places to sell products as complex as the Atari ST. The computers are often locked in glass cases, inaccessible to browsers, and the salespeople know almost nothing about the machines. Also, these stores generally provide very little follow-up support—crucial for new computer owners.

To strengthen its network of traditional computer dealers, Atari is adding some new requirements for those who want to carry the more powerful 1040ST. Among other things, these dealers must maintain an actual storefront with on-site parking (thus eliminating part-time dealers who operate out of a home or post office box); an authorized repair facility with a specified inventory of spare parts; and a certain inventory of 1040STs and peripherals. This is one reason the 1040ST has been harder to find than the 520ST in most parts of the U.S.

Atari has two goals with this approach. First, it wants to sell more STs to small businesses and

power users. This means it has to provide the businesslike image and follow-up support that these kinds of customers demand. Second, the 32-bit computers that Atari plans to introduce in 1987 will be aimed squarely at the small business, workstation, and desktop publishing markets. This will put Atari in more direct competition with IBM and Apple, so it has to muster a dealer network to match.

Since the days when Jack Tramiel was running Commodore and beginning to sell personal computers, he has never enjoyed rosy relations with dealers. When the original Commodore PET was announced in 1977, Commodore took advance deposits and kept hundreds of people waiting for many months before the machines were actually available. In 1983, dealers were alienated again when Tramiel slashed the prices of the VIC-20 and Commodore 64 and shipped the computers to mass merchants. Although it's arguable that these moves helped launch the personal computer age as we know it today, it also suggests that dealer distribution is Atari's Achilles heel. Witness the fact that U.S. sales of the ST have been outpaced by sales in Europe, where Atari has a stronger dealer network. This is one major problem that Atari is still struggling to solve.

The software problem, on the other hand, appears to be solving itself. Although gaps remain, enough software is beginning to appear to satisfy the appetites of most hungry ST users. Still, there are three important developments to watch for as the ST achieves software maturity.

First, the ST must start attracting software from the big-name developers. Although technically there's no reason why small startup companies can't produce top-notch software—witness the spectacular growth of Borland International in the IBM market—the larger, more established companies lend an air of legitimacy that's valued by many business customers. Already, there are strong indications that this development is just around the corner—a major announcement was expected this summer.

Second, as more software appears, there will be a shakeout among developers. Like all new computers, the ST has accumulated its share of substandard programs, and the companies which can't deliver the higher-quality programs that users will increasingly demand will be out of business.

Third, somebody will come up with some software for the ST (or its 32-bit successor) that can't be done as effectively or maybe at all on any other personal computer. A likely candidate: affordable desktop publishing. Atari is pulling out all the stops to drastically cut the costs of laser printers, and the 1024 × 1024-pixel screens of Atari's upcoming computers will make it possible to lay out a full page of text and graphics on a single video screen.

If or when these three developments occur, Atari computers could join the same league as the IBM PC and Apple Macintosh in terms of solution-oriented software. And with their present price advantage, the Atari machines will be very serious competition.

ST

ATARI ST

ACTION PAK™

4 Atari ST™ Programs
1 Low Price!

39.95

DISK LABELER

Custom design your own unique 3.5 inch disk labels.

- Create professional-looking labels using three windows on screen. Fully mouse driven.
- Up to 38* files on each label, alphabetically, with monochrome monitor. Up to 18 files with color monitor.
- 35 borders and 4 type styles*.
- Includes 50 pin feed disk labels for 3.5 inch disks.
- Uses ST's full graphics capabilities*.

BANNER MAKER

Express yourself! Create original 1 to 4 line banners up to 72 characters in length.

- Includes 5 fonts. Custom design your own with Degas** or use any fonts compatible with Degas**.
- Left, right or center justification.
- Does *not* require "Action Pak (4 in 1)"** to print banners. Banner files can be given on disk or sent by modem.

TYPEWRITE

Great for addressing envelopes, filling out forms or writing short memos.

- Turns your ST into a line-at-a-time typewriter.
- Always handy! Resides in RAM as a desk-top accessory or runs as a stand alone program.

SYNFILE** CONVERTER

*A real work-saver! Convert your 8-bit Synfile** files fast without having to reenter data.*

- Creates ST files compatible with Hippo Simple**, Zoomracks**, DB Master One**, H & D Base**, DB Man**, dbOne**, or Regent Base**.
- Easy to use! Completely self-prompting. Just answer the simple questions.
- Requires modem or null modem.

Available at your local ST retailer. If you are unable to find it, then send coupon to: Action Software — 69 Clementina, Dept. AL, San Francisco, CA 94105. Please include $3.00 shipping & handling. Calif. residents please add 6.5% sales tax.

☐ Check enclosed Bill my credit card: ☐ VISA ☐ MC

Name _____

Address _____

City _____ State _____ Zip _____

Credit card no. _____ Exp. Date _____

Signature _____

We are looking for quality ST programs to publish. Call (415) 974-6638.

*Requires Epson** or Epson** compatible graphics printer. Other non-Epson** compatibles will print up to 18 file names, but may not accept graphics.

***Action Software, Atari Corp., Batteries Included, Synapse Software, Quickview Systems, Mirage Concepts, Versasoft Corp., Oxxi, Inc., Regent Software and Epson Corp. respectively.

The Pawn

Arthur Leyenberger

Requirements: Atari ST computer with disk drive and monochrome or color monitor.

One of the first games to take advantage of the ST's advanced graphics is Firebird's *The Pawn*, an illustrated text adventure created by Magnetic Scrolls, a British software group. It is a graphics-and-text adventure which is startling in both its sophistication and playability.

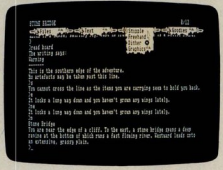
There's a lot more to *The Pawn* than just flashy graphics—it has one of the best parsers in the industry. (The parser is the portion of the program that interprets your typed commands.) If you've grown used to, or bored with, the parsers in other text adventures, you'll be amazed at the high quality of this parser.

Not A Timex

As the adventure opens, it seems that you've just awakened in the mythical land of Kerovnia with a mysterious wristband clamped to your forearm. You don't know anything about the wristband or even how you arrived in Kerovnia.

You're not the only one with troubles, though. Soon you learn that the king of Kerovnia is losing the loyalty of his subjects. The royal experts attribute this loss of devotion to King Erik's refusal to reinstate citizenship to the Roobikyoub dwarfs, who are thought to have been the instigators of the assassination of Queen Jendah II. The dwarfs have been banished by King Erik, but the facts of the assassination have never been discovered.

Kerovnia's economy once thrived as a result of the dwarfs' production of the strongest malt whiskey this side of Obakanga. But vested interests are now at work to prevent the return of the dwarfs. These forces have gone as far as to form the Farthington Real Ale company, which produces a refreshing spring water.



A typical text screen in The Pawn. *Note the scroll-shaped drop-down menus.*

One of The Pawn's *unique features: You can pull down a low-resolution graphics screen atop the medium-resolution text screen.*

Escaping From Nowhere

The land of Kerovnia is somewhere in the middle of nowhere, which makes sense within the context of the game. The object of *The Pawn* is to find a way to remove the mysterious wristband, because while it's attached, it prevents you from leaving the country. It is imperative that you leave the land of Kerovnia as soon as possible.

The program's parser understands relative phrasing in addition to more traditional noun-adjective-verb sequences. The parser is unique in its handling of objects. Objects may be referenced by location as well as name. Thanks to this powerful parser, complex sentences can be entered to let you interact realistically with *The Pawn*'s characters and objects.

Another attractive feature of the game is that drop-down menus, shaped like scrolls, are used to issue additional commands. Even more impressively, when you pull down on the top of the screen with the mouse, a low-resolution color picture lowers like a window shade on top of your medium-resolution text screen. This effect is one of the hallmarks of the Commodore Amiga, and it's a remarkable sight on the Atari ST.

Creating *The Pawn* was a massive project that took more than a year of development time. This effort shows in the game's quality, sense of humor, imaginative play, and stunning graphics. *The Pawn* is distributed in the U.S. by Firebird Software, an American division of Rainbird Software and a subsidiary of British Telecommunications PLC. The entire program is written in machine language rather than compiled C or Pascal, so it runs faster than most other text adventures. The graphics are dazzling, the plot is rich, and the implementation is top-notch.

The Pawn
Firebird Licensees
74 North Central Avenue
Ramsey, NJ 07446
$44.95

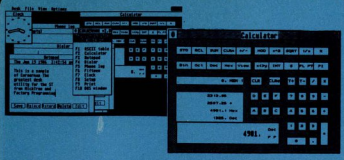
Personal Pascal

James W. Maki

Requirements: Atari ST computer with a disk drive, color or monochrome monitor, and the TOS operating system in ROM.

When the Atari 520ST first came out, I hesitated about buying it for two reasons: The TOS operating system was not yet in read only memory

FACTORY PROGRAMMING now for the ST – published exclusively by MichTron



CORNERMAN $49.95

What Sidekick did for the IBM, *Cornerman* does better for your Atari ST! This utility gives you a host of desk-top tools in one simple, neat package. With everything from a built in clock to a full function calculator, a cluttered desk is a thing of the past. As a Desktop Accessory, *Cornerman* is a available from most GEM applications.

- **REAL-TIME CLOCK** in digital or analog form.
- **NOTEPAD** lets you write and store notes for handy future reference and reminders.
- **ASCII TABLE** shows the ST's symbols with their decimal & hex values. Great for programmers!
- **PHONE BOOK** stores your names & numbers, dials through the modem and records elapsed calling time!
- **16 DIGIT CALCULATOR:** works in binary, octal, decimal, and hex; 3 memory registers; math and logic functions; base conversions; "tape" printer output.
- **DOS WINDOW** for access to MichTron's Dos Shell.
- **15-SQUARES GAME** simply for entertainment.
- Doesn't interfere with other programs.

MI-TERM $49.95

This smart-terminal modem program lets your computer communicate! Through your modem and *Mi-Term*, ordinary phone lines become a wealth of information. Contact computer bulletin boards and information services, or transfer files to other systems.

- **EASY TO USE,** with GEM's pull-down menus.
- **COMPLETE SUPPORT** for 300/1200/9600 baud.
- **26 REDEFINABLE PRESET KEYS** for automated log-ons and commands.
- **ON-LINE CHANGES** to parameters.
- **3 TRANSFER MODES:** Error free DFT, Xmoden, and ASCII file protocols.
- **AUTOMATIC BUFFER** captures and saves data.
- **CONNECT-TIME CLOCK** shows your calling time.
- **AND MUCH MORE!**

ALT $29.95

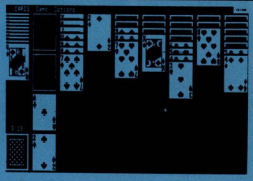
Typing all those reduntant phrases common in your daily forms and letters steals your energy and wastes your time. With MichTron's new *ALT* utility, you can convert whole strings of tedious typing into a single, elegant keystroke.

- **REPLACE ALL 36 ALT KEYS** with often used text phrases, program commands, etc.
- **UP TO 60 CHARACTERS** for each key.
- **SAVE YOUR KEY SET-UPS** to disk.

CALENDAR $29.95

This complete appointement and reminder calendar is the perfect addition to your GEM desktop.

- **DISPLAY CALENDERS** of any year and month to screen or printer.
- **STORE REMINDER MESSAGES** for any date and time, from 1980 to 2099.
- **EASY ACCESS** from almost anywhere in GEM.
- **SET "ALARMS"** to remind you of noted events! Messages appear from within almost any application.



CARDS $39.95

Take a refreshing break from arcade games without sacrificing the excitement. Enjoy 5 different computer card games. The rules are simple enough to learn quickly, but lively enough to become addictive. The detailed graphics and accurate rules make each game amazingly realistic.

- **SOLITARE:** This classic will boggle your mind!
- **KLONDIKE:** Stack red cards on black to uncover the buried aces (pictured).
- **POKER SQUARES:** Align the cards one at a time to build the best possible poker hand.
- **CRIBBAGE:** Test your strategy against the computer.
- **BLACK JACK:** It's four players against the house in this Vegas favorite.

All reasonably priced, with more coming every day. Ask for our latest catalog!

Dealer inquiries welcome • Visa and Mastercard accepted • Add $3.00 shipping and handling to each order.

MichTron

576 S. Telegraph, Pontiac, MI 48053
Orders and Information (313) 334-5700

(ROM), which meant that about half of the memory in the computer was unavailable for application programs; and there was no Pascal compiler available. I'd been using Borland International's *Turbo Pascal* on an eight-bit Atari with an ATR8000 in CP/M mode for three years, and I had grown accustomed to the speed and versatility of a high-level compiled language. When Optimized Systems Software (OSS) announced a Pascal compiler for the ST at about the same time that Atari announced shipment of TOS in ROM, I knew the time had come.

Over the past several years, OSS has earned the respect of the Atari eight-bit community with the release of excellent programs such as the *MAC/65* assembler, *ACTION!*, and *BASIC XL/XE*. With the addition of *Personal Pascal*, OSS now markets a high-level compiled language for the ST that promises speed, full access to routines in the Graphics Environment Manager (GEM), and more.

Ideal For Development

Personal Pascal comes on a disk without copy protection and is accompanied by a 276-page reference manual. If you're interested in writing programs for commercial or public domain distribution, you'll be pleased to learn that OSS permits such distribution without requiring fees or royalties. The only requirement is to sign and return the registration card and give OSS credit in the documentation or title screen of your program.

Personal Pascal adheres to the specifications established by the International Standards Organization (ISO), but offers extensions which make the language more useful to ST programmers. These powerful additions revolve around the support of GEM functions such as windows, dialog boxes, alert boxes, drop-down menus, and graphics. What's more, *Personal Pascal* works in both the GEM and TOS environments. It should be easy to port Pascal programs written on another computer to the ST for use in the TOS mode. Writing programs that support GEM is more difficult because many standard Pascal procedures (such as those for reading and writing) do not work with GEM.

Personal Pascal supports the scalar variable types Integer (range −32,767 to +32,767), Long_Integer (−2,147,483,647 to +2,147,483,647), Real (positive or negative, 1.0E−38 to 1.0E38), Char (ASCII characters), String (as an array of char), and Boolean (True/False).

The manual is concise and points out that beginners should purchase additional text to actually *learn* the Pascal language. The manual is logically structured to allow easy access to information; it is divided into six sections covering the text editor, compiler, linker, GEM and Pascal library routines, and a reference section. The largest portion of the manual (138 pages) is devoted to accessing the GEM/Pascal library and to managing events. GEM (and the programmer) must keep track of "events" such as window movement/redraw, mouse

```
program ListFile;
const
    {$I GEMCONST.PAS }
type
    {$I GEMTYPE.PAS }
    OneLine = string[80];
    Ascii = file of Text;
var
    PathName        : Path_Name;    { Default path name for selection box }
    i               : integer;
    m, n            : Long_Integer;
    LineOutput      : OneLine;      { Input variable from disk file }
    FileName        : OneLine;      { Filename returned from selection box }
    fv
    : ascii;                        { Disk file variable }
    Printer         : file of Text; { Printer file variable }
    b               : Boolean;
    exitprompt      : Boolean;
{$I GEMSUBS.PAS }
begin
    if Init_Gem >= 0 then { Check to see if GEM is ready }
    begin
        PathName := 'A:*.PAS'; { Default path name for selection box }
        Rewrite(Printer, 'LST:');
        if Get_In_File(PathName, FileName ) then { GEM access for selection box }
        begin
            Reset(fv, FileName);
            m := 1;
            n := 1;
            repeat { Print page heading }
                Writeln(Printer);
                Writeln(Printer,'Personal Pascal Program Lister':55);
                Writeln(Printer, 'File ':36, FileName);
                Write(Printer,'-----------------------------------------');
                Writeln(Printer,'-----------------------------------------');
                Writeln(Printer);
                for i := 1 to 56 do
                begin
                    b := EOF(fv); { Check for end of file }
                    if not b then
                    begin
                        Readln(fv, LineOutput); { read disk file }
                        Writeln(Printer, n:4, ' ', LineOutput);
                        n := n +1; { increment line counter }
                    end
                    else
                    begin
                        ExitPrompt := true; { Boolean for exit flag }
                        WriteLn(Printer);
                        n := n + 1; { increment line counter }
                    end;
                end;
                Writeln(Printer);
                Writeln(Printer, 'Page':38, m:3);
                Writeln(Printer);
                Writeln(Printer);
                Writeln(Printer);
                m := m + 1; { increment page counter }
            Until ExitPrompt;
            Exit_Gem;
        end;
    end.
```

movement/button pressed, and keyboard input.

The *Personal Pascal* disk comes with examples of how to use Pascal to access many of the GEM functions. With these examples to work from, I was quickly able to learn how to position windows, draw simple graphics, and use dialog boxes, alert boxes, and drop-down menus. In contrast, I had trouble with mundane tasks such as printer output because neither the manual nor the demos included examples of the proper syntax.

Text Editor Included

When you run *Personal Pascal*, you're presented with the Manager—a GEM screen that offers the ability to access desk accessories (if any), manipulate files (Edit, Compile, Link, Run Program, and Quit), and set program options (Compiler Options, Linker Options, and Save Options). On a 520ST, *Personal Pascal* leaves 347K of workspace for the text editor, but this can vary depending on the number of desk accessories loaded.

You can enter a new program or edit an existing one with the built-in text editor, or with any text editor/word processor that can save files in ASCII format (for instance, *1ST Word* with WP Mode turned off). The *Personal Pascal* editor does not support GEM features, although this may change in a future update. The manual says the text editor is "adequate," and I found it easy to use and very useful due to its integration with the compiler/linker. Cursor movements, inserts, and deletes can be accomplished with the cursor keypad or by using *WordStar*-style key commands. Search-and-replace and file operations make use of the special function keys and have no *WordStar* equivalents.

When you're using the text editor, help is only a keystroke away. Pressing the Help key displays a full screen of information about the function keys, moving the cursor, inserting and deleting, cutting and pasting, and manipulating files.

Choosing the Compiler Options selection from the Manager allows you to customize the compiler's method of handling errors during compilation and execution. During the compilation phase, errors are written to a disk file, but the programmer can choose to have the compiler pause after every error. At this point, the program presents three choices: exit back to the editor to the point of the error, continue the compilation, or abort the compilation and return to the Manager. You can also choose a full debugging mode which locates and identifies any errors encountered during program execution. Of course, this mode adds more length to your executable code.

Other compiler options include automatic linking of successfully compiled programs, and a choice of compiling for either the GEM or TOS environment. These options, as well as others such as modular compilation, can also be selected via compiler directives included in the source code.

Fast Compilation

Personal Pascal uses a series of overlay programs for editing, compiling, and linking. To compile and link a program, the compiler overlay is loaded, the program is compiled, the linker overlay is loaded, and the files are linked. While this uses memory efficiently, it is somewhat time-consuming.

Still, a small program compiled for GEM has a respectable compilation/linking time of 1 minute, 42 seconds. You can significantly speed up this process by using a hard disk drive, or better yet, a RAM disk to hold the *Personal Pascal* programs and overlays. (This is a good reason to upgrade your 520ST to one megabyte of memory, or to use a 1040ST.)

OSS has been very active in supporting and upgrading *Personal Pascal* since its release. Additional information and sample programs can be found in the Atari 16-bit Special Interest Group on CompuServe (enter GO ATARI16) and on OSS's own Bulletin Board System (408-446-3451). In both places, you'll find numerous files available for downloading that expand upon the manual, especially in the area of GEM functions.

Upgrades are also available from these sources. I purchased version 1.00 of *Personal Pascal* and, via CompuServe, was able to upgrade to the equivalent of version 1.02 simply by downloading a short patch pro-

COPY II ST™

BACKUP PROTECTED SOFTWARE FAST.

From the team who brought you COPY II PLUS (Apple), COPY II PC (IBM) and COPY II MAC (Macintosh) comes a revolutionary new copy program for the Atari 520 and 1040 ST computers.

- Copies many protected programs—automatically. (We update COPY II ST regularly to handle new protections; you as a registered owner may update at any time for $15 plus $3 s/h.)
- Supports single and double sided drives.
- Includes both a fast sector-based copier and a true bit copy mode for protected disks.

Requires an Atari 520 or 1040 ST computer with one or two drives.

Call 503/244-5782, M—F, 8—5:30 (West Coast time) with your VISA in hand. Or send a check for $39.95 U.S. plus $3 s/h, $8 overseas.

$39.95

Central Point Software, Inc.
9700 S.W. Capitol Hwy. #100
Portland, OR 97219

Central Point Software
INCORPORATED

Backup utilities also available for the IBM, Apple II, Macintosh and Commodore 64.
This product is provided for the purpose of enabling you to make archival copies only.

gram. This is the type of upgrade policy the industry should try to emulate.

Rough For Beginners

My only reservation about *Personal Pascal* is the manual. I feel a beginner would be overwhelmed by the complexity of GEM-based Pascal due to the paucity of examples and the inability to use some standard input/output procedures in the GEM environment. I've had to rely on other sources and my own experimentation to deal with input/output problems, disk file access, and other more complex tasks. Although I was able to learn *Turbo Pascal* from the manual provided, I cannot say the same about *Personal Pascal*.

However, this doesn't mean that *Personal Pascal* is not a powerful and useful programming language, because it most definitely is. If you're a beginner, be prepared to dig for some of the information required to produce that gem of a Pascal program.

Overall, I believe OSS has another successful programming language on its hands. It's a quality program at a fair price, and it's backed up by excellent technical support. If you're looking for an alternative to interpreted languages and would like full access to the power of GEM, you should consider *Personal Pascal*.

I've included a short program written in *Personal Pascal* which exhibits the ease of GEM access. The program prompts the user with the familiar GEM item-selector window to choose a program to be listed to the printer. I find that a program listing is invaluable for debugging, yet *Personal Pascal* requires you to exit the Manager and use the desktop file dump to get hardcopy. This program solves the problem. It prints 56 lines of code per page, inserts a page break, and numbers each page. It can be run from the *Personal Pascal* Manager and will print any ASCII file. The executable program can be found on the magazine disk under the filename LISTER.PRG, and the *Personal Pascal* source code is saved as LISTER.PAS.

Personal Pascal
Optimized Systems Software
1221B Kentwood Avenue
San Jose, CA 95129
$74.95

Megamax C

George Miller
Assistant Technical Editor

Requirements: Atari ST computer with disk drive and monochrome or color monitor.

Many ST programmers are experimenting with the various programming languages that have become available during the past year. Just about every popular high-level language has been adapted for the ST, and after discovering the limitations of ST BASIC and DR Logo, even casual programmers are investigating the alternatives.

It's always tough to learn a new language. It's even more difficult when you must also learn a new operating system—such as the Atari ST's TOS—and how to communicate with a compiler at the same time. When something goes wrong, it seems as if there are too many places to begin looking for the problem. Is the source code at fault, or was there a mistake in the batch program? Sometimes the task appears insurmountable.

After having worked with several languages, I was pleased to discover that *Megamax C* provides a very comfortable working environment on the ST. The C programming language, developed in the early 1970s by Dennis Ritchie and Brian Kernighan, has become extremely popular on the ST, Commodore Amiga, and Apple Macintosh computers because of its power, flexibility, and easy portability from one computer to another.

A Shell Game

The *Megamax C* version of the language consists of two disks. The System disk contains the editor, compiler, linker, system library, header files, a desk accessory library, a double-precision math library, and a shell program. The shell is a command interpreter which functions as an alternative to the GEM desktop. Although the *Megamax C* shell is somewhat limited in terms of the commands it offers, it does allow you to manipulate the system quite easily. Everything you need to use is accessible from the shell.

If you'd prefer to program in a Unix-like environment, *Megamax C* works well when used with *Micro C-Shell* from Beckemeyer Development Tools. Portions of the *Megamax C* documentation even refer to the proper command line syntax to be used when running it from within *C-Shell*.

The second disk in the *Megamax C* package is the Utility disk. It contains a folder of example programs, a resource construction program, a librarian, a disassembler, a sample game compiled with *Megamax C*, and a code improver. It's really quite a package.

The compiler was designed to adhere to the standards espoused in *The C Programming Language*, a book written by Kernighan and Ritchie that has become the bible for C programmers. I referred to this book frequently during testing and have found *Megamax C* to be compatible.

The *Megamax C* manual is an attractive looseleaf notebook. No attempt is made to teach C; it's assumed that you're already familiar with the language. The manual provides you with information about this specific implementation of C, the support programs, and the interface to GEM. With this understanding, the documentation is quite complete.

The Text Editor

Megamax C includes a text editor that is simple to use, yet very powerful. You can access the editor from the desktop or the shell. In many ways, the shell is more convenient.

The text editor isn't designed to be used as a word processor; it simply creates source code in ASCII format to be compiled later. However, since it creates ASCII files, at times I've found myself using this fine editor to create files for other purposes—even in place of the ST BASIC editor.

As with any good text editor operating under GEM, text is manipulated in a screen window. Up to four windows may be open at a time, and text may be moved between the windows. The windows may be moved and resized to suit your preference. As you edit a file, you can select options from drop-down menus or by typing keyboard commands. A help screen is available by pressing the Help key.

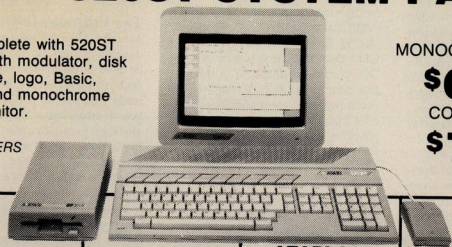
Naturally, the cut, paste, and search functions you'd expect are included. About the only function that seems to be missing is a means to print source files from within the editor.

ATARI SPECIALISTS

ATARI 520ST SYSTEM PACKAGE

Comes complete with 520ST computer with modulator, disk drive, mouse, logo, Basic, 1st Word, and monochrome or color monitor.

1040ST OWNERS CALL!



MONOCHROME SYSTEM
$649.00
COLOR SYSTEM
$749.00

SUPRA DISK
20 Meg HARD DRIVE
$829.00

SUPRA MODEM
MODEL 1200ST $159.00

ATARI 314
1 Meg Double Sided DISK DRIVE
$219.00

ANCHOR 520
Direct Connect MODEM
300/1200 Baud
$149.00

— PRINTERS —

CITIZEN
MSP-10 (80 col.)....$279.00
MSP-15 (132 col.)....$419.00
MSP-20 (80 col.)....$349.00
MSP-25 (132 col.)....$499.00

C.ITOH
Prowriter 1550P....$299.00
Starwriter 10-30....$299.00

EPSON
Homewriter 10, LX80....CALL
FX85, FX286, RX100....CALL
SQ2000, HI80, HS80, AP80....CALL
LQ800, LQ1000....CALL

JUKI
6000 Letter Quality....CALL
6100 Letter Quality....CALL
6200 Letter Quality....CALL
6300 Letter Quality....CALL
6500 Letter Quality....CALL
5510 Lot Matrix....CALL
5510 Color....CALL

OKIDATA
182, 183, 192, 193, 292, 293, 84....CALL
Okidata 93....$279.00
Okimate 20....$149.00

LEGEND
880 Dot Matrix 100 cps....$179.00
1080 Dot Matrix 100 cps....$259.00
1380 Dot Matrix 130 cps....$289.00
1385 Dot matrix 165 cps....$339.00

NEC
3000 Series....$779.00
8000 Series....$1099.00
ELF 360....$399.00
Pinwriter 560, 660, 700....CALL

PANASONIC
KX1091....$239.00
KX1092....$359.00
KX1592....$469.00

TOSHIBA
P321 (80 column)....$489.00
P341 (132 column)....$749.00
P351 (132 column)....$1049.00

STAR
NX/SB/SD/SR Series....CALL
Powertype Letter Quality....CALL

SUBLOGIC Flight Simulator II
$37.99

ACCESS
Leaderboard....$29.99

ACTIVISION
Hacker....$29.99
Borrowed Time....$34.99

BATTERIES INCLUDED
D.E.G.A.S....$27.99

ELECTRONIC ARTS
Financial Cookbook....$37.99

HABA/ARRAYS
Hippo-C....$44.99
Business Letters....$29.99
Write Your Own Will....$29.99
Haba Writer....$37.99
Habadex Phonebook....$27.99
Mail Room Manager....$39.99

SOFTWARE

INFOCOM
Cutthroats....$29.99
Deadline....$34.99
Enchanter....$29.99
Hitchhiker's Guide....$29.99
Infidel....$34.99
Planetfall....$29.99
Sea Stalker....$29.99
Sorcerer....$34.99
Starcross....$34.99
Suspect....$29.99
Suspended....$34.99
Wishbringer....$29.99
Witness....$29.99
Zork I....$29.99
Zork II....$29.99
Zork III....$29.99

MINDSCAPE
Deja Vu....$37.99

MIRAGE
Express....$34.99

MUSE SOFTWARE
Final Word....$99.99
Hex....$27.99
PC Intercom....$79.99

O.S.S.
Personal Pascal....$49.99

PENQUIN SOFTWARE
Crimson Crown....$29.99

SIERRA-ON-LINE
Ultima II....$39.99
King's Quest....$37.99

V.I.P
Professional....$139.00

ATARI 520ST Hardware/Peripherals

Atari 520 CPU....$369.00
Atari 124 Monochrome Monitor....$189.00
Atari 1424 14'' Color Monitor....$269.00
Atari 354 Single Sided Drive....$179.00
Haba 10 Mb Hard Drive....$669.00

BLANK DISKETTES & ACCESSORIES

AMARAY
Disk Tub 3½''....$9.99

MAXELL
(10) 3½'' SS/DD....$18.99
(10) 3½'' DS/DD....$29.99

SONY
(10) 3½'' SS/DD....$18.99
(10) 3½'' DS/DD....$29.99

CURTIS Surge Suppressors
Emerald....$39.99
Ruby....$59.99
Diamond....$29.99
Safety Strip....$19.99
Sapphire....$49.99

COMPUTER MAIL ORDER

CALL TOLL-FREE 1-800-233-8950

TELEX: 5106017898

477 East Third Street, **Dept. A608,** Williamsport, PA 17701

POLICY: Add 3% (Minimum $7.00) shipping and handling. Larger shipments may require additional charges. Personal and company checks require 3 weeks to clear. For faster delivery use your credit card or send cashier's check or bank money order. Pennsylvania residents add 6% sales tax. All prices are subject to change and all items are subject to availability. Defective software will be replaced with the same item only. Hardware will be repaired or replaced at our discretion within the the terms and limits of the manufacturer's warranty. We cannot guarantee compatibility. All sales are final and returned shipments are subject to a restocking fee.

1-800-268-3974
Ontario/Quebec

1-800-268-4559
Other Provinces

— CANADIAN ORDERS —

All prices shown are for U.S.A. orders. Call the Canadian Office for Can. prices.

1-416-828-0866
In Toronto

Telex:06-218960
2505 Dunwin Drive,
Mississauga, Ontario
Canada L5L1T1

The Compiler

The compiler portion of *Megamax C* may be executed from the desktop or the shell. Compiling is fast and easy. In one pass it checks syntax and generates the object code.

To help you optimize your programs, *Megamax C* includes a machine language assembler. The manual contains instructions for formatting 68000 code for assembly. Of course, you'll need a 68000 reference book if you're not already familiar with this subject.

If your source code contains errors—and whose doesn't?—*Megamax C* creates a special disk file with information about the errors. This makes it much easier to correct many of the more common mistakes. When an error forces the compiler to abort while you're working from within the shell, *Megamax C* reloads the text editor, places your source code in a window, and opens another window to display a list of the errors. Line numbers make it easier to find the portion of your program which needs further attention.

C does not use line numbers for program execution, of course, but the text editor does have a GOTO function that positions the cursor a specified number of lines down from the top of your source code.

The Linker

Using the linker is also very easy. Linking can be a confusing process with some development systems, but not with *Megamax C*. All you have to do is select Link from within the shell, then use the mouse pointer to select the files to be linked from the dialog window. Only valid link files are displayed, so you can't link an improper file type.

The linker is a semi-intelligent program. Although it reads an entire object file from the library, it loads only those routines and external data definitions which are referenced by your program. This results in smaller, more efficient executable object code. To test this, we ran some sample source code through *Megamax C* and another popular C compiler for the ST. The *Megamax C* object code was up to 66 percent smaller.

The resource construction program included with *Megamax C* makes it easier to create and edit disk-based resources such as drop-down menus, dialog boxes, and GEM icons. The manual takes you step by step through the process of creating your own resource files.

Another helpful section of the manual covers the use of the standard input/output functions in C. There are also sections on using the GEM Application Environment Services (AES), GEM Virtual Device Interface (VDI), Basic Input/Output System (BIOS), Extended BIOS (XBIOS), and GEM Disk Operating System (GEMDOS) functions of the ST. Although this information is very complete, you'll probably want additional reference material as well. The manual provides a complete description of each function, but it makes no mention of any necessary preparatory routines.

Overall, *Megamax C* is worth a look by serious programmers. You'll find it fast and easy to use. *Megamax C* just might become the C standard for the Atari ST.

Megamax C
Megamax, Inc.
P.O. Box 851521
Richardson, TX 75085
$99.95

Thunder!

Tom R. Halfhill, Editor

Requirements: Atari ST computer with disk drive and monochrome or color monitor.

One of the trendiest features appearing in word processors recently is the integrated spelling checker—a program that compares every word in your document against a disk-based dictionary, then flags any unrecognized words as possible misspellings. However, most word processors for the ST still lack a spelling checker, even though the ST's speed and memory make it ideal for this purpose.

Batteries Included has a solution: a spelling checker that's designed to work with practically any word processor. In fact, it works with virtually *any* program that supports GEM, including text editors, database managers, terminal programs, and spreadsheets (although obviously, a spelling checker may not be very useful in some of these cases).

It's called *Thunder!*, in apparent homage to Borland International's *Turbo Lightning*, one of the most popular spelling checkers for the IBM PC. *Thunder!*, like *Turbo Lightning*, is a *realtime* spelling checker—miraculously, it's capable of detecting your misspellings and typos the very instant they're entered.

Two Programs In One

Actually, *Thunder!* consists of two independent spelling checkers that can be used in completely different ways. As a realtime checker, *Thunder!* proofreads your text as you type, alerting you to errors instantly. As a standalone program, *Thunder!* works more like a conventional spelling checker, proofreading a finished document that is stored on disk. Let's examine the realtime version first.

Installed as a desk accessory when you boot up the ST, *Thunder!* hovers invisibly in the background and constantly monitors your keystrokes—whether you're using a word processor or any other kind of program. (It can be temporarily disabled if you wish.) Whenever you finish typing a word (as delimited by a space character or sentence terminator), *Thunder!* checks to see if that word exists in its 50,000-word dictionary. If it can't find a match, it alerts you by beeping the monitor speaker.

Now you have three options. First, if you're sure the word is correctly spelled (remember that *Thunder!* beeps at any word it can't find in its dictionary), you can simply ignore the beep and continue typing. Second, you can keep typing up to 80 more characters, perhaps to finish the phrase or sentence while it's still fresh in your mind. Then you can ask *Thunder!* to suggest a correct spelling for the word it beeped at. Or third, you can stop typing after the beep and ask *Thunder!* to suggest a correct spelling immediately.

To ask *Thunder!* for its suggestion, you can either press the Alt-Z keys or select Thunder! from the desk accessory menu. This pops open a window that displays the beeped word and a list of alternatives suggested by *Thunder!*.

If the word you intended to type is on the list, you can simply point to it with the mouse cursor and click the mouse button. In a flash, the *Thunder!* window closes, the mistyped word in your document is deleted, and the corrected word is automatically retyped in its place. If you typed a few

Atari ST Extravaganza

ATARI 520 SYSTEM PACKAGE*

*Including RGB or Monochrome Monitor, Mouse, Disk Drive, Basic, Logo, Neochrome, 1st Word, TOS on ROM, and RF Modulator

CALL MONOCHROME SYSTEM

CALL RGB/COLOR SYSTEM

ATARI 1040 ST SYSTEM PACKAGE*

*Including RGB or Monochrome Monitor, Mouse, Double-sided Disk Drive, Basic, Logo, Neochrome, 1st Word, TOS on ROM, and Built-in Power Supply

CALL FOR CURRENT PRICE

***All ST System Packages are sold with a ComputAbility 90 Day Warranty**

SUPRA 20 MEG 3.5 INCH HARD DRIVE only $659

ATARI SF 314 DISK DRIVE
Double sided/ 1 Megabyte storage
$209

PRINTERS

Panasonic 1091	239	Epson	Call
Panasonic 1080	215	Star SG-15	369
Star NX-10	239	Star SD-10	339
Star NL-10	Call	Star SD-15	449
Silver Reed	Call	Star SR-10	Call
Juki	Call	Star SR-15	Call
Toshiba 321	529	Star NB-15	Call

DISKETTES
SUPER SPECIALS*

3.5 SS/DD DISKETTES (BOX OF 10) ONLY **15.95**

3.5 DS/DD DISKETTES (BOX OF 10) ONLY **24.95**

BUY DISKETTES AT THESE LOW PRICES WHEN ADDED TO ANY OTHER ORDER

*PRICES EFFECTIVE NOW THRU OCTOBER 31, 1986

QMI ST 1200 BAUD MODEM
Direct connect with ST talk
$159

MODEMS

Prometheus 1200	299
Maxwell 1200	229
QMI 1200 ST	159
Supra 1200 ST	159

520 ST ONE MEGABYTE UPGRADE
Computer must be sent in to service center
$175

PRINTER BUFFERS

U-BUFF 16K	79.95
U-BUFF 64K	99.95

We receive ST Products on a weekly basis. If you don't see it here, call.

ATARI 520 ST • ATARI 1040 ST SOFTWARE

ABACUS

Textpro	37.95
Filepro	37.95
Form/MT	37.95
Paintpro	37.95
Textdesigner	37.95
Assempro	37.95
PC Board Designer	299.95

INFOCOM ST

Forever Voyaging	29.95
Ballyhoo	27.95
Cutthroats	27.95
Deadline	34.95
Enchanter	27.95
Hitchhiker	27.95
Infidel	29.95
Planetfall	27.95
Sea Stalker	27.95
Sorcerer	29.95
Spellbreaker	34.95
Starcross	34.95
Suspect	29.95
Suspended	34.95
Witchbringer	27.95
Witness	27.95
Zork I	27.95
Zork II or III	29.95
Trinity	27.95
Moonmist	27.95
Leather Goddess	27.95

ST BUSINESS

VIP Professional	199.95
Synsoft Gen. Ledger	44.95
SBM (Point of Sales)	84.95
Cash Disbursements	69.95
Sierra Accts. Rec.	69.95
Sierra Gen. Ledger	69.95
Sierra Payroll	69.95
VIP Ledger	67.95
Financial Cookbook	34.95
Swiftcalc ST	59.95
DAC Easy Accounting	49.95
DAC Payroll	34.95
Ingur Portfolio	Call

HIPPOPOTAMUS

Almanac	23.95
Disk Utilities	34.95
Eprom Burner	119.95
Jokes & Quotes	23.95
Ramdisk	23.95
Backgammon	27.95
Hippocalendar	62.95
Hippopixel	27.95
Hipposimple	34.95
Hippospell	27.95
Hippoword	62.95
X-10 Powerhouse	Call
Hippofonts	27.95
Digitizer	119.95

MICHTRON

Alt	34.95
BBS	Call
Business Tools	84.95
Calendar	20.95
Cornerman	34.95
D.F.T.	34.95
DOS Shell	27.95
Echo	27.95
Flipside	27.95
Goldrunner	27.95
Kissed	34.95
Lands of Havoc	16.95
M-Copy	34.95
M-Disk	27.95
Major Motion	27.95
Mi-Term	34.95
Michtron Utilities	34.95
Mudpies	27.95
Softspool	27.95
Solitaire	27.95
The Animator	27.95
Time Bandits	27.95
Mighty Mail	84.95
Miss'n Mouse (Mono)	27.95
Easy Record	54.95
Intro to Logo	34.95
Personal Money Mngr	34.95

ST WORD PROCESSORS

Final Word	94.95
Paperclip Elite	Call
Regent Spell	31.95
Habawriter	54.95
Regent Word II/ Gem-Based	64.95
Wordwriter ST	59.95
Thunder	27.95

ST DATABASES

DB Man	109.95
Zoomracks	54.95
Regent Base (Gem Based)	64.95
Data Manager ST	59.95
DB Master One	Call

ST GRAPHICS

Degas	27.95
N-Vision	27.95
Easy Draw	99.95
PC Board Designer	299.95
Paintworks	49.95

ST UTILITIES

ST Toolbox	24.95
Macrodesk	27.95
Deskmaster	27.95
ST Form	31.95
Music Studio	39.95
Sound Wave	33.95
Abacus Books	Call
Time Link	34.95
ST Music Box	31.95
Micro Manager	49.95
TOS on ROM	25.00
Logichron Clock	39.95

ST ADVENTURES

Ultima II	39.95
Farenheit 451	33.95
Treasure Island	27.95
King's Quest II	33.95
Perry Mason	33.95
Word Invaders	22.95
Borrowed Time	33.95
Hacker	29.95
9 Princes/Amber	33.95
Sundog	24.95
Mindshadow	33.95
Winnie The Pooh	19.95
Black Cauldron	27.95
Amazon	33.95
Spiderman	16.95
Fantastic Four	16.95
Apshai Trilogy	27.95
Universe II	47.95
The Pawn	27.95
Ultima III	39.95
Star Quest	33.95
Kings Quest III	33.95
Hacker II	33.95

ST LANGUAGES

Personal Pascal	49.95
Personal Prolog	59.95
Fast Basic	89.95
Fast C Compiler	99.95
Fast Fortran	199.95

ST TELE-COMMUNICATION

PC Intercom	84.95
ST Talk	17.95

ST PRINT UTILITIES

Typesetter	24.95
Rubber Stamp	24.95
Printmaster	24.95
Art Gallery I	19.95
Art Gallery II	19.95
Fontwriter	27.95

ST ARCADE GAMES

Hex	27.95
Monkey Business	16.95
Delta Patrol	16.95
Computerbase	19.95
Bridge 4.0	20.95
Winter Games	27.95
Rogue	27.95
Diablo	20.95
Super Huey	27.95
Phantasie	27.95
Mean 18	27.95
Leader Board	27.95
Bratacus	33.95
Donald Duck	20.95
Silent Service	27.95
Flight Simulator II	34.95
Champ. Wrestling	27.95
World Games	27.95
Masterpiece	27.95
Video Vegas	23.95
Blazing Paddles	27.95
Chessn ter 2000	32.95
Computer Softball	27.95
Strip Poker	27.95
Super Huey	27.95
Starglider	29.95
Alternate Reality	Call
Little Computer People	34.95
Smoothshaker	Call

Est. 1982
ComputAbility
P.O. Box 17882, Milwaukee, WI 53217
ORDER LINES OPEN
Mon.-Fri. 11 a.m.-7 p.m. CST • Sat. 12 p.m.-5 p.m. CST

To Order Call Toll Free
800-558-0003

For Technical Info, Order Inquiries, or for Wisc. Orders
414-351-2007

ATARI is a trademark of ATARI, INC.

No surcharge for MasterCard or Visa

ORDERING INFORMATION: Please **specify system.** For fast delivery send cashier's check or money order. Personal and company checks allow 14 business days to clear. School P.O.'s welcome. **C.O.D. charges are $3.00.** In Continental U.S.A. include $3.00 for software orders. ■ shipping for hardware. Minimum $4.00. MasterCard and Visa orders please include card # expiration date and signature. WI residents please include 5% sales tax. HI, AK, FPO, APO, Puerto Rico and Canadian orders, please add 5% shipping, **minimum $5.00.** All other **foreign orders add 15% shipping, minimum $10.00.** All orders shipped outside the Continental U.S.A. are shipped first class insured U.S. mail. If foreign shipping charges exceed the minimum amount, you will be charged the additional amount to get your package to you quickly and safely. All goods are new and include factory warranty. Due to our low prices all sales are final. **All defective returns must have a return authorization number.** Please call (414) 351-2007 to obtain an R.A. # or your return will not be accepted. Prices and availability subject to change without notice.

words past the misspelled word before opening the window, *Thunder!* even restores this text after retyping the corrected word. Either way, you're free to pick up where you left off.

Transparently Fast

We tested *Thunder!* with a few popular word processors, including *1ST Word* and *HabaWriter*. Besides beeping virtually instantly after you mistype a word, *Thunder!* also has negligible effect on the word processor's keyboard response. Kudos for Mark Skapinker, the programmer behind this bit of magic.

Part of the secret is that unlike most spelling checkers, *Thunder!* stores its entire main dictionary in memory instead of on disk. And the main dictionary employs a bit of magic itself, because it squeezes 50,000 words into only 86K.

Even with 50,000 words, however, there are some inevitable omissions. In particular, the dictionary's weak points are contractions and compound words. *Thunder!* beeps at all contractions, such as *don't, I'm, isn't, can't*, and so on. Compound words such as *online* also provoke beeps, because *Thunder!* apparently thinks you left out the space by mistake—it suggests *on line* as the proper spelling (which, in this case, is debatable, anyway).

Fortunately, like all spelling checkers, *Thunder!* lets you customize the dictionary by adding words which are common to your vocabulary. In fact, *Thunder!* offers three ways to do this.

One way is to expand the main dictionary. You can add about 2,000 words, but there's a catch. Because of the data-compression techniques used, you can't delete a word after it's added.

Another way to add words is to create a supplementary dictionary. Although a supplementary dictionary can hold only about 200 words, you can create as many of these as you want. However, only one supplementary dictionary can be loaded into memory at one time.

The third way to expand *Thunder!*'s vocabulary is to create what is called a learned-words dictionary. This dictionary is unique in that it's more like a realtime search-and-replace feature. It consists of up to 100 word pairs, and the second word in a pair automatically replaces the first word as soon as it's typed. For instance, one word pair could be *recieve* and *receive*. Whenever you type *recieve*, *Thunder!* automatically deletes it and substitutes *receive*.

Another use for the learned-words dictionary is to automatically expand abbreviations. For example, *JCD* could be instantly expanded into *John Carter Doe*. The learned-words dictionary even includes two built-in abbreviations: *tdate* and *ttime* automatically expand into the current date and time as set by the system clock.

The Stand-Alone Checker

Realtime spelling checkers are fun to watch in action, but some people find them intrusive. To keep everyone happy—and to make *Thunder!* useful with programs that don't support GEM desk accessories—*Thunder!* comes with a stand-alone spelling checker as well. This version of *Thunder!* lets you check any ASCII file stored on disk. It loads and runs as an independent program, but can use the same dictionaries as the realtime *Thunder!*.

Once the stand-alone *Thunder!* is loaded, you can open a text file for spell-checking. Just be sure this is an ASCII file; when one of our testers tried checking a non-ASCII *1ST Word* file that contained style codes, the file was irreparably scrambled.

As the stand-alone *Thunder!* proofreads your document, it displays the text in a small window and beeps whenever it doesn't recognize a word. The list of alternatives then appears in another window. As usual, you can substitute a suggested word for a beeped word by pointing and clicking with the mouse. If you decide that the beeped word is actually correct, you can tell *Thunder!* to skip the word and even subsequent occurrences of that word.

Additional options let you automatically change all occurrences of a beeped word to a corrected word throughout the document, enter correct words yourself from the keyboard, check words that you enter against the dictionary, add beeped words to the main or supplementary dictionaries, and exit the spell-checking process.

Like the realtime version of *Thunder!*, the stand-alone program is fast. When proofreading a text file stored on floppy disk, it checks about 40 words per second. That speed doubles if the document is stored in a RAM disk.

Extra Features

As a bonus, the stand-alone *Thunder!* can also analyze your document for readability. Among other things, it counts the number of characters, syllables, words, words longer than three syllables, sentences, and paragraphs. Then it applies two common formulae which evaluate the readability of the document in terms of school grade levels.

The stand-alone *Thunder!* has a few other advantages, too. It can handle learned-word dictionaries of up to 300 word pairs and supplementary dictionaries of up to 600 words.

Both versions of *Thunder!* can be customized. For instance, you can specify that certain supplementary and learned-word dictionaries should be loaded automatically, and that words beginning with uppercase letters should be ignored (this keeps *Thunder!* from beeping at proper names). These and other options are saved to disk in a configuration file that is checked the next time *Thunder!* is loaded.

Thunder! is not copy-protected, so you're free to make backups for personal use and put *Thunder!* on all your word-processing disks.

Overall, *Thunder!* is an extremely fast and flexible spelling checker that is a worthwhile addition to the software library of any writer, student, or professional. If *Thunder!* saves you from embarrassing misspellings in a resume, term paper, business report, or important letter, it's probably well worth the price. We suspect *Thunder!* is destined to make a lot of noise.

Thunder!
Batteries Included
30 Mural Street
Richmond Hill, Ontario
Canada L4B 1B5
$39.95

DISCOUNT SOFTWARE

ATARI HARDWARE

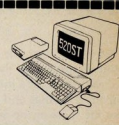
520 ST Computer w/Modulator,
Mouse & Software
SF 354 3½" SS/DD Disk Drive
SF 314 3½" DD/DD Disk Drive
SM 124 12" Monochrome Monitor
SC 1224 12" RGB Color Monitor
SH 204 5½" 20 Meg. Hard Drive

ATARI 520 ST

PERSONAL COMPUTER SYSTEM

CALL FOR PRICES!!

INDIVIDUAL ITEMS OR PACKAGES



CALL ON THESE AND OTHER PRODUCTS
WE CARRY A COMPLETE LINE OF SOFTWARE AND HARDWARE

ABACUS SOFTWARE
PC Board Designer $279

ACADEMY SOFTWARE, INC.
Typing Tutor and
Word Invaders $27

ACCESS
Leaderboard Golf $29

ACTIVISION
Borrowed Time $35
Hacker $39
Hacker II ..... CALL
Mindshadow $39
The Music Studio $39

ATARI CORP.
1st Word ..... CALL
ST Planetarium $24
CP/M-Z80 Emulator $39
DB Master One $39
Joust $24
Millipede $24
Star Raiders $24

AUDIO LIGHT
N-Vision $21

BATTERIES INCLUDED
Degas ..... $ 29
Homepak ..... $ 54
Paperclip Elite ..... CALL
Isgur Portfolio System $149

BAYVIEW SOFTWARE
Word For Word $29

ELECTRONIC ARTS
Financial Cookbook $39
Marble Madness ..... CALL

ELECTRONIC MUSIC PUBLISHING HOUSE
MIDI Play I $39

EPYX
Rogue $25
Temple of Apshai Trilogy $27
Winter Games $29

FIREBIRD LICENSES, INC.
The Pawn $34
Starglider $34

HABA SYSTEMS
Haba Check Minder $59
Haba Hippo "C" $59
Haba Solutions:
Business Letters $39
Haba Solutions: Wills $39
Haba Dex Phone Book $45
Haba View $59
Haba Writer $54

HIPPOPOTAMUS SOFTWARE
Hippo Almanac $25
Hippo Art I $29
Hippo Backgammon $29
Hippo Fonts I $29
Hippo ST Disk Utilities $39
Hippo ST Ramdisk $26
Hippo ST Sound Digitizer $109
Hippo Video Digitizer ..... CALL

HOLMES & DUCKWORTH
Toolbox $29
Forth $39
H & D Base $79

INFOCOM SOFTWARE
A Mind Forever Voyaging $35
Ballyhoo $29
Cutthroats $29
Deadline $39
Enchanter $29
Hitchhiker's Guide
to the Galaxy $29
Infidel $35
Planetfall $29
Seastalker $29
Sorcerer $37
Suspect $37
Witness $29
Wishbringer $29
Zork I $29
Zork II $37
Zork III $37
Invisiclue Books ..... $ 7

MICHTRON
Animator $39
Calendar $29
Cornerman $39
Desk Diary $29
DOS Shell $29
Echo $79
Flip Side $29
Gold Runner $29
Kissed $29
Major Motion $29
Mi-Dupe ..... CALL
Mighty Mail $39
Soft Spool $29
Time Bandit $29

MICRO-W DISTRIBUTING, INC.
MIDI Magic $39

MICROPROSE
Silent Service $29

MIGRAPH
Easy Draw $129

MINDSCAPE
Bratacccus $39

OMNITREND
Universe II $54

OPTIMIZED SYSTEM SOFTWARE, INC.
Personal Pascal $59

PHILON
FAST/BASIC-M $99
FAST/"C" $109
Henry's Fundamental
BASIC $39
FAST/FORTRAN $109
FAST/PASCAL $109

POLARWARE
The Crimson Crown $29
Transylvania $29

SIERRA ON-LINE
Black Cauldron $29
King's Quest II $34
Ultima II $49
Winnie The Pooh $19

SPINNAKER
Homework Helper Math $35

SUBLOGIC
Flight Simulator $29
Jet $29

TELARIUM CORP.
Fahrenheit 451 ..... $29
Nine Princes in Amber $29
Perry Mason $29

WINDHAM CLASSICS
Treasure Island $29

UNISON WORLD, INC.
Printmaster $29
Art Gallery $24
Art Gallery II $24

VIP TECHNOLOGIES
VIP Professional $129

XLENT SOFTWARE
Megafont $29
Music Box $39
Rubber Stamp $29
Typesetter $29

ST BOOKS
ST Internals $18.00
GEM Programmer's
Reference $18.00
ST Tricks & Tips $18.00
ST Peeks & Pokes $15.00
Basic Training Guide $15.00
ST 3D Graphics $19.95
Presenting The ST $15.00
Beginner's Guide $12.95
Machine Language $18.00
Graphics & Sound $18.00
ST Logo Guide $18.00
ST Basic to C $18.00

PRINTERS
OTHERS AVAILABLE – PLEASE CALL
Star-NX-10 ..... $279
Epson FX-85 ..... $399
Panasonic 1091 ..... $229
Okimate 20 ..... $145

DISKS
3½" Sony DSDD $39
3½" Sony SSDD $25
5¼" Bonus SSDD ..... $ 8
5¼" Bonus DSDD $10

DUST COVERS
520 ST Computer $7.00
SF 354 SS/DD $7.00
SF 314 DS/DD $7.00
Mouse House $5.50
Mouse Mat $7.50

ALL TITLES ON DISK UNLESS MARKED (R) FOR ROM CARTRIDGE
WE CHECK FOR STOLEN VISA & MASTERCARD

Order Line
1-800-282-0333
ORDERS ONLY! Sales Personnel Do Not Have Information on Previous Orders or Product Specs.

M-F 10 a.m.-8 p.m.; Sat. 10-3 ET
37 S. Broad Street, Fairborn, OH 45324

Customer Service & Ohio Residents
1-513-879-9699

AD #CPST-096

VISA

SOFTWARE ONLY – Prepaid orders over $50 receive free shipping via UPS in continental U.S. Please add $2 orders under $50. **HARDWARE** and all orders requiring shipment via U.S. Post Office are subject to additional freight charges. Add $5 for COD orders. VISA/MasterCard orders add $2 service charge. Ohio residents add 5.5% sales tax. Personal checks require a three-week waiting period. No waiting when paid by credit card, certified check or money order. All items subject to availability and price change. PLEASE CITE AD NUMBER WHEN ORDERING. ALL RETURNS MUST BE ACCOMPANIED BY AN AUTHORIZATION NUMBER.

MasterCard

Major Motion And *Time Bandit*

Andy Eddy

Requirements: Atari ST with disk drive and color monitor. Joystick optional for Time Bandit, *but recommended.*

Michtron has been one of the most prolific supporters of the Atari ST since the computer's introduction. The company's utilities and applications make the most of the ST, putting the power of the computer at your command. Now they've released two nearly addictive games as well: *Major Motion* and *Time Bandit.*

Hit The Road

Major Motion appears to be patterned after the popular arcade game *Spy Hunter.* You're the driver of a sleek, but deadly automobile on the prowl for enemy spies. Your mission is to remove the spies from the road by any means possible—by using guns, oil slicks, smoke screens, or just plain metal-to-metal bumping.

You start with a machine gun–equipped car, but by successfully continuing your pursuit, your arsenal will be frequently beefed up by a weapons van that cruises by. When you roll up the ramp behind the van—much like Kitt does in TV's "Knight Rider"—your car is outfitted with new weapons. Each weapon is different: helicopter-destroying rockets, repulsors that push all nearby cars from the road, turbo power, and other lethal items.

Some cars can be blown off the road with your guns, while others are armored against bullets and must be bumped off the highway. More points are awarded for knocking the cars from the road, but don't think they're defenseless. Some of the cars bristle with spikes that cut your tires. They seem to hover around the weapons van and strike before you have a chance to get full control of your car when you exit. In later levels, the enemy cars fire machine guns and rarely miss. There are even twins of your car which have fallen into enemy hands. They're almost impossible to shake off, and usually succeed in turning you into twisted roadside wreckage.

Still more obstacles impair your chase. The missile-firing helicopters leave craters in the road that must be avoided. The road itself forks and twists, posing hazards that are bad enough, even without the constant harassment from your adversaries.

Of course, the road isn't devoid of civilian drivers, either. If you become recklessly destructive and wipe out a few too many of your fellow motorists, it's assumed that you have become a traitor, and a roving aircraft will surgically remove you from the road. Each time your car is destroyed, you'll lose some of the weapons you had stored. Fortunately, you win a bonus car for each 10,000 points you score.

Waterlogged Roadhog

As a change of pace, occasionally your car takes to the water. It becomes mandatory to scan the roadside for the indication that this transformation is about to take place—a tiny, flashing sign that reads, "Bridge Out." Most of your trailing opponents will pile up at this obstacle, but there are plenty more where they came from. Mines, rocks, and islands dot the waterway as hazards, along with the continuing contingent of villains. Your strategy on water differs from your strategy on pavement. You're moving at nearly a crawl as you weave around all the dangers in your path. Still, the pace of the contest remains frantic.

Major Motion is obviously the result of careful programming. The richness of detail is remarkable. Signs are positioned along the roadside both as humorous touches and required reading. Burned-out cars and boats litter the path, getting bashed around until they're off the screen. And the action is realistic. When you hit another vehicle, your car rebounds with an equal and opposite reaction—remember physics class?—sometimes causing your car to careen to the other side of the road.

Among the other cute touches are familiar musical passages that pop up now and then. The theme tunes from "Mission: Impossible" and "Batman" float out while you're driving. And when your boat hits the water, you'll chuckle at the "Hawaii Five-O" theme. All this contributes to the enjoyable experience of playing *Major Motion.*

Hurtling down the highway in Major Motion.

A Busy Mouse

As good as it is, though, *Major Motion* is not flawless. One drawback in this high-energy arcade game is the use of the mouse controller instead of a joystick. Sometimes I feared I would damage the mouse with all my flailing. The mouse also requires a large amount of table space.

In addition to the mouse, you also have to use the keyboard to activate the various weapons you gather, since you may have more than one weapon available at any given time. The program defaults to the keypad, but you can set whichever keys you prefer with the aid of a function key. All of this means that the keyboard must be close by, and glancing down at the keys may distract you just enough to cause a fatal collision.

Another problem is extremely rare, but an annoyance nevertheless. In your travels down the road, the surface can change from pavement to dirt to ice. I discovered that if you're destroyed on ice or dirt, the weapons van sometimes doesn't come back to drop off a new car. Perhaps this has something to do with the way the program simulates sluggish responses on these surfaces. At any rate, you end up watching all the cars and helicopters pass by while you sit impatiently waiting. You can always restart the game, but it's frustrating if you were doing particularly well at the time.

Fortunately, these few problems don't significantly detract from the game. In fact, *Major Motion* is so fascinating that you'll find yourself becoming more and more deeply absorbed. This is one of those just-one-more-time games. You'll probably lose sleep and miss some appointments over this one.

ATARI®

STation

The one STop shop

Power Without the Price™

TO ORDER

CALL TOLL FREE

1-800-225-5878

IN MICHIGAN
313-595-0788

SUITE 101A
35540 MICHIGAN AVE.
WAYNE, MI. 48184

HARDWARE

ATARI 520st	$349
ATARI 1040st	$629
SF354 Drive	$139
SF314 Drive	$199
SM1224 RGB Monitor	$309
SM124 Mono Monitor	$139
SHANNER SD 2000	$349
Supra 20 MEG H.D.	$649
Supra 30 MEG H.D.	$899
AVATEX 1200 Modem	$89
Challenger 120 Modem	$149
Casio CZ-101	$299
Casio CZ-230S	CALL
MEGARAM UPGRADE	$129
Panasonic 1080	$199
Panasonic 1091	$239
Panasonic 1092	$299
STAR NX-10	$229
STAR LV-1210	$169

SUPPLIES

Diskettes

	Box
Bulk 5¼ w/sleeve	$ 4.50
Bulk 3½	$12.95
SKC 3½	$13.95
Maxell	$14.95

Ribbons

Black, Red, Green, Blue

Star-OKI	$2.99
IMAGEWRITER	$6.95
XMM801/Legend	$7.95
Panasonic	$9.95
Epson FX/RX	$4.95

Paper & Labels & Cards

Pink, Yellow, Blue, Green, Salmon, White
500 Sheets per pack

20 lb. lazer cut	$6.95
Rainbow pastel	$6.95
Rainbow Bright	$6.95

Mail Labels

1000 pack, colors	$9.95
Disk Labels, contin.	$4.95
Disk Labels Pac 50	$1.95
Continuous card stock (250) Red, Green, Blue, Orange, White	
3X5	$4.95
4X6	$7.95
5X7	$9.95

ACCESSORIES

Storage

Disk File 40-3½	$6.95
Disk File 60-5¼	$6.95
Library Case 10-5¼	$1.95
Library Case 3-5¼	$ .99
Library Case 5-3½	$ .99
Disk Directory	$22.95
Pocket Pac	$14.95

Dust Covers

520 st	$6.95
1040 st	CALL
SF354	$5.95
SF314	$5.95
SM1224	$11.95
SM124	$11.95
All Printers	$6.95

Cable-Interfaces-Etc.

St Printer cable 6'	$6.95
St Printer cable 10'	$9.95
RS 232 M/F 6'	$9.95
ST Drive Cable 6'	$14.95
XE SERIAL 6'	$9.95
P.R. Connection	$59.95
Microprint	$34.95
U Print A	$49.95
APE FACE	$42.95
Microstuffer	$59.95

Controllers

Atari	$6.95
Wico Bat	$17.95
Wico 3-Way	$22.95
TAC 2	$8.95

Miscellaneous

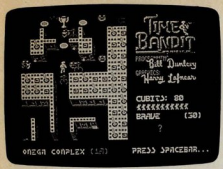
Amaray Printer Stand	$14.95
5¼ Disk Notcher	$7.95
Six outlet surgestrip	$14.95
Mouse Mat	$7.95
Mouse House	$4.95

SOFTWARE

H & D BASE	$69.95
TIME BANDITS	$27.95
LANDS OF HAVOC	$14.95
MASTERTYPE	$27.95
HIPPOWORD	$62.95
HIPPOCONCEPT	$62.95
HIPPOSIMPLE	$34.95
HIPPOSPELL	$27.95
HIPPO RAMDISK	$24.95
HIPPO COMP. ALMANAC	$24.95
HIPPO JOKES & QUOTES	$24.95
PRINT MASTER	$27.95
MINDSHADOW	$34.95
MUSIC STUDIO	$41.95
SILENT SERVICE	$24.95
BRIDGE 4.0	$20.95
BRATACCUS	$34.95
TYPING TUTOR WORD IN	$24.95
UNIVERSE II	$48.95
BLACK CAULDRON	$27.95
KING'S QUEST	$27.95
F-15 STRIKE EAGLE	$27.95
HOME PAK	$48.95
ROGUE	$27.95
EASY DRAW	$104.95
1ST WORD	$34.95
NEOCHROME	$34.95
JOUST	$18.95
CP/M EMULATOR	$34.95
FILE & REPORT	$41.95
CHESSMASTER 2000	$32.95
NEW TECH COLOR BOOK	$14.95
TRINITY	$27.95
MEAN 18	$34.95
COMPUTER BASEBALL	$37.95
PHANTASIE	$27.95
MINDWHEEL	$32.95
CHECKMINDER	$52.95
LEADER BOARD	$77.95
FILE	$52.95
DB GRAPH	$52.95
MAJOR MOTION	$27.95
ANIMATOR	$27.95
WINTER GAMES	$27.95
TEMPLE OF APSHAI TRI 1	$27.95
TOURNAMENT DISK-L.B. 1	$13.95
D B MASTER	$34.95
2KEY ACCOUNTING	$34.95
HOME PLANETARIUM	$18.95
DEADLINE	$34.95
STARCROSS	$34.95
ZORK I	$27.95
ZORK II	$32.95
ZORK III	$32.95
WITNESS	$29.95
SUSPENDED	$34.95
PLANEFALL	$27.95
SORCERER	$32.95
SEA STALKER	$27.95
CUTTHROATS	$27.95
HITCHHIKERS GUIDE G	$27.95
SUSPECT	$32.95
WISHBRINGER	$27.95
INFIDEL	$32.95
ENCHANTER	$27.95
VIP PROFESSIONAL	$125.95
MIND FOREVER VOYAGIN	$32.95
ULTIMA II	$41.95
KING'S QUEST II	$34.95
HEX	$27.95
SPELLBREAKER	$34.95
SUNDOG	$27.95
FAHRENHEIT 451	$34.95
AMAZON	$34.95
PERRY MASON:MANDARIN	$34.95
NINE PRINCES	$34.95
DRAGONWORLD	$34.95
HOMEWORK HELPER MATH	$34.95
HOMEWORK HELP/WRITIN	$34.95
WORK PROC W/SPELL CH	$41.95
TREASURE ISLAND	$27.95
HACKER	$32.95
DEGAS	$27.95
BORROWED TIME	$34.95
WINNIE THE POOH	$17.95

No surcharge for MasterCard or Visa

ORDERING INFORMATION: Please **specify system.** For fast delivery send cashier's check or money order. Personal and company checks allow 14 business days to clear. School P.O.'s welcome. **C.O.D. charges are $3.00.** In Continental U.S.A. include $3.00 for software orders. 4% shipping for hardware, minimum $4.00. MasterCard and Visa orders please include card #, expiration date and signature. MI residents please include 4% sales tax. HI. AK. FPO, APO, Puerto Rico and Canadian orders, please add 5% shipping, **minimum $5.00.** All other **foreign orders add 15% shipping, minimum $10.00.** All orders shipped outside the Continental U.S.A. are shipped first class insured U.S. factory warranty. Due to our low prices all sales are final. **All defective returns must have a return authorization number.** Please call (313) 595-0788 to obtain an R.A. # or your return will not be accepted. Prices availability subject to change without notice.



Negotiating a monster-filled maze in Time Bandit.

Another example of the many different screens to be found in Time Bandit.

Time Bandit

If *Major Motion* makes you lose track of time, *Time Bandit* might make you lose your sanity. The attention to detail in this game is phenomenal—over 350K of program code attests to that—and the artwork resembles fine drawings, something noticeably lacking in most game graphics.

Best described as an arcade adventure, *Time Bandit* contains 16 varied puzzles, each with 16 levels of advancing difficulty. The best part is that the difficulty isn't increased merely by speeding up the action, an easy way out that most programmers take. Instead, *everything* in the puzzle changes, including the shape of the maze, the placement of treasures, and the rate of movement. This requires a global change in your strategic thought as well.

All of these scenarios are comprised of various challenges. Most require running through the scene, blasting away at the local creatures, collecting treasures, and avoiding death. But each puzzle is different—there is even a *Pac Man*–style maze to test your mettle.

Each location is represented by an icon on the main map, and moving your character onto the icon will transport you to that place. The title of the location helps you keep track of where you are. Names like "Bomb Factory" and "Ghost Town" give you an idea of what you'll be up against in that situation.

Teasingly Tough

Throughout *Time Bandit*, you can gauge the success of your quest by your progress through the various levels. Level 1A is the first you'll face, and each completed puzzle (1B, 1C, and so on) brings you closer to the ultimate level, 4D, and the game's completion—a goal that may not be humanly possible. I had trouble just getting through each of the contests on level 1A.

You'd think that with a supply of ten characters and a bonus character awarded for every 1,000 points, you could play for quite a while before depleting your reserves. Fat chance. This game is tough, although not so tough as to become discouraging. Just tauntingly, teasingly tough.

Most of the game's quests involve searching for a key (or keys) which will enable you to depart from your present level. Whenever you enter a maze, you can't exit until the task is completed. Along the way, you may choose to pick up the treasures for additional bonus points, but beware—the swarms of nasties and baddies constantly cruising the grounds are just hoping for a shot at you. Some are easily dispatched with your gun, but you must be careful to strike others from a distance. Bombs, for example, can explode in many directions, and standing nearby may cost you a life.

The monsters in this contest are all different and look like little works of art, a testimony to the programming and artistic prowess of the authors. When the monsters are destroyed, they usually go out with some kind of unique farewell—a big cloud of smoke, a sour look, a tiny *POW!* balloon, and so forth. It's like jumping into a comic book and running wild amidst the creatures in their native land.

Split Personality

Three of the lands you visit require text-adventure skills to solve the mysteries. Walking up to a sign, scroll, or character will bring up a screen which gives information or a question for you to answer. The text parser is bare-bones, probably due to the fact that *Time Bandit* doesn't rely on it a great deal.

But this added dimension to the game means that you have to use *all* of your skills to make as much progress as possible. Someone who's good only at arcade games will be somewhat handicapped, as will the player who excels only at text adventures. But overall, *Time Bandit* has a great deal to offer to all gamesters.

Major Motion
Time Bandit
Michtron
576 South Telegraph
Pontiac, MI 48053
$39.95 each

Sundog: Frozen Legacy

Selby Bateman, Features Editor

Requirements: Atari ST with disk drive and color monitor.

Sundog: Frozen Legacy has already gained a solid reputation in Apple II circles as one of the best action/adventure games available. Happily, FTL's Atari ST version is even better.

The ST's excellent color graphics and fast disk drive combine to give this science-fiction adventure the kind of detailed multicolor screens, smooth scrolling, rapid response, and mouse-driven playability it deserves.

A Big World

The game itself is huge. You'll wander through 50 cities on 18 different planets in a dozen star systems. Your star freighter, the *Sundog*, is no small environment in itself: six engineering bays to maintain; a sophisticated navigation system with 3-D star map display, a star system map, and a planetary map; a pilot tactical display with combat status, weapons, shield, and ship condition indicators; and a self-propelled cargo pod for land travel on the different planets.

Learning your way around the ship, and then figuring out how to move around planets and star systems, is as entertaining as it is challenging. But that's just the start. The real fun begins as you undertake the adventure itself.

You've inherited the *Sundog* from the estate of an uncle you never knew. The facts surrounding his death are unclear, but the uncle also bequeathed to you all of his debts. That includes a contract to help build a religious colony on the planet Jondd. You not only have to find the colony, named Banville, but you must also amass all the materials for the colony and then find the cryogenically frozen colonists who are temporarily stored throughout the Drahew Region of star systems.

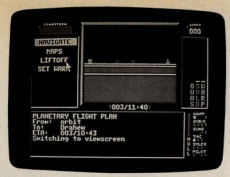
Finding Banville is not easy. Learning the intricacies of trading to get all the building materials is even harder. And, finally, recovering the cryogenic colonists from storage is no snap, either. For these reasons, *Sundog* is a game that's quite difficult to conquer even though it's relatively easy to play.

Pirates and muggers make your task harder still. You've also got to keep your ship in repair and take time out for nourishment and sleep, or you'll collapse from exhaustion or die of starvation. Fortunately, you can save the current status of your game on disk at different stages to resume playing later. If your character is eliminated for some reason, you can go back to the point where you last saved the game and continue.

Life Goes On

Sundog's attention to detail is one of its strongest features. The interior of the ship is colorful, and there's plenty to keep you busy. The cities offer everything from hotels and restaurants to trade exchanges and banks (called *unitellers*). You can go just about anywhere. For example, you can walk into the lobby of a hotel and up to the front desk. The lobby is busy, with customers coming and going. A clerk comes over and asks if you'd like a room. You say yes, and in a moment, you're checked in and ready for a nap. The room looks nice, and it's yours until checkout time. *Sundog* is a realtime game, so life goes on around you even when you're catching up on sleep or moving from planet to planet.

There are also nice touches of humor built into *Sundog*. In one of the first games I played, I made the mistake of illegally parking my cargo pod outside a store. All I needed was to pick up a couple of control nodes, a j-junc module, two photon bridges, and



a scanner flux modulator for my ship. When I came out, the local police had ticketed the cargo pod and I had to go to a uniteller to pay a 90-credit parking fine.

The streets of a city can be unfriendly, so it's wise to keep your sidearm and shield with you while walking about. However, trying to shoot your way out of every potential confrontation is not the best approach. Often, passersby will have information or items to sell you. Learning to spot a good deal and an honest person is a valuable ability. But be cautious, or you could end up out of money and out of luck.

Each time you start a new adventure, you create your character, mixing varying levels of strength, intelligence, charisma, dexterity, and luck. From then on, you're on your own. The game comes with a helpful manual, but you'll have to stumble around at first to find out how things work, what needs to be repaired, where to get new parts, and how to keep yourself in food and fuel. City buildings aren't marked, and it's easy to get lost as you wander around the streets. Take notes on what you need, learn to recognize the shapes of buildings and what each building offers, and draw a few maps to help you remember what you've seen of value.

It's Up To You

One of the more interesting aspects of *Sundog* is that you have to learn a great deal on your own. It's not just a matter of completing tasks that are laid out for you. You have to figure out many of the problems and their solutions. Ingenuity and experimentation are rewarded, although the latter can sometimes result in a short life.

Learning the ins and outs of trading is crucial to your survival and to completing your tasks. Just when you think you're a pro, someone comes along and tricks you, steals your cargo, or worse. But you won't be able to keep your ship running or gather all the goods you need to succeed without becoming a savvy bargainer. You'll also need a bit of luck, but that's part of life for the captain of a star freighter.

Sundog offers you all the elements of a first-rate computer game: a well-planned world within the computer, balanced game play, excellent graphics, and an engaging story line and goal. Originally designed for the Apple II, it's now a worthy addition to your Atari ST software collection.

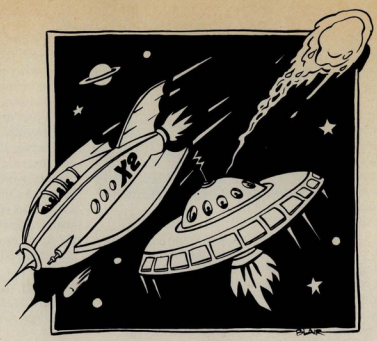
Sundog: Frozen Legacy
Oasis/FTL Games
P.O. Box 112489
San Diego, CA 92111
$39.95

ST

Attention Programmers

Send submissions with source and object code to **COMPUTE!'s Atari ST Disk & Magazine, P.O. Box 5406, Greensboro, NC 27403.**

Or write for our Author's Guide.



AstroPanic!

Charles Brannon, Program Editor

Alien ships weave about, bobbing and diving. Don't let them hypnotize you, though—it's your duty to stop the cosmic horde from achieving total dominance of your monitor screen. This entertaining action game works on any 520ST or 1040ST with either a color or monochrome monitor in all three screen resolutions. Written in compiled C, the program's source code is included on the magazine disk along with the executable game.

Just when you're beginning to think life is a picnic, here they come. That's right—the aliens—strange, wicked creatures from another world (or who knows, perhaps another dimension altogether). They have entered earth orbit, and their six-ship attack squadrons have managed to penetrate earth's orbital defense system, one wave after another. You're earth's last hope, the hottest laser jock yet to graduate from Defense Command's rigorous training program.

Via a video link, you control the massive neutron beam cannon, an instrument of fury that hurls a devastating bolt of matter-shredding energy. Since no mirror system can deflect this beam without itself being destroyed, the neutron cannon is shuttled back and forth at high speed across a magnetic levitation (*maglev*) track. The aliens know that the cannon is too heavily shielded to be attacked by energy weapons, so they use the only tactic possible—a *kamikaze* strike.

The aliens bounce about (in an attempt to evade your shots while calculating the best collision trajectory), then careen in for a confrontation. The experts at Defense Command have anticipated even this mad strategy, so at horrendous cost they have manufactured *three* neutron cannons, each popping up to replace the previous one. But after the three cannons are vaporized, there are no more chances left—the invaders will finally achieve their victory.

Playing AstroPanic!

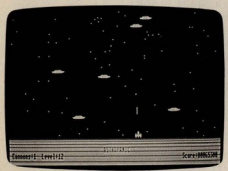
The executable program file for "AstroPanic!" is stored on the magazine disk as PANIC.PRG. Choose the screen resolution you want with Set Preferences, then run PANIC.PRG from the magazine disk menu or from the desktop by double-clicking on the file. The game automatically adjusts itself to the screen resolution you have selected.

AstroPanic! is almost like three games in one. In low resolution, the saucers are large, multicolored objects that reveal lots of detail and crowd the screen. In medium resolution, the objects are smaller and show less detail. Since the saucers are smaller in relation to the screen, the "sky" seems bigger, so the saucers are harder to hit. In high resolution, the saucers are even smaller and the sky is even bigger. If you're lucky enough to have both a color and a monochrome monitor, try playing the game in all three resolutions to experience the different feel.

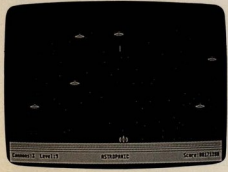
When you run AstroPanic!, a *Let's Play!* button appears in the center of the screen. Click on it and get ready to start shooting—the aliens immediately swarm into action. Use the mouse to move your cannon left and right across the bottom of the screen, and press the *right* mouse button to fire.

The awesome energies of the neutron cannon aren't unleashed casually—you can only fire one bolt at a time. If you see that your shot has already missed, though, you can fire another bolt immediately, canceling the previous shot. Keep your cannon moving (to avoid destruction) and watch out for the edges of the screen, where the saucers ricochet off the sides. When you have destroyed all six ships, you move on to the next level.

"AstroPanic!"—low resolution: The alien saucers are large, finely detailed objects that crowd the screen.



"AstroPanic!"—medium resolution: The saucers are harder to hit because they're smaller in relation to the sky.



"AstroPanic!"—high resolution: On the monochrome screen, the saucers are tiny objects and the sky seems even bigger.

Answer: *1040ST*™

Question: *Which computer is the first in the world to give you 1 Megabyte of power for under $1,000?*

The 1040ST is a major breakthrough in personal computers. Indeed, it's the world's first computer with an original list price that represents less than $1 per kilobyte.

To give you an idea of what an extraordinary accomplishment that is, let's look at the price-per-kilobyte figures for some well-known competitors.

The Macintosh,™ for example, comes in at over $4 per kilobyte, the Amiga™ is over $5 per kilobyte and the PC AT™ is a whopping $9.

In contrast, the 1040ST comes in at an incredible 98 cents per kilobyte and a total price of just $999[95] for the complete system: CPU, disk drive and high-resolution monochrome monitor.

ATARI and 1040ST are trademarks of Atari Corp.
Amiga is a trademark of Commodore-Amiga, Inc.
AT is a trademark of International Business Machines, Inc.
Macintosh is a trademark licensed to Apple Computer, Inc.
© 1986 Atari Corp.

ATARI®

A status display at the bottom of the screen shows the number of cannons remaining, the current level, and your score. The closer the alien is to the bottom of the screen, the more points you get for hitting it. Every time you hit a saucer, the remaining ones move a little bit faster. The speed of the survivors gradually increases as you move through each level of the game, and all of the saucers speed up after each level. By the time you reach level 20, you're bound to lose. Try it; you'll see.

How It Works

The source code for AstroPanic! is included on disk under the filename PANIC.C. You can load the source code into any ASCII text editor or word processor, or display it from the desktop by double-clicking on PANIC.C and selecting SHOW or PRINT. AstroPanic! was developed with the *Megamax* C compiler, but you should be able to modify and compile it with any C compiler and linker with the proper header and library files.

Accompanying this article is a copy of the source code interspersed with extra comments. If you're interested in writing your own ST games or simply learning more about C, you should find the source code very educational.

Perhaps the most remarkable aspect of the game is its speed, considering that it is written in a high-level language. In addition, the executable program is only about 13K long. This is due partly to the exceptional efficiency of the compiler. We examined a disassembly of the compiler's object code and found that it would be difficult to write much better code by hand in machine language. For example, integer variable assignments such as SCORE=0 can translate into a single 68000 instruction. A program such as a game can exploit this efficiency by using only integer math, thus avoiding some of the larger C library modules.

AstroPanic
Annotated Listing

Charles Brannon, Program Editor

```
/* AstroPanic ST by Charles Brannon */
/* created June 12 1986 */
/* last modified June 27 1986 */
```

These are obligatory include files needed to access GEM and AES routines, and to call the standard input/output (STDIO) functions. You may need to use different header files with your C compiler; include only as many as are needed to eliminate undeclared identifier error messages.

```
#include <define.h>
#include <gemdefs.h>
#include <obdefs.h>
#include <osbind.h>
#include <stdio.h>
```

These are the raster operations that can be performed when copying a shape to the screen. They specify how the bits in the source are to be combined with the bits on the screen. XOR is eXclusive OR; REPLACE overwrites the display; and ERASE clears only the part of the display overlapped by one bits in the source image, while TRANSparent sets to 1 only the part of the display overlapped by one bits in the source image, leaving the background alone. REVerse TRANSparent *complements* (flips zeros to ones, and ones to zeros) the source image before merging the source with the background.

```
#define fdb_XOR 6
#define fdb_REPLACE 3
#define fdb_ERASE 4
#define fdb_TRANS 7
#define fdb_REVTRANS 13
```

This macro is used to extract a pseudorandom number from 0 to (x−1), using the XBIOS random number trap.

```
#define rnd(x) (Random( )%(x))
```

Other useful macros for hiding or displaying the arrow pointer. (The pointer doesn't need to be on the screen during the game.)

```
#define HIDE_MOUSE graf_mouse(M_OFF,&dummy)
#define SHOW_MOUSE graf_mouse(M_ON,&dummy)
```

Since we need to adjust the game throughout for color or monochrome mode, this definition makes the source code more readable.

```
#define COLORMODE work_out[35]
```

You can change NUMSPRITES and recompile the game to get more saucers, but this slows down the action. Using six saucers is the best compromise.

```
#define NUMSPRITES 6
```

The height and speed of the missile can also be changed if you want to fine-tune the game. However, since the saucers can move in increments of up to 12 pixels per move, don't make the missile move in too large an increment, or it will sometimes skip over the saucers. The collision routine checks to see whether a saucer is somewhere within the height of the missile, so a tall missile can prevent the fast-moving missile from skipping over a saucer (but also affects the speed of the game). Notice the use of the ternary conditional operator to select one of two values. If the value preceding the question mark is

The ST video hardware doesn't have any provision for sprites. On machines like the Atari 400/800/XL/XE, Commodore 64, and Amiga, sprites greatly simplify game programming (or any programming that employs movable objects). Sprites exist on a separate video plane, so they don't interfere with an underlying background display. Since the video hardware merges the sprites with the video at hardware speed, sprites can be moved quickly without tying up the microprocessor. On the other hand, the 68000 has power to spare—it can easily simulate sprites by virtue of its high-speed memory-moving capabilities.

ST Sprites?

The best way to simulate sprites on the ST would be to write your own routines in machine language. Yet AstroPanic! is written completely in C, using only documented operating system routines. The core of the animation is based on a function called *vrocpyfm( )*, which can be found in the Virtual Device Interface (VDI) library. It's used to copy a rectangular block from one area of memory to another. It can be used to copy one part of the screen to another, or to copy a shape from a memory buffer to any part of the screen.

These memory buffers are supported through a C language structure called a *memory form definition block*, or MFDB. The contents of an MFDB include a pointer to the memory containing the shape data; variables specifying the width, height, and number of bit planes (range of allowable colors) in the shape; as well as a flag specifying whether the format of the shape data conforms to the GEM standard or is machine-specific, using the same memory organization expected by the video hardware.

You can use two methods to animate an object without erasing the background graphics. The first method is to preserve and restore the background as the shape passes over it. Before drawing a shape, save in a buffer the rectangular area that would be overlapped by the shape. When you move the shape to the next position, you then restore the overwritten area from the buffer.

This works fine for one shape or for shapes that don't pass through each other. But imagine what happens when these kinds of shapes do pass over each other. Each shape first saves the image of the shape it overlaps. After the shapes pass through each other, they have both restored the area they overlapped, leaving behind images of the shapes.

Another method relies on a special binary mathematical operation known as *exclusive-OR* (XOR). The binary truth table for XOR is (0 XOR 0=0, 0 XOR 1=1, 1 XOR 0=1, 1 XOR 1=0). If you know something about binary math, you can see that XOR works much like binary OR, or even normal addition—except that when you XOR two 1's together, you get a 0. (Interestingly, binary addition yields the same result, but with a *carry* of 1 that must be added to the bit to the left.) When you copy a shape to the screen, you can specify the way the bits in the shape are combined with the bits in the background image.

A Magic Stamp

Let's use a simple example. On a monochrome ST system, white is represented by 0 and black by 1 (the opposite of most computers—the ST monitor displays its screen in reverse to simplify programming). If you XOR a black shape (1) against a white background (0), you would see the shape 0 XOR 1 as 1 (black). On the other hand, if screen memory was filled with 1's (black), and you attempted to XOR a shape made out of 0's (white), you would see nothing, since 0 XOR 1 is 1 (black).

But notice what happens if you put a black shape against a white background, then copy the black shape back on top of itself.

Answer: *1040ST*™

Question: *Which computer was specially designed for people who hate to wait?*

Let's face it, any time you spend waiting on a computer is time wasted. That's why Atari® built the 1040ST with a sizzling clock speed of 8 MHz.

And with 1024K bytes of Random Access Memory, the ST™ gives you an incredible combination of power and speed. (The PC AT,™ for example, has 512K of memory.)

So you'll spend time working on your ST, instead of waiting on it.

In addition, the 1040ST costs an amazingly low $999[95], which makes it the first computer in the world to deliver 1 Megabyte of memory for under $1,000. (The PC AT costs about $4,500.)

So if you haven't checked out the ST yet, what are you waiting for?

ATARI, ST, and 1040ST are trademarks of Atari Corp.
AT is a trademark of International Business Machines, Inc.
© 1986 Atari Corp.

ATARI®

nonzero (true), the first value before the colon (:) is used as the value of the three-part expression; otherwise, the value following the colon is used. I use this technique in many parts of the source code.

```
#define MISSILE_H (COLORMODE? 8 : 16)
#define MISSILE_SPEED (COLORMODE? 7 : 12 )
```

TOPSCREEN is the top border from which the saucers bounce. TEXTBOX is the size of the score box at the bottom, in pixels (this is doubled in monochrome mode).

```
#define TOPSCREEN 4
#define TEXTBOX 32

/* global variables */
```

These declarations are required for the sake of GEM and the library routines.

```
int dummy,ch,cw;
int work_handle,contrl[12],pxyarray[10];
int intin[128],intout[128],ptsin[128],ptsout[128];
int work_in[11],work_out[57];
```

At least with *Megamax C*, I have to create an MFDB (memory form definition block) "manually." Each shape has an MFDB to specify the bit image and proportions of the object.

```
struct my_fdb
{
    char *fd_addr; /* address of raster */
    int fd_w; /* width in pixels */
    int fd_h; /* height in rows */
    int fd_wdwidth; /* width in words */
    int fd_stand; /* 0 for ST, 1 for standard */
    int fd_nplanes; /* how many planes */
    int fd_r1, fd_r2, fd_r3; /* reserved */
} saucer,screen,cannon;
```

The following comments explain the purposes of these variables.

```
int colortab[16][3]; /* used to save colors */
unsigned long score; /* you know what this is! */
int ships; /* how many cannons are left */
int missile; /* flag for whether missile is in flight or not */
int missile_x,missile_y; /* position of missile in flight */
int cannon_x, cannon_y; /* horizontal & vertical position of cannon */
int xborder,yborder; /* screen boundaries */
int x[NUMSPRITES],y[NUMSPRITES]; /* holds x/y position of sprites */
int xacc[NUMSPRITES],yacc[NUMSPRITES]; /* acceleration factors */
int isdead[NUMSPRITES]; /* is this sprite dead? */
int death_toll; /* saucers shot this round */
int textline; /* line where text box starts */
int round; /* current level of game */
int speed; /* saucer speed */
```

The main( ) routine controls the game. Most routines have meaningful names, so it is pretty easy to follow the "recipe" for AstroPanic!.

```
main( )
{
    int sprite; /* sprite index */
    int prev_x,prev_y; /* stores previous position of a sprite */
    appl_init( );
    init_workstation( );
```

Set screen to black and hide the arrow cursor.

```
    set_colors( );
    HIDE_MOUSE;
```

Fill the screen with stars.

```
    clear_sky( );
```

Wait for the player to click on the alert button to start.

```
    form_alert(1,"[1][AstroPanic!|Charles Brannon|(C) 1986 COMPUTE!][ Let's Play! ]");
```

Load the appropriate shapes.

```
    init_shapes( );
```

Here, we adjust the boundaries of the screen according to workstation width (work_out[0]) and height (work_out[1]). The height and width of the saucer are found in saucer.fd_h and saucer.fd_w, members of the my_fdb structure. The value of work_out[31] is 0 for a color screen or 1 for monochrome, so we can select the position of the text lines in the score box. This figure needs to be doubled (by left-shifting it by one) for the 400-line monochrome mode.

```
    xborder = work_out[0] - saucer.fd_w - 4;
    textline = work_out[1] - ((COLORMODE)? TEXTBOX : TEXTBOX<<1);
    yborder = textline - cannon.fd_h;
```

The first operation is 1 XOR 0=1. When you XOR the black shape on top of itself, though, the operation is 1 XOR 1=0—the shape has removed itself. This method works no matter what the background data is; XOR is a reversible operation.

One way to think of XOR animation is that you're using a rubber stamp inked with a magical negative ink—an ink that reverses the color of whatever it touches. Naturally, stamping twice is the same as not stamping a shape down at all. If you are careful, you can stamp two different shapes so that they overlap. Then, when you restamp these shapes, the background will be completely restored. The only problem is that the area where the shapes overlap is reversed. The 1's in the shapes XOR together in the overlapped area to give white.

It's a little more complicated with a color display, since the XOR is performed on the binary screen data. A binary pattern of 11 XORed with a binary pattern of 10 gives a result of 01. Two different colors, when overlapped, give a third color. Despite this color variation, though, using XOR is fast and effective as a technique for sprite simulation. When shapes are moving quickly, you rarely notice the strange overlap effects.

AstroPanic! uses the XOR method of animation. When a saucer moves, the program draws the shape, moves it to the next position, and erases the old shape. XOR lets us perform this erasure without cutting holes in the background display (the star field).

Simulated Simultaneity

The inner loop of AstroPanic! updates all of the moving objects in the game: the saucers, the cannon, and the missile in flight. Of course, all of these objects aren't really moving simultaneously—they just seem to be because the program alternates the animation quickly and smoothly. First, sprite one gets to move one notch, then sprite two, then sprite three, and so on. Then the program checks to see if the cannon should move and if the missile has been fired.

When a saucer is destroyed, the program stops drawing it, skipping to the remaining shapes instead. Since drawing the shapes consumes the most time, skipping a saucer makes it possible for the others to move that much faster. When there is just one saucer left, it moves six times faster than when there are six saucers being animated. To ease this, the program uses a delay loop to simulate the time taken to draw a shape when one of the saucers is no longer being displayed.

To prevent unsightly flicker, the cannon is redrawn only when it needs to move to a new position. Therefore, the saucers slow down somewhat if you keep moving the cannon. Since drawing and erasing the missile takes some time, too, it can slow down the animation loop. Despite all this, AstroPanic! is still quite effective.

If you are familiar with GEM, and have your GEM reference manuals handy, this program is fairly easy to follow, with a good sprinkling of comments and liberal use of meaningful variable and label names. The program is broken up into a number of small modules, which makes it easier to understand than if the whole game were written as a large main loop. Using these modules is somewhat analogous to using subroutines in BASIC, except every module has its own private variables.

AstroPanic! contains most of the elements found in arcade games, and can serve as a model for your own game programming. Even if you aren't ready to write your own game from scratch, it can be very instructive (and fun) to modify this game, customizing certain details. If you come up with something interesting, send it to us. Perhaps in an upcoming issue a reader will provide us with Super–AstroPanic!. ST

Answer: *1040ST*™

Question: *Which computer builds in multiple features instead of hidden costs?*

It seems that a lot of our competitors design stripped down computers, and then charge extra for every feature and upgrade you add.

Atari® doesn't do that, because we believe the features and level of performance you want should be built in to begin with.

That's why the 1040ST gives you a full Megabyte of memory. While the competition only gives you the chance to spend big dollars trying to improve their memories.

Another trick they use is to make sure their interfaces don't meet industry standards, so you're locked into their system. In contrast, the ST™ uses standard interfaces across the board, such as the RS-232C port for serial modem communications and the parallel interface for an industry standard printer.

Of course, the ST's best built-in is the price, which is an incredible $999[95]!

ATARI, ST, and 1040ST are trademarks of Atari Corp.
© 1986 Atari Corp.

ATARI®

Set the vertical position of the cannon.

```
    cannon_y = yborder;
```

Set the score to zero and the number of cannons to three. Initialize the saucers (ufos) and draw the screen.

```
    reset_game( );
    /* ye olde main loope */
```

FOREVER is defined in *define.h* as *for(;;;)*, an infinite loop (exited with the Terminate( ) call when the game is over).

```
    FOREVER
    {
```

Look for a keypress and pause if one is found.

```
        check_for_pause( );
```

Begin looping for all sprites, updating their positions and images.

```
        for (sprite=0;sprite<NUMSPRITES;sprite++)
        {
```

If a sprite has been shot, we don't draw it, but we do delay a bit via dumdum( ) to simulate the time it would have taken to draw the image. That way the surviving saucers don't speed up too much. However, this delay loop is decreased after each wave, so the survivors gradually move faster and faster. The *continue* keyboard breaks out of the current iteration of the loop, but continues with the statement following the above *for* (as opposed to *break*, which terminates the entire loop).

```
            if (isdead[sprite]) { dumdum( ); continue; }
```

We remember the previous position of the "sprite," since we are about to add in the horizontal and vertical acceleration factors (displacement from current position) in order to move the saucer to the next position.

```
            prev_x=x[sprite]; prev_y=y[sprite];
            x[sprite]+=xacc[sprite];
            y[sprite]+=yacc[sprite];
```

If the position exceeds the screen boundaries, we reverse the direction by negating the displacement, and reset the position to the previous position so that the shape doesn't escape from the screen.

```
            if (x[sprite]<4 | x[sprite]>xborder)
                xacc[sprite]=-xacc[sprite],x[sprite]=prev_x;
            if (y[sprite]<TOPSCREEN | y[sprite]>yborder)
                yacc[sprite]=-yacc[sprite],y[sprite]=prev_y;
```

The first put( ) erases the previous image of the saucer; the second updates it at the new position.

```
            put(&saucer,prev_x,prev_y,fdb_XOR);
            put(&saucer,x[sprite],y[sprite],fdb_XOR);
```

If this sprite is within eight pixels of the saucer, we check to see whether the horizontal positions of the sprite and the saucer overlap. If so, the cannon is destroyed and the loop is canceled with *break*, since we will call *init_ufos( )* within the *kill_cannon( )* routine to reset the saucer positions for the next round.

```
            if (cannon_y-y[sprite]<8)
                if ( (cannon_x>=x[sprite] && cannon_x<=x[sprite]+saucer.fd_w) |
                     (cannon_x+cannon.fd_w>=x[sprite] &&
                      cannon_x+cannon.fd_w<=x[sprite]+saucer.fd_w) )
                {
                    kill_cannon( );
                    break;
                }
        } /* end for */
```

We move the cannon and update the missile only after all the saucers have been updated; otherwise, the saucers would move too slowly. This works just fine, though. The missile is updated only if the missile-is-in-flight flag is active.

```
        move_cannon( );
        if (missile) update_missile( );
    }
}
```

End of the main function and the beginning of the supporting modules.

Terminate( ) ends the game when all the cannons have been destroyed. We restore the colors we've changed, close the virtual workstation, tell AES that we've finished, and exit( ) back to the desktop. The flag allows the calling routine to pass an error value back to the operating system (such as −1 for an emergency exit), but we really don't use this feature in this game.

```
Terminate(flag)
int flag;
{
     reset_colors( );
     v_clsvwk(work_handle);
     appl_exit( );
     exit(flag);
}
```

The endpoint of the loop decreases as the round increases. Unfortunately, I had to use this busy-wait technique, which prevents background programs from running during the delay, since the shortest time waited for by the XBIOS Delay( ) routine (despite the claim of its millisecond resolution) is too long.

```
/* dummy routine, for short delay */
dumdum( )
{
     int i;
     for (i=0;i++<42-(round<<1));
}
```

We need to reset the game the first time it is run and also after the end of a game.

```
reset_game( )
{
     missile=FALSE; /* kill missile */
     clear_sky( );
     speed=2; /* maximum speed */
     score=round=death_toll=0; ships=3;
     update_scorebox( );
     cannon_x=0;
     put (&cannon,cannon_x,cannon_y,fdb_XOR); /* cannon appears */
     init_ufos( );
}
```

We plot a hundred dots (a polyline with two identical coordinates) for a starfield background. Since the XOR animation technique used to move the saucers also preserves the background, it would be a shame not to provide a background to preserve.

```
/* fill sky with stars */
clear_sky( )
{
     int star;
     v_clrwk(work_handle);
     vsl_color(work_handle,1);
     vswr_mode(work_handle,1); /* replace */
     for (star=0;star<100;star++)
     {
          pxyarray[2]=pxyarray[0]=rnd(work_out[0]);
          pxyarray[3]=pxyarray[1]=rnd(work_out[1]);
          v_pline(work_handle,2,pxyarray);
     }
}
/* allow player to pause game by pressing a key */
```

The trick here is that *evnt_multi* normally waits for an event, but we just want to check for a keypress. If there is no keypress, we need to return to the main loop—otherwise the saucers will move only when you have pressed a key. The secret (divulged by Tim Oren in his GEM tutorials) is to wait for both a keypress event *and* a time event. It works because you are waiting for a duration of zero milliseconds, which makes *evnt_multi* return almost immediately. However, if the event that occurred was not a timer event, you know that the other event you were waiting for has happened. It's a little roundabout, but I couldn't find any other way to scan for any key within GEM or AES (I couldn't get *vsm_choice( )* to work, the next best thing). You have to really watch those zeros, though, to make sure the fields you *do* use fall in the right place. If there is a keystroke, we then wait for another keystroke with *evnt_keybd( )* before continuing.

```
check_for_pause( )
{
     int key,which;
     /* poll keyboard by waiting for a null time duration */
     which=evnt_multi(MU_TIMER|MU_KEYBD,0,0,0,0,0,0,0,0,0,0,0,0,&dummy,0,0,
          &dummy,&dummy,&dummy,&dummy,&key,&dummy);
     if (which & MU_KEYBD) evnt_keybd( );
}

/* initializes positions and vectors for saucers */
init_ufos( )
{
     int sprite;
     death_toll=0; /* no sprites dead yet */
```

For each sprite (saucer), we choose a random acceleration (from −*speed* to +*speed*, where *speed* ranges from 2 to 6 or 12). A zero acceleration can't be used, though, or the saucers will move only horizontally, vertically, or not at all (if both xacc and yacc are zero). The saucer is then drawn on the screen so that the animation loop will have a previous image to erase; otherwise, the first put( ) in the animation cycle will leave an image behind.

```
    for (sprite=0;sprite<NUMSPRITES;sprite++)
    {
        isdead[sprite]=xacc[sprite]=yacc[sprite]=0;
        while (xacc[sprite]==0) xacc[sprite]=(speed>>1)-rnd(speed+1);
        while (yacc[sprite]==0) yacc[sprite]=(speed>>1)-rnd(speed+1);
        x[sprite]=8+rnd(xborder-8);
        y[sprite]=8+rnd(yborder-50);
        put (&saucer,x[sprite],y[sprite],fdb_XOR); /* make it appear */
    }
}
```

This one is chock-full of VDI calls. It redraws the score box every time a saucer is hit, so it has to be fast.

```
update_scorebox( )
{
    char temp[20];
    int y,d;
```

First we draw a solid white bar (rectangle) using replace mode.

```
    pxyarray[0]=0; pxyarray[1]=textline;
    pxyarray[2]=work_out[0]; pxyarray[3]=work_out[1];
    vswr_mode(work_handle,1); /* replace */
    vsf_color(work_handle,1); /* white */
    vsf_interior(work_handle,1); /* solid */
    v_bar(work_handle,pxyarray);
```

Now we draw an expanding series of green lines (black in monochrome mode) over the white box, just for looks. (All these colors assume the default color palette.) Transparent mode is initialized so that the background of a text cell doesn't erase the white box or the green lines.

```
    vswr_mode(work_handle,2); /* transparent */
    vsl_color(work_handle,COLORMODE? 3 : 0);
    for (y=d=0;y<TEXTBOX;y+=d++)
    {
        pxyarray[0]=0; pxyarray[1]=y+textline;
        pxyarray[2]=work_out[0]; pxyarray[3]=y+textline;
        v_pline(work_handle,2,pxyarray);
    }
    vsl_color(work_handle,1);
```

That ternary ? operator is really handy. It lets us choose either red or black, depending on the color mode.

```
    /* draw text in red, if possible */
    vst_color(work_handle,COLORMODE? 2 : 0);
```

The title is centered.

```
    vst_alignment(work_handle,1,0,&dummy,&dummy); /* center */
```

The text origin is given as the center of the screen (the width shifted right by one, faster than dividing by two). The variable *ch* is the character cell height, defined by *v_opnvwk( )*. The text is positioned one line lower in monochrome mode so that it isn't drawn through the horizontal line effect.

```
    v_gtext(work_handle,work_out[0]>>1,textline+(ch<<1)+(COLORMODE? 0 :
ch),"ASTROPANIC");
```

In black, we prepare to draw the text with left alignment.

```
    vst_color(work_handle,0); /* draw text in black */
    vst_alignment(work_handle,0,0,&dummy,&dummy); /* left */
```

The sprintf( ) function works like printf( ) except that the output is stored in a string instead of appearing on the display. This makes formatted output possible with any display routine.

```
    sprintf(temp,"Cannons:%d Level:%d",ships,round+1);
    v_gtext(work_handle,cw,textline+ch*3,temp);
```

The score line is right-aligned, and the origin is one character cell width to the left of the screen's right margin.

```
    vst_alignment(work_handle,2,0,&dummy,&dummy); /* right */
    sprintf(temp,"Score:%07lu0",score);
    v_gtext(work_handle,work_out[0]-cw,textline+ch*3,temp);
    vst_alignment(work_handle,0,0,&dummy,&dummy);
    vst_color(work_handle,1);
```

```
}
```

The main loop calls this routine when a saucer collides with the cannon.

```
/* when cannon is hit, kill it */
kill_cannon( )
{
    int lum;
    put (&cannon,cannon_x,cannon_y,fdb_XOR); /* remove cannon */
```

In color mode, we'll turn the screen red, and, after the explosion, gradually decrease the luminance back to black while the explosion sound effect decays.

```
    if (COLORMODE) setcolor(0,1000,0,0); /* flash screen */
    else setcolor(0,1000,1000,1000); /* monochrome */
    explode(cannon_x+(cannon.fd_w>>1),cannon_y+(cannon.fd_h>>1),8,1);
    if (COLORMODE)
        for (lum=1000;lum>=0;setcolor(0,lum--,0,0));
    else setcolor(0,0,0,0);
    missile=FALSE; /* kill missile */
```

One fewer ship now, so we show this by updating the score box display.

```
    --ships; update_scorebox( );
```

The game is over when all the cannons are used up.

```
    if (ships==0) { end_game( ); return; }
    SHOW_MOUSE;
```

One nice touch is that we distinguish between the next and the last cannon, just to remind the player that this is the last chance.

```
    if (ships==1)
        form_alert(1,"[3][!Last Cannon!][Ready!]");
    else
        form_alert(1,"[3][!Next Cannon!][Ready!]");
    HIDE_MOUSE;
```

We reset part of the game by redrawing the screen and setting up new saucer positions. We can't use *reset_game( )* since it sets the score to zero and the number of cannons back to three.

```
    clear_sky( );
    update_scorebox( );
    cannon_x=0;
    put (&cannon,cannon_x,cannon_y,fdb_XOR); /* cannon appears */
    init_ufos( );
}
```

The explosion is used to destroy both the cannon and the saucers. It draws an expanding circle in white, then erases the circle from the inside. Because XOR mode is used, this explosion doesn't erase the background, and it improves the explosion effect, since some of the circles slightly overlap to create the illusion of an expanding shell. The game pauses when the saucer explodes, or else the explosion would be too slow. We need a longer duration for the explosion when the cannon is hit to leave time for the color effect. The sound effect happens in the background, allowing the program to continue immediately after Dosound( ).

```
/* explosion effect radiating from center */
/* flag controls duration of sound effect */
explode(xcenter,ycenter,radius,flag)
int xcenter,ycenter,radius,flag;
{
    static char boom[ ]=
    {0,0, 1,0, 2,0, 3,0, 4,0, 5,0, 6,63, 7,0xf7, 8,0x10, 9,0,
    10,0, 11,0, 12,10, 13,0, 255,0};
    int r;
    boom[25]=flag? 20 : 10;
    Dosound(boom);
    vswr_mode(work_handle,3); /* XOR */
    vsf_interior(work_handle,0); /* hollow circle */
    for (r=0;r<radius;v_circle(work_handle,xcenter,ycenter,r+=2));
    for (r=0;r<radius;v_circle(work_handle,xcenter,ycenter,r+=2));
    vswr_mode(work_handle,1); /* normal */
}
```

At the end of the game, if the player wants to play again, we reset the game. Otherwise, Terminate( ) cancels the program and returns us to the desktop.

```
end_game( )
{
    SHOW_MOUSE;
    if (form_alert(1,"[2][Play Again?][YES|NO]")==1)
    {
        reset_game( );
        HIDE_MOUSE;
```

```
    }
    else Terminate(0);
}
```

We periodically check for cannon movement and missile firing in the main loop.

```
/* moves cannon, checks for fire button */
move_cannon( )
{
    int button,x,y,oldx;
```

The *released* flag is used to make sure the player has released the mouse button before we allow another shot.

```
    static int released=TRUE;
    static char blip[ ]=
    {0,0, 1,0, 2,10, 3,0, 4,0, 5,0, 6,0, 7,0xfd, 8,0, 9,16,
    10,0, 11,0, 12,8, 13,4, 255,0};
```

We save the old position of the cannon and poll the mouse for the horizontal position.

```
    oldx=cannon_x;
    vq_mouse(work_handle,&button,&x,&y);
```

We use the right mouse button to fire the missile. The left mouse button is represented by bit 0, with a value of 1. The right button is bit 1, with a value of 2. (If both buttons are pressed, we'd get 1+2=3). The binary AND masks out the unwanted bits. We also check to see whether the button is released. The *released* flag is set only if the right button test fails. If a missile is fired, *released* is set to 0 (FALSE).

```
    if (button&2)
    {
        if (released)
        {
            if (missile) draw_missile( ); /* erase old missile */
            Dosound(blip);
```

We fire the missile by enabling the missile-is-in-flight flag and giving the missile its horizontal and vertical positions. The first missile is drawn so that the *update_missile( )* routine has something to erase the first time through the loop.

```
            missile=TRUE; missile_x=cannon_x+(cannon.fd_h>>1);
            missile_y=cannon_y;
            draw_missile( );
            released=FALSE;
        }
    }
    else released=TRUE;
```

The horizontal mouse cursor position is used as the cannon's position, although the cannon can be wider than the cursor, so we use only the legal horizontal coordinates.

```
    cannon_x=(x<work_out[0]-cannon.fd_w)? x : work_out[0]-cannon.fd_w;
```

We redraw the cannon only if it has moved, to eliminate flicker.

```
    if (cannon_x != oldx)
    {
        put (&cannon,oldx,cannon_y,fdb_XOR);
        put (&cannon,cannon_x,cannon_y,fdb_XOR);
    }
}
```

This is called in the main loop if a missile is in flight. Each call to this routine moves the missile up one notch (defined by MISSILE_SPEED) and then checks this missile position against all the sprites.

```
/* moves missile to next position, if missile is onscreen */
update_missile( )
{
    int sprite;
    draw_missile( ); /* erase old missile */
```

When the missile reaches the top of the screen, it is turned off; otherwise, we draw it at the new position.

```
    if ((missile_y-=MISSILE_SPEED)>TOPSCREEN)
        draw_missile( ); /* draw new missile */
    else missile=FALSE; /* end of mission */
```

The check for the missile collision seems complicated, but we're just checking to see whether the saucer lies anywhere within the bounds of the missile. Even though this loop executes six times every time the missile moves up one notch, it slows down the animation loop only a little.

```
    for (sprite=0;sprite<NUMSPRITES;sprite++)
    {
        if (!isdead[sprite] && missile_x>=x[sprite] && missile_x<=x[sprite]+saucer.fd_w)
            if ( (y[sprite]>=missile_y && y[sprite]<=missile_y+MISSILE_H) |
                 (y[sprite]+saucer.fd_h>=missile_y &&
                  y[sprite]+saucer.fd_h<=missile_y+MISSILE_H) )
            {
                killsprite(sprite);
                break;
            }
    }
}
```

This routine removes a saucer (sprite) from the screen and from the animation loop.

```
killsprite(which)
int which;
{
    put (&saucer,x[which],y[which],fdb_XOR); /* remove saucer */
    draw_missile( ); /* remove missile */
    missile=FALSE;
```

Explode the poor devil.

```
    explode(x[which]+(saucer.fd_w>>1),y[which],8,0);
    isdead[which]=TRUE;
```

Your score is the vertical position of the saucer when it is hit. However, I see no reason why players in monochrome mode should get twice the score, since there are 400 scan lines in monochrome mode versus 200 in the color modes, so in monochrome mode we right-shift the value by one (which quickly divides by two).

```
    score+=(COLORMODE? y[which] : (y[which]<<1));
    update_scorebox( );
```

We simultaneously increment the counter which keeps track of the number of saucers shot and check to see whether all the sprites have been shot. If so, new saucers are provided, and we move up to the next level of difficulty, to a maximum of 20. The speed of the saucers is allowed to reach 8 pixels per move in color mode or 12 in monochrome mode (since there's more screen real estate to be covered).

```
    if (++death_toll==NUMSPRITES)
    {
        init_ufos( );
        if (round<20)
        {
            round++; /* next round */
            update_scorebox( );
        }
        if (speed<(COLORMODE? 16 : 24)) speed++;
    }
}
```

This is the core routine for updating the missile, a polyline drawn in the XOR mode so that the missile doesn't erase any background.

```
/* draws missile at missile_x, missile_y, with XOR */
draw_missile( )
{
    vswr_mode(work_handle,3); /* XOR drawing mode */
    vsl_color(work_handle,1); /* white */
    pxyarray[0]=missile_x; pxyarray[1]=missile_y-MISSILE_H;
    pxyarray[2]=missile_x; pxyarray[3]=missile_y;
    v_pline(work_handle,2,pxyarray);
    vswr_mode(work_handle,1); /* replace mode */
    vsl_color(work_handle,1); /* black */
}
```

Before we choose custom colors, we save the existing colors in an array so that it can be restored when the game ends.

```
/* Saves colors in global array colortab[ ] */
save_colors( )
{
    int i;
    for (i=0;i<16;i++)
        vq_color(work_handle,i,0,colortab[i]);
}
```

This routine employs a convenient setcolor( ) routine which simplifies palette redefinition.

```
/* sets colors for this program */
set_colors( )
{
```

```
    save_colors( );
    setcolor(0,0,0,0); /* black */
    setcolor(1,1000,1000,1000); /* white */
}
```

It's easier to use this routine than to fill an array every time you want to call *vs_color( )*.

```
setcolor(index,red,green,blue)
int index,red,green,blue;
{
    int rgb_in[3];
    rgb_in[0]=red; rgb_in[1]=green; rgb_in[2]=blue;
    vs_color(work_handle,index,rgb_in);
}
```

The colors are restored when the game is over.

```
reset_colors( )
{
    int i;
    for (i=0;i<16;i++)
        vs_color(work_handle,i,colortab[i]);
}
```

Oops. I forgot to remove these two routines from the program after I no longer needed them. We could have deleted them from this listing, but they're in the disk file as well. Anyway, they're a worthy reminder of any programmer's fallibility. (A good C source code analyzer like *lint* will detect such foibles as unused code.)

```
/* waits for a period of time */
delay(period)
int period;
{
    evnt_timer(period,0);
}
/* returns TRUE if mouse button clicked, else FALSE */
int clicked( )
{
    int pstatus;
    vq_mouse(work_handle,&pstatus,&dummy,&dummy);
    return(pstatus&2);
}
```

This is one of the longest parts of the program, containing the data for the saucers and cannon. Since every screen resolution has different proportions and color capabilities, as well as varying internal memory layout, we need different shapes for every resolution. A program should always support both color and monochrome modes, but needn't work in both low and medium resolution. It's fun to take advantage of the characteristics of these modes, though: low res is the most colorful—with big, detailed ships. The smaller ships in medium res make for the fastest game.

```
/* initializes the shapes according to screen resolution */
init_shapes( )
{
    screen.fd_addr=0; /* screen memory */
    switch (work_out[13]) /* number of colors */
    {
    /* high res, 640 x 400 */
    case 2: ufo_high( );
            cannon_high( );
            break;
    /* medium res, 640 x 200 */
    case 4: ufo_med( );
            cannon_med( );
            break;
    /* low res, 320 x 200 */
    case 16: ufo_low( );
             cannon_low( );
             break;
    }
}
```

Each word of data represents 16 pixels. The raster array is filled; then the *fd_addr* field of the FDB structure is filled in by pointing to the array. This member is defined as a pointer to a *char*, so we cast the array name (which is similar to "pointer to *int*") to avoid compiler warnings.

```
/* initializes data for high-res saucer shape */
ufo_high( )
{
    static int ufohigh[ ]=
    {7,0x8000,0x18,0x6000,0x20,0x1000,0x40,0x800,0x1ff,0xfc00,
    0x1e49,0x27c0,0x7fff,0xfff0,0x8000,8,0x6aaa,0xaab0,0x1d55,
    0x55c0,0x3ff,0xfe00,0,0};
```

```
    saucer.fd_addr=(char *) ufohigh; /* raster memory */
    saucer.fd_w=29; /* width in pixels */
    saucer.fd_h=11; /* height in rows */
    saucer.fd_wdwidth=2; /* width in words */
    saucer.fd_stand=1; /* standard FDB? */
    saucer.fd_nplanes=1; /* one plane */
}
cannon_high( )
{
    static int cannonhigh[ ]=
    {16,0,16,0,16,0,0x38,0,0x54,0,0x306c,0x19c0,0x68aa,0x2df0,
    0x68aa,0x2c08,0xc4ba,0x46b0,0xd3ab,0x97c0,0xc8ba,0x2600,
    0xd6aa,0xd600,0xd6aa,0xd600,0xc8ba,0x2600,0xd3ab,0x9600,
    0xc4ba,0x4600,0x68aa,0x2c00,0x6828,0x2c00,0x307c,0x1800,0x38,0};
    cannon.fd_addr=(char *) cannonhigh; /* raster memory */
    cannon.fd_w=23; /* width in pixels */
    cannon.fd_h=20; /* height in rows */
    cannon.fd_wdwidth=2; /* width in words */
    cannon.fd_stand=1; /* standard FDB? */
    cannon.fd_nplanes=1; /* one plane */
}
```

In medium res, two words are needed for every 16 pixels. If the second word is placed beneath the first word, then each pixel will take its color from the palette register pointed to by the top and bottom bit of each of the 16 columns in the stacked words. The left pixels of the two-pixel pair required to identify a color from 0 to 3 (00, 01, 10, 11) come from the first word, and the right pixels from the second word.

```
/* initializes data for medium-res saucer shape */
ufo_med( )
{
    static int ufomed[ ]=
    {
        0,0,0xf800,0,0,0x701,
        15,8,0xff00,0x100,0,0x800,
        0x1ff,0x1ff,0xfff8,0xfff8,0x28,0x701,
        0xffff,0x5555,0xffff,0x5555,0xf1c0,0x5800,
        0x3fff,0x3fff,0xffff,0xffff,0x8000,0x8000,
        0,0xff,0,0xffe0,0x8b0,0
    };
    saucer.fd_addr=(char *) ufomed; /* raster memory */
    saucer.fd_w=36; /* width in pixels */
    saucer.fd_h=6; /* height in rows */
    saucer.fd_wdwidth=3; /* width in words */
    saucer.fd_stand=0; /* not a standard FDB */
    saucer.fd_nplanes=2; /* two planes */
}
cannon_med( )
{
    static int cannonmed[ ]=
    {    /* plane zero */
        0x40,0, 0x40,0, 0xa0,0, 0x1f0,0, 0x21f0,0x8000,
        0x51f1,0x4000, 0xd7fd,0x6000, 0xdfff,0x6000,
        0xd9f3,0x6000, 0x50a1,0x4000, 0x2000,0x8000,
        /* plane one */
        0,0, 0,0, 0xe0,0, 0x1b0,0, 0x21f0,0x8000,
        0x7111,0xc000, 0xf7fd,0xe000, 0xff1f,0xe000,
        0xf9f3,0xe000, 0x70e1,0xc000, 0x2000,0x8000
    };
    cannon.fd_addr=(char *) cannonmed; /* raster memory */
    cannon.fd_w=19; /* width in pixels */
    cannon.fd_h=11; /* height in rows */
    cannon.fd_wdwidth=2; /* width in words */
    cannon.fd_stand=1; /* standard FDB? */
    cannon.fd_nplanes=2; /* two planes */
    vr_trnfm(work_handle,&cannon,&cannon);
}
```

In low res, it takes four words to define 16 pixels. Again, the words are stacked out, and each column is read top to bottom. The leftmost bit of the color number comes from the first word, and so on.

```
/* initializes shapes for low resolution */
ufo_low( )
{
    static int ufolow[ ]=
    {0,7,0,7,0,0x8000,0,0x8000,
     1,0x1e,0,0x1e,0x8000,0x6000,0,0x6000,
     0,0x3f,0,0x3f,0x4000,0xb000,0,0xb000,
     0,0x7f,0,0x7f,0,0xf800,0,0xf800,
     0x1ff,0,0,0,0xfc00,0,0,0,
     0,1e49,0x1b6,0x1b6,0,0x27c0,0xd800,0xd800,
```

```
    0,0x7fff,0x7fff,0x7fff,0,0xfff0,0xfff0,0xfff0,
    0x8000,0,0xffff,0,8,0,0xfff8,0,
    0x6aaa,0x1555,0x7fff,0,0xaab0,0x5540,0xfff0,0,
    0x1d55,0x1d55,0x1d55,0x1d55,0x55c0,0x55c0,0x55c0,0x55c0,
    0x3ff,0x3ff,0,0,0xfe00,0xfe00,0,0};
    saucer.fd_addr=(char *) ufolow; /* raster memory */
    saucer.fd_w=29; /* width in pixels */
    saucer.fd_h=11; /* height in rows */
    saucer.fd_wdwidth=2; /* width in words */
    saucer.fd_stand=0; /* not a standard FDB */
    saucer.fd_nplanes=4; /* four planes */
}
cannon_low( )
{
    static int cannonlow[ ]=
    {0,16,0,0,0,0,0,0,
     0,16,0,0,0,0,0,0,
     0,16,0,0,0,0,0,0,
     0,0x38,0,0,0,0,0,0,
     0x28,0x54,0,0,0,0,0,0,
     0,0x307c,0x3010,0x3000,0x1c0,0x1800,0x1800,0x1800,
     0x1000,0x78fe,0x7854,0x7800,0x11f0,0x3c00,0x3c00,0x3c00,
     0x1000,0x78fe,0x7854,0x7800,0x1008,0x3c00,0x3c00,0x3c00,
     0x3800,0xfcfe,0xfc44,0xfc00,0x38b0,0x7e00,0x7e00,0x7e00,
     0x2c00,0xefff,0xff55,0xef01,0x69c0,0xee00,0xfe00,0xee00,
     0x3701,0xf7ff,0xff45,0xf701,0xd800,0xde00,0xfe00,0xde00,
     0x2901,0xe9ff,0xff55,0xe901,0x2800,0x2e00,0xfe00,0x2e00,
     0x2901,0xe9ff,0xff55,0xe901,0x2800,0x2e00,0xfe00,0x2e00,
     0x3701,0xf7ff,0xff45,0xf701,0xd800,0xde00,0xfe00,0xde00,
     0x2c00,0xefff,0xff55,0xef01,0x6800,0xee00,0xfe00,0xee00,
     0x3800,0xfcfe,0xfc44,0xfc00,0x3800,0x7e00,0x7e00,0x7e00,
     0x1000,0x78fe,0x7854,0x7800,0x1000,0x3c00,0x3c00,0x3c00,
     0x1000,0x787c,0x7854,0x7800,0x1000,0x3c00,0x3c00,0x3c00,
     0,0x307c,0x3000,0x3000,0,0x1800,0x1800,0x1800,
     0x38,0,0,0,0,0,0,0};
    cannon.fd_addr=(char *) cannonlow; /* raster memory */
    cannon.fd_w=23; /* width in pixels */
    cannon.fd_h=20; /* height in rows */
    cannon.fd_wdwidth=2; /* width in words */
    cannon.fd_stand=0; /* not standard FDB */
    cannon.fd_nplanes=4; /* four planes */
}
```

Here is the primitive for drawing shapes with MFDBs. It fills in the *pxyarray* from the structure, passed through *shape*, a pointer to an MFDB structure. We are copying from (0,0) in the source bitmap (the arrays defined above) to the *x* and *y* position passed through *(xpos, ypos). Copy raster, opaque* is used to do the actual work. This routine is not blindingly fast. If only this routine were rewritten in machine language, the whole game could be speeded up considerably. There is some elegance to a game written entirely in C, however.

```
put(shape,xpos,ypos,mode)
struct my_fdb *shape;
int xpos,ypos,mode;
{
    pxyarray[0]=0; pxyarray[1]=0;
    pxyarray[2]=shape->fd_w-1; pxyarray[3]=shape->fd_h-1;
    pxyarray[4]=xpos; pxyarray[5]=ypos;
    pxyarray[6]=xpos+pxyarray[2];
    pxyarray[7]=ypos+pxyarray[3];
    vro_cpyfm(work_handle,mode,pxyarray,shape,&screen);
}
```

This routine combines all the elements needed to initialize a virtual workstation.

```
init_workstation( )
{
    int i, handle;
    work_handle=handle=graf_handle(&cw,&ch,&dummy,&dummy);
    for (i=0;i<10;work_in[i++]=1); work_in[10]=2;
    v_opnvwk(work_in,&work_handle,work_out);
    if (!work_handle) exit(-1); /* error if we can't open */
}
```

ST

Short-Term

Robert M. Birmingham

This brief program illustrates the ease with which a terminal program may be written on the Atari ST. Although it's a bare-bones program, the C source code is provided on the magazine disk so programmers can experiment and add their own features. The program works on a 520ST or 1040ST in any mode: low- or medium-resolution color and high-resolution monochrome.

"Short-Term" is a simple telecommunications program that emulates a VT-52 terminal and operates in either half or full duplex. It also has an auto-redial function, selectable baud rates, and a linefeed on/off option. Although it's not intended to be a full-blown terminal program, it does provide the basics and can be expanded upon if you wish. The source code is written with Digital Research's *Alcyon C*, which comes with the Atari Developer System Kit.

VT-52 emulation isn't very difficult because all of the necessary functions are part of the ST's operating system. Some of the functions needed for most terminal programs can be found as part of the ST's BIOS (Basic Input/Output System). After explaining how to use Short-Term, we'll cover in detail how it works.

Using The Program

You'll find the executable program file on the magazine disk under the filename SHORTERM.TOS. Notice the .TOS extender; Short-Term does not support drop-down menus and mouse input under GEM (Graphics Environment Manager). If you modify and recompile the C source code (disk filename: SHORTERM.C), be sure to install the new object code with a .TOS filename extender.

Table 1 shows the options in Short-Term which may be toggled on and off by pressing a function key. You can also press the Help key to get a full list of commands available in Short-Term.

Other options not included in Short-Term may be set by first installing the VT-52 desk accessory, which came with your ST on the Language Disk. When the VT-52 emulator is installed, select it from the Desk menu and choose the configure terminal option by pressing the Help key when the title screen appears. Change the settings as required, press the Undo key to return to the GEM desktop, and then run Short-Term. Settings such as parity and start and stop bits which are not selectable from Short-Term can be set in this way with the VT-52 emulator. (See Figure 1.)

Figure 1: VT-52 Terminal Emulator

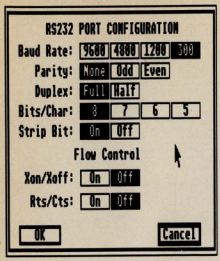


Table 1: Function Key Options

Key	Function
F1	Changes the duplex from full to half or vice versa.
F2	Changes the baud rate from 300 to 1200 or vice versa.
F3	Forces a linefeed upon sensing a carriage return.
F4	Auto-redial feature. Dial the number and then choose this option.
F5	Quit.

For instance, F1 controls the duplex mode. In half duplex, everything you type is echoed back to the screen. If the remote computer also echoes back everything it receives, you'll see every character you type displayed on your screen *ttwwiiccee*. To avoid this problem, switch to full duplex mode. On the other hand, if you don't see your typing displayed on your screen at all, switch to half duplex.

Once you've dialed a phone number using the appropriate dialing code for your modem (for example, ATDT for Hayes-compatibles), you can activate the auto-redial function. Short-Term will redial the number every 30 seconds until it receives a carrier signal. (Note: In compliance with Federal Communications Commission regulations, Short-Term will not redial a number more than 15 times.)

Since Short-Term emulates a VT-52 terminal, there are certain VT-52 keypresses that you can enter to clear the screen, control word-wrap, change the background and foreground colors, and so on. Table 2 is a short list of some of the more useful ones. Esc means to hold down the Esc key while pressing the corresponding key.

Table 2: Keystroke Commands

Keystroke	Action
Esc-A	Move cursor up a line.
Esc-B	Move cursor down a line.
Esc-C	Move cursor to the right.
Esc-D	Move cursor to the left.
Esc-E	Clear the screen and home the cursor.
Esc-H	Move cursor to home position.
Esc-V	Word-wrap on.
Esc-W	Word-wrap off.

How It Works

The following discussion is for C programmers who want to analyze the source code and possibly make modifications.

Short-Term is really a very simple program. To keep the code short, BIOS functions are extensively used. One problem when using BIOS functions, however, is that sometimes they do not return control to the program until they have actually performed their task. For example, if a program calls the BIOS function which gets a character from the keyboard—Cconin( )—the program waits until a character has been received. This could cause a loss of data if the remote computer is sending characters at this time. One solution to this problem is to use some additional functions, also provided by the BIOS, to check the status of the keyboard or auxiliary port.

To read a character from the auxiliary port, check the status of the port by using the function Cauxis( ). If it returns a nonzero value, data is waiting. You can then call Cauxin( ) to get the next character from the port.

Table 3 lists the primary BIOS routines used by Short-Term. By taking advantage of these built-in routines, Short-Term compiles into a much smaller executable program.

Table 3: BIOS Functions Used By Short-Term

Function	Description
Cauxin( )	Gets the next character from the RS-232 port (the modem).
Cauxis( )	Checks the auxiliary port input status. Returns a nonzero value if a character is available at the RS-232 port.
Cauxos( )	Checks the auxiliary port output status. Returns a nonzero value if the RS-232 port is ready to receive a character.
Cauxout( )	Sends a character to the RS-232 port (the modem).
Cconis( )	Checks the keyboard status to see if a character has been entered. Returns a nonzero value if a character is available.
Cconout( )	Writes a character to the screen.
Crawcin( )	Gets a character from the keyboard. All control characters are returned.
Rsconf( )	Configures the RS-232 port.
Setscreen( )	Sets the logical base, physical base, and resolution for the screen.

When you first run Short-Term, the variables must be initialized. These variables include settings for the baud rate, linefeeds, duplex, and so on. The help screen must be initialized too, but here it gets a little tricky. Instead of printing the help screen a character at a time whenever the user presses the Help key, the program prints it once by calling init_-help( ) and then displays it afterwards by page flipping. That is, subsequent presses of the Help key simply change the screen address to point to an area of memory which holds the help screen. The

program then waits for a keypress before returning to the main screen.

After initialization, the main body of the program begins executing. This section consists of two parts: the routine to send a character and the routine to receive a character. The routine to send a character first checks to see if there has been a keypress and if the auxiliary port is ready to send a character. If both of these conditions are true (if both BIOS routines return nonzero values), the program checks whether the key being pressed is a function key. If so, the key is processed accordingly.

If the key is not a function key, the character is sent to the modem. If there was no keypress, a check is made to see whether there is any data coming from the modem. This data could be a message such as NO CARRIER, or a character sent by the remote computer.

Changing Baud Rates

When you press the F2 key, the baud rate is toggled between 300 and 1200 bits per second (bps). If you have a 2400-bps modem, you can change the baud rate settings by altering the values of the parameters passed by the function Rsconf( ). The function Rsconf( ) uses this general form:

Rsconf(*speed, flow control, ucr,* −1, −1, −1)

Speed sets the baud rate according to Table 4.

Table 4: Baud Rate Settings

Speed	Baud Rate
0	19,200
1	9,600
2	4,800
3	3,600
4	2,400
5	2,000
6	1,800
7	1,200
8	600
9	300
10	200
11	150
12	134
13	110
14	75
15	50

Flow control sets the RS-232 port for error checking. Short-Term is initialized with no flow control. Other possible settings are shown in Table 5.

Table 5: Flow Control Settings

Setting	Type of Flow Control
0	No flow control (default value)
1	XON/XOFF
2	RTS/CTS
3	Both XON/XOFF and RTS/CTS

The *ucr* parameter sets the corresponding registers of the 68901 chip. Table 6 shows the meaning of the bit settings.

Table 6: ucr Bit Settings

Bit	Meaning				
0	Not used				
1	Parity (1=even, 0=odd)				
2	Parity enable (1=enabled)				
3, 4	Start/Stop bits:				
	Bit 3	Bit 4	Start bits	Stop bits	Format
	0	0	0	0	Synchronous
	1	0	1	1	Asynchronous
	0	1	1	1.5	Asynchronous
	1	1	1	2	Asynchronous
5, 6	Word length:				
	Bit 5	Bit 6	Word length		
	0	0	8 bits		
	1	0	7 bits		
	0	1	6 bits		
	1	1	5 bits		
7	Clock mode (1=1/16 rate, 0=full speed. Usually set to 1.)				

The −1 values in the parameter list are for other settings which are not useful and don't need setting. The −1 value keeps the Rsconf( ) routine from changing these values.

Auto-Redialing

The remaining routines in Short-Term are called from the main routine. The dial( ) routine automatically redials a number if a busy signal is reached, and continues redialing every 30 seconds for 15 attempts. You can stop the redialing by pressing any key.

What actually dictates-whether a number will be redialed or not is the message received from the modem. The only messages that a Hayes-compatible modem will return after dialing (under normal circumstances) are NO CARRIER and CONNECT. If the redial routine receives a NO CARRIER message, it sends the characters A/ to the modem. This is the Hayes command to repeat the last command sent to the modem. If Short-Term receives a CONNECT message from the modem, control will return to the main body of the program, which handles input and output.

The other routines in Short-Term—message( ) and show_help( )—are used to print a string and to page-flip to the previously drawn help screen.

Modifying Short-Term

All programmers love to tinker with a program. The source code included on the magazine disk is well documented and lends itself to tinkering.

There is something to watch out for, though. Some C compilers will have a problem with the size of the buffer defined for the help screen. One solution would be to compile parts of the program separately, then hook them together with your linker. The source code included on the magazine disk compiled without any problems using *Alcyon C*, but modifications were necessary when using *Megamax C*.

ST

Encryptor

Douglas Wheeler

Do you have sensitive information you want to protect from prying eyes? This utility automatically encodes any type of disk file and locks it with a password of your own choosing. It works on a 520ST or 1040ST in any resolution mode: in low or medium resolution with a color monitor or in high resolution with a monochrome monitor.

Security is an important issue in today's world. In fact, everybody is becoming more security-conscious. Businesses have important documents and contracts which should not be viewed by just anybody hanging around the office. Lawyers, doctors, and other professionals have to guarantee their clients complete confidentiality. Even the average teenager may have letters or diaries which he or she would like to keep secret. If you've refrained from storing something with a computer for any of these reasons, this program is for you.

"Encryptor" is a utility which will encode and decode any ST disk file, whether it is your latest programming creation, a database file, a text file made with your favorite word processor, or even your mother's secret recipes. Once a file is encrypted, no one can decrypt it without entering the correct password or cracking the code.

Using Encryptor

You'll find two files on this month's disk for Encryptor. ENCRYPT.TOS is the executable program file, and ENCRYPT.C is the source code for the curious. Encryptor is written in C, and although the file on disk is in *Megamax C* format, the source code is fairly generic and should work with most versions of C with little or no modification. (More on this below.)

Encryptor is very easy to use. Simply double-click on the ENCRYPT.TOS icon to run the program, then answer the two questions it asks you. But before you begin, there are a few things you should know. First, the file you wish to encrypt must be on the disk in the default drive, the same drive which contains ENCRYPT.TOS. Second, there must be enough room on the disk for a copy of the file. Third, the file must not be in a folder; it must be on the disk's root directory.

If the file is in a folder, the easiest way to move it to the root directory is to drag the file's icon to the drive icon. This makes a copy of the file outside of any folders. The original copy of the file remains in the folder, however, so you should delete it if you want to make sure no unencrypted version of the file remains on the disk.

When you run Encryptor, the first question it asks is the name of the file you wish to encrypt. Simply type in the filename (wildcard symbols are not allowed) and press Return.

The second question Encryptor asks is the password you wish to choose. The password may be up to 40 characters long and may contain any displayable character except the space. Each file you encrypt can be secured with a different password if you like. But don't forget any of these passwords, because you'll need them to decrypt the files later. To decrypt a file, you simply enter the password at this same prompt after rerunning ENCRYPT.TOS. Either memorize the passwords or write them down and keep them in a safe place.

Because of the nature of the encryption scheme, you can super-encrypt a file by encrypting it more than once using different passwords. To read such a file, you'd have to decrypt it as many times as you encrypted it using the same passwords in reverse order.

How It Works

Although it is a short program, Encryptor is quite complex. If you're interesting in learning how it works, read on; otherwise you already know everything you need to use Encryptor.

Encryptor begins by opening the file you selected for input and creating a temporary file named *qqqq* for output. The program then reads the input file, encrypts the data, and writes the results to the output file one character at a time. When it's done, the program deletes the original file and changes the *qqqq* filename to the name of the original file. Most of these operations are pretty straightforward; obviously, the encrypting is where the action is.

As Encryptor reads each character, it matches that character with a corresponding character in the password. That is, the first character in the file is matched with the first character in the password; the second character in the file is matched with the second character in the password; and so on, repeating the password as many times as necessary to match every character in the file.

For example, if the file you're encrypting is a text file which begins *Now is the time for all good men*, and the password is *Help*, Encryptor would match up the characters as follows:

```
Now is the time for all good men
HelpHelpHelpHelpHelpHelpHelpHelp
```

Now picture, if you will, the complete ASCII character set as a continuous line of characters, repeating indefinitely in each direction. For each character to be encrypted, Encryptor effectively reverses the ASCII character set around the corresponding letter of the password. Using the first letter of our sample file and its corresponding password letter, we would have the following:

```
Original ASCII set:
... A B C D E F G H I J K L M N O P Q R S T U V W X Y Z ...

New ASCII set:
... O N M L K J I H G F E D C B A @ ? > = < ; : 9 8 7 6 ...
```

As you can see, the ASCII character set has been reversed "around" the letter H (the first letter of the password). If you then look up the first letter of the file—N, on the top line—you will see that this corresponds to the letter B, which would be written to the output file. Encryptor would then continue to the next character in the file and reverse the ASCII set around the next letter of the password.

This process repeats until the complete file is encrypted and written to the output file. Encryptor then deletes the original file and changes the output filename from *qqqq* to the original filename.

Incidentally, if you're adapting the source code for Encryptor to a different version of C to make modifications, the last two lines of the program which perform this delete/rename operation are the only lines which may require rewriting. The *filename* is stored in a variable of type char and *qqqq* is a literal string.

Improvements And Modifications

One way Encryptor could be improved would be to add parameter passing to the program. This would make it easier to use when called from a command-line processor shell. Instead of the program prompting you for the filename and password, you would include this information as part of the command which runs the program. For example, Encryptor could be made to emulate the Unix operating system's CRYPT utility, which uses the format

CRYPT [*password*] <*oldname*> *newname*

where *password* is the password you choose, *oldname* is the filename of the original file, and *newname* is the name of the new, encrypted file.

Another possible modification would be to read the file into a buffer in memory and then write out the encrypted data to the original file. This would eliminate the need to have room on the disk for the temporary output file, and would also prevent snoops with disk utilities from examining the disk for traces of the deleted original file. **ST**

ProCopy

ST BACKUP UTILITY $34.95

★ Duplicate virtually any disk

★ ProCopy works with all ST, S/S & D/S floppy disk drives

★ Updating policy: $10 with return

★ Protects against accidental loss of expensive software

★ Not copy protected

You can't back up your investment because copy protection locks you out. **ProCopy** is the key!

Send $34.95 (check or money order). VISA & MC accepted. Add $2.00 shipping & handling in North America. Overseas shipping & handling add $4.50. Dealer inquiries welcome.

VISA MasterCard

PROCO PRODUCTS

P.O. BOX 665, CHEPACHET, RHODE ISLAND 02814 USA

For ordering only, call: (800) 843-1223

For information, call: (401) 568-8459

Crash Analyzer

George Miller, Assistant Technical Editor

Here's a useful utility for advanced programmers: If your ST crashes when running a new program you're writing, run this utility to find out why. It recovers special information preserved in memory by the 68000 microprocessor to give you a report on the cause of the failure. The utility works in all monochrome and color modes on any ST, but some of the information it displays is not visible in the low-resolution color mode.

Sooner or later it happens to everyone. As your carefully crafted program begins to run, bomb icons suddenly appear on the left side of the screen and you find that you've become another victim of a system crash.

But now you can clear away some of that smoke and find out *why* your program crashed the computer. "Crash Analyzer," a utility written in machine language, provides an explanation of those bomb symbols and helps you diagnose what caused your program to fail. It works even after you reboot the computer by pressing the reset button. And it's useful to programmers who are working in practically any language: 68000 machine language, C, Pascal, Forth, and even ST BASIC.

Taking Exception

Since computers do exactly what you tell them to do—not always what you want them to do—the system crash is a fact of life faced by every programmer. In the past, on the eight-bit computers, your only choice was to turn off the power and start over. The computer would refuse to respond to any of your attempts to get its attention.

The designers of the Motorola 68000 family of microprocessors felt the chips should be intelligent enough to recognize when they are heading for a system crash, and take steps to recover from it, if possible. So the designers made that kind of help possible by including a feature called *exception processing*.

Basically, an *exception* is the ability of the chip to interrupt whatever it is doing, do something else, and then return to the original task. Exceptions fall into two categories: those caused by external sources, such as input/output devices, and those caused by internal operations, like programming errors and TRAP instructions. An exception caused by an external source is called an *interrupt*.

Each 68000 exception is processed by a different routine. Pointers to these routines are stored in the first 1024 bytes in memory in a 68000 system. Each pointer, or *vector*, is stored in a long word memory location (four bytes). There are 256 possible vectors, numbered from 0 to 255. Therefore, the address of a vector is the vector number multiplied by four.

Exceptions generated by programming errors use certain vector numbers. Table 1 lists the more important preassigned exception vectors. Table 2 is a list of the vectors used by the ST.

Here's why the vector numbers are important. As the ST begins to crash, the 68000 senses that something is going wrong. Immediately it stores a copy of the values in its address and data registers into an area of low memory. This area of memory usually survives the crash, and is not overwritten when you press the reset button on the rear of the ST to reboot. Then, to alert you that something has gone awry, the ST displays a number of bomb icons near the left side of the screen. The number of bombs corresponds to the vector number of the exception encountered. This information can point you to the right track when you are debugging your programs.

AC/FORTRAN™

Mainframe quality, full feature **ANSI FORTRAN 77** compiler includes: **Debugger,** Linker, Library Manager, Runtime Library, **IEEE** math, and **C** interface. Supports **Complex** numbers, **Virtual** arrays, **Overlays** and Dynamic Linking generates directly executable object code for MC68000. Not copy protected.

AC/BASIC™ - Coming Soon

From the authors of **Microsoft BASIC** compiler for Macintosh, comes AC/BASIC for the Atari ST. Fully compatible with the **Microsoft 68000** interpreter: plus programs written in MS BASIC can easily be ported from other machines. Includes **Debugger, BLOCK IF, CASE** statement, and **STATIC** keyword extensions; executes up **50x** faster than interpreted BASIC. AC/BASIC is the new reference standard for MC68000 based personal computers. Not copy protected.

Telephone orders welcome

absoft

MasterCard VISA American Express Card

Scientific/Engineering Software
4268 N. Woodward, Royal Oak, MI 48072/(313) 549-7111

Atari and ST, trademarks of Atari Corp. Microsoft and MS trademark of Microsoft Corp.

ELECTRONIC ONE*

CALL (614) 864-9994 WRITE P. O. BOX 13428 COLUMBUS, OHIO 43213

ATARI 520 SOFTWARE

PERSONAL PASCAL	44.99
COPY II S.T.	26.99
VIP PROF.	99.99
VIP LIGHT	74.99
MUSIC STUDIO	34.99
UNIVERSE II	47.99
PEGAS II	26.99
PRINTMASTER	26.99
P.M. ART GALLERY	17.99
TYPESETTER	22.99
ST TALK	13.99
SILENT SERVICE	24.99
SUN DOG	24.99
THE PAWN	26.99
BRIDGE	18.99
ROGUE	24.99
TEMPLE APSHAI	24.99
WINTER GAMES	24.99
MACRO ASSEMBLER	54.99
BRATACCUS	32.99
N-VISION	26.99
MAJOR MOTION	26.99
LEADER BOARD	24.99
HIPPO RAM DISK	24.99

ATARI 520 ST

COLOR SYSTEM **748.00**

BLACK & WHITE SYSTEM **468.00**

BOTH SYSTEMS:

- 360K DRIVE
- BUILT IN TOS
- 512 KEYBOARD
- SOFTWARE

EXTRA SGL DRIVE 59.99
w/purchase of system

ATARI 520 ST HARDWARE

SF 354 DISK DRIVE	119.99
SF 314 D/S D/D	199.99
SHD 20A ATARI HARD DRIVE	649.99
SM 124 12" B/W	
HIGH RES MONITOR	119.99
12" MED RES	
R6 B MONITOR	279.99

HOW TO ORDER: CASHIER CHECK, MONEY ORDER, MASTERCARD*, or VISA* (Add 4% for charge cards) . . . NO PERSONAL CHECKS . . . NO C.O.D.'s . . . SHIP-PED U.P.S. PRICES SUBJECT TO CHANGE.
SHIPPING: Prompt one day shipping on in-stock merchandise. Ohio residents add 5.5% sales tax. Add $3.00 on all orders under $100.00. Add $5.00 on all orders over $100.00.
INTERNATIONAL: Actual freight charge on all orders outside the continental United States including A.P.O.

CALL OR WRITE FOR FREE CATALOG

Software Discounters of America

S D of A

Orders Outside PA—1-800-225-7638
PA Orders — 1-800-223-7784
Customer Service 412-361-5291

Open Saturday

- **Free Shipping on orders over $100 in continental USA**
- **No surcharge for VISA/MasterCard**
- **Your card is not charged until we ship**

Your ST Software Connection!

ABACUS BOOKS
ST Gem Prog. Ref.
ST Graphics & Sound
ST Internals
ST Logo
ST Machine Language
ST Peeks & Pokes
ST Tricks & Tips
Call For Low Prices!
*Optional Program Disks are available for all books! Call

ABACUS SOFTWARE
Assem Pro CALL
Forth/MT FOR
ST File Pro LOW
ST Test Pro PRICES
Text Designer

ACADEMY
Typing Tutor $23

ACCESS
Leader Board Golf $25

ACCOLADE
Mean 18 Golf $29

ACTIVISION
Borrowed Time $33
Hacker $29
Hacker 2 $33
Little Computer People $33
Mindshadow $33
Music Studio $39
Paint Works $44

ARTWORX
Bridge $19
Compubridge $19
Mail List $14
Strip Poker $25

BATTERIES INCLUDED
D.E.G.A.S $25
D.E.G.A.S. Elite $49
Home Pak $33
I^S Talk $49

Bgur Portfolio System $129
Paperback Elite Call
Thunder $25
Time Link $33

CENTRAL POINT
Copy 2 $25

ELECTRONIC ARTS
Financial Cookbook $33

EPYX
Championship Wrestling Call
Rogue $25
Temple Apshai Trilogy $25
Winter Games $25
World Games Call

FIREBIRD
Starglider $29
The Pawn $29

FTL/SOFTWARE HEAVEN
Sundog $25

INFOCOM
A Mind Forever Voyaging $29
Ballyhoo $25
Cutthroats $25
Deadline $33
Enchanter $25
Hitchhikers Guide to the Galaxy $25
Infidel $29
Seastalker $25
Spellbreaker $33
Trinity $25
Wishbringer $25
Witness $25
Zork 1 $25
Zork 2 or 3 $29

MARK OF THE UNICORN
Hex $25
PC Intercomm $79
Final Word $79

MEGASOFT
ST Copy $23
X-Rated Graphics Library Disk for Printmaster $16

MICHTRON
Bulletin Board System $49
Calendar $19
DOS Shell $25
Kissed $25
M-Disk $25
Major Motion $33
Mighty Mail $33
Mi-Term $33
Personal Money Mgr. $33
Pro-Football Wizard $25
Soft Spool $25
Time Bandits $25
Utilities $39

MICROPROSE
Silent Service $26

MINDSCAPE
Bratacuss $33
Defender of the Crown Call
King of Chicago Call
Sinband: Throne of the Falcon Call

OSS
Personal Pascal $49
Personal Prologue Call

OMNITREND
Universe 2 $49

ORIGIN
Ultima 3 $39

PENGUIN/POLARWARE
Crimson Crown $14
Oo-Topos $14
Transylvania $14

QUICK VIEW
Zoom Racks $49

REGENT
Regent Base $65

SIERRA ON LINE
Black Cauldron $25
Hint Books Call
Kings Quest 2 $33
Ultima 2 $39
Winnie the Pooh $19

SSI
Phantasie $25

SUBLOGIC
Flight Simulator Call
Jet Call

SYNAPSE
Breakers $29
Mindwheel $29

TELARIUM
Amazon $33
Fahrenheit 451 $33
Nine Princes in Amber $33
Perry Mason: Case of the Mandarin Murder $33

TIMEWORKS
Data Manager $59
Swiftcalc $59
Sylvia Porter's Personal Financial Planner $59
Word Writer $59

UNISON WORLD
Art Gallery 1 $19
Art Gallery 2 $19
Print Master $26

VIP TECHNOLOGIES
VIP Professional $99
VIP Professional Lite $65

XLENT
Rubber Stamp $25
ST Music Box $33
Typesetter $25

ACCESSORIES
Bulk Disks 3½ Call
Compuserve Starter Kit $19
Disk Case (Holds 30) $9
Dow Jones Starter Kit (5 hr) $14
Dust Covers Call
Kraft Joystick $9
Panasonic 1080 Cps Dot Matrix Printer Great Deal
Panasonic 3131 Letter Quality Daisy Wheel Printer Call
Supra 20 Meg Hard Disk Drive Call
Supra 300 Baud Modem w/Omega Terminal $59
Supra 300/1200 Baud Modem w/Omega Terminal $139
Wico Boss $12
Wico Bat Handle $17

Hours: Eastern Time
Mon.-Fri. 9AM-5PM
Sat. 10AM-5PM

P.O. BOX 111327—DEPT. CST—BLAWNOX, PA 15238

***Please Read The Following Ordering Terms & Conditions Carefully Before Placing Your Order:** Orders with cashiers check or money order shipped immediately. Personal & Company checks allow 3 weeks clearance. NO C.O.D.'s! **Shipping: Continental U.S.A.**—Orders under $100 add $3; **free shipping on orders over $100. AK, HI, FPO, APO—add $5 on all orders. Canada & Puerto Rico—add $10 on all orders.** Sorry, no other international orders accepted! **PA residents add 6% sales tax on the total amount of order including shipping charges!** Prices subject to change without notice. **REASONS FOR CALLING CUSTOMER SERVICE—412-361-5291 (1)** Status of order or back order **(2)** If any merchandise purchased within 60 days from S.D. of A. is defective, please call for a return authorization number. Defective merchandise will be replaced with the same merchandise only! NO CREDITS! After 60 days please refer to the manufacturers warranty included with the merchandise & return directly to the manufacturer. **Customer service will not accept collect calls or calls on S.D. of A.'s 800# order lines!**

Exception Errors

Here's a brief explanation of the more commonly encountered types of exceptions. Refer to any good book on 68000 programming for a more detailed explanation.

Resets (exceptions 0 and 1). Vectors 0 and 1 are used when the computer is first powered up (or, on the ST, when the reset button is pressed) to set the initial stack and the program counter.

Bus error (exception 2). This indicates that your program tried to access a nonexistent area of memory.

Address error (exception 3). This may be caused by referencing a long word at an odd-numbered memory address. Long words consist of four bytes and must coincide with even addresses.

Illegal instruction (exception 4). Your program tried to execute an instruction which is not part of the 68000 instruction set. Check your source code for typing errors.

Division by zero (exception 5). Since the 68000 instruction set includes division instructions, a check is made for the mathematically illegal operation of division by zero.

CHK instruction (exception 6). Caused by the CHK instruction.

TRAPV instruction (exception 7). Caused by the TRAPV instruction.

Privilege violation (exception 8). This happens when a privileged instruction is attempted while the 68000 is not in supervisor mode.

Trace (exception 9). Used by many debugger programs to single-step through a program that is being debugged.

Line A emulator (exception 10). A trap for opcodes using the format $A*xxx*.

Line F emulator (exception 11). A trap for opcodes using the format $F*xxx*.

Safe Deposit Boxes

In addition to reading the exception vectors, you can gather even more information about the conditions within the computer at the time of the crash. Before grinding to a halt, the 68000 saves the vital contents of its registers in some special memory locations which act as safe deposit boxes. After a crash which has been handled by an exception, it's possible to examine the information held in low memory (unless the ST is powered down).

The first place to look is at location $0380. If this address contains the magic number $12345678, then the information about the crash is good.

The data registers D0–D7 are saved beginning at location $0384.

The address registers, A0–A6, and the supervisor stack pointer, A7, are stored starting at $03A4.

The exception number is stored at $03C4 as a long word, and the user register is stored at $03C8. Beginning at $03CC is a list of 16 words saved from where the supervisor stack was pointing.

As you can see, when the 68000 crashes, it tries to help you by putting information where you can find it. However, unless you're using a monitor or debugger, it's difficult to examine all of these locations after a crash. That's when Crash Analyzer comes to the rescue.

Run the Analyzer (disk filename: ANALYZER .TOS) immediately after rebooting the computer with the reset button. It informs you which exception was triggered and lists the contents of the data registers (D0–D7) and address registers (A0–A7). It shows you a list of flags which were set at the instant of the crash and displays the starting addresses of both the supervisor and the user stacks. Finally, it shows the contents of the supervisor stack at the time of the crash.

In effect, Crash Analyzer takes a snapshot of the activity in your ST at the moment of the crash, then lets you examine it at your leisure.

This information—plus a good book on 68000 programming—will help you debug your program to correct the condition which triggered the crash. **ST**

Table 1: Preassigned Vectors

Vector	Address	Function
0	$000	Reset initial supervisor stack pointer
1	$004	Reset initial program counter
2	$008	Bus error (nonexistent memory)
3	$00C	Address error
4	$010	Illegal instruction
5	$014	Division by zero
6	$018	CHK instruction
7	$01C	TRAPV instruction
8	$020	Privilege violation
9	$024	Trace
10	$028	Line A emulator
11	$02C	Line F emulator
12–14	$030–$038	Unassigned
15	$03C	Uninitialized interrupt vector
16–23	$040–$05C	Unassigned
24	$060	Spurious interrupt
25–31	$064–$07C	Level 0–7 autovector interrupts
32–47	$080–$0BF	TRAP 0–15 instruction vectors
48–63	$0C0–$0FC	Unassigned
64–255	$100–$3FF	User interrupt vectors

Table 2: Vectors Used by the ST

Vector	Function
10	Line A emulator
26	Level 2 interrupts
28	Level 4 interrupts
33	TRAP #1 GEMDOS
34	TRAP #2 GEM
45	TRAP #13 BIOS
46	TRAP #14 XBIOS

(All unused vectors are available to the programmer.)

Move up to a high paying career Servicing Computers

Get started now by building NRI's complete 16-Bit IBM PC compatible computer

You get it all . . . training for America's fastest growing career opportunity . . . training to service all computers . . . training on a total computer system. Only NRI can give you the well-rounded training you need because only NRI gives you a complete computer system . . . computer, monitor, disk drive, printer, software, even test instruments like a digital multimeter and logic probe. It all adds up to training that builds the knowledge, competence and ability you need to succeed as a computer service specialist.

Get inside the IBM PC compatible Sanyo

As an NRI student, you'll get total hands-on training as you actually build your own Sanyo IBM PC compatible computer from the keyboard up. As you assemble it, you'll perform demonstrations and experiments that will give you a total mastery of computer operations and servicing techniques. You'll do programming in BASIC language—even run and interpret essential diagnostic software.

Understanding you can get only through experience

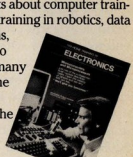
You need no previous knowledge to succeed with NRI. You start with the basics, rapidly building on the fundamentals of electronics with bite-size lessons to master advanced concepts like digital logic, microprocessors, and computer memories.

Do it at home in your spare time

NRI trains you in your own home at your convenience backed at all times by your own NRI instructor and the entire NRI staff of educators and engineers. They're always ready to give you guidance, follow your progress, and help you over the rough spots to keep you moving toward your goal.

100 page free catalog tells more . . . send today

Send the postage-paid reply card today for NRI's 100 page catalog that gives all the facts about computer training plus career training in robotics, data communications, TV/audio/video servicing, and many other fields. If the card is missing, write to NRI at the address below.



NRI SCHOOLS
McGraw-Hill Continuing Education Center
3939 Wisconsin Avenue
Washington, DC 20016
We'll give you tomorrow

BASIC Animatics

Robert G. Geiger

This dazzling demonstration program shows how you can create attractive animation with simple color-cycling techniques. It's written in ST BASIC and runs on any 520ST or 1040ST in the low-resolution color mode only.

Thanks to the graphics commands in ST BASIC, it's possible to use the technique of color cycling to create the illusion of motion on the screen. "BASIC Animatics" (disk filename: ANIMATIC.BAS) demonstrates this technique. Each time the program is run, it draws a different set of randomly generated multicolored ellipses. When the colors of these ellipses are cycled, they appear as spinning rings, collapsing tunnels, or twisting gyroscopes.

ANIMATIC.BAS works only in the low-resolution mode because all 16 colors are needed to achieve the effect. User input is minimal, clearly prompted, and entered entirely with the mouse. Since the program is less than 4K in length, you should have enough memory to load and run it without turning off the buffered graphics option, even on early 520STs without the TOS operating system in ROM.

Illusory Motion

All animation in film and television relies on the phenomenon of *persistence vision*. Our brains perceive movement when our eyes see a progressive sequence of at least nine images every second. When we see fewer than nine images per second, each image fades on the retina before the next image refreshes it. The illusion of movement disappears and we perceive separate images. In practice, to avoid flicker, more than nine images per second are required to achieve smooth animation; TV screens, for example, are refreshed 30 times per second.

Computer graphics animation also relies on persistence vision, and computer animation techniques can be classed into three groups, each with its own special concern for movement. First, the most sophisticated form of animation is realistic simulation. Here, the whole world of movement and all of its facets must be considered in order to convincingly reproduce reality. Examples are seen in animated TV network logos, commercials, and films.

The second type is cartoon animation, where the concern is over the characteristics of movement, which are often exaggerated for effect. Examples of cartoon animation can be found in commercials and children's TV shows.

The third type of computer animation is presentation graphics, where movement is used in a highly abstract way, primarily for the purpose of communicating information. Simple representations of movement add dimension to an illustrator's artwork, better explaining a process or concept that requires both movement and a diagrammatic treatment. Network news and weather bulletins are the main applications, but many educational and training programs also use this technique.

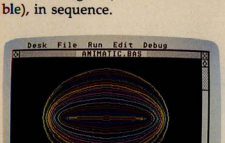
An animatic is the simplest type of presentation graphics. Each animatic consists of several frames, continuously cycled—one-two-three, one-two-three, and so on. Each image is held on the screen long enough to be absorbed by the eye. Alternatively, a single image can present the illusion of movement when it creates a "flow" effect with simple color cycling. This is the technique used by ANIMATIC.BAS.

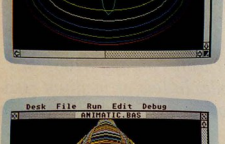
One Frame Per Color

Animation produced by color cycling, though simple, can be very effective for certain applications. The number of "frames" you can create is determined by the number of colors available. In the 16-color low-resolution mode on the ST, you can have a maximum of 15 frames (one color is required for the background). ANIMATIC.BAS uses 14 colors, retaining white (palette 0) for the background and black (palette 1) for the text.

You can create effective animation with far fewer frames. Let's say you're drawing a diagram which shows the flow of a liquid through a pump. You might need only five arrows, placed within the pipe at strategic locations, to show the pattern of the flow. This means you could draw

your diagram in 11 colors and still have enough colors left over to achieve the illusion of a single arrow rushing along the pipeline. Each arrow would change from the color of the pipe to a contrasting color (thus becoming visible), then back again (to become invisible), in sequence.





In the same manner, clouds can be made to move over a weather map, or a graph can grow before your eyes. Ideally, you would use a graphicsdesign program to draw the complicated images. If the program has color-cycling features, you can bring the picture to life.

How It Works

The design of ANIMATIC.BAS is straightforward, with labeled subroutines at the start, followed by the initialization section, the main drawing section, and finally a WHILE-WEND loop to create the animation effects.

When you run ANIMATIC-.BAS, the program checks the current resolution. If the computer is set for medium or high resolution, an alert box pops open to signal that the program must be run in low resolution. Click the mouse on the I'LL CHANGE button to abort the program, then change resolutions.

If the mode is correct, the program jumps to its BEGIN-SHOW routine. Various functions are initialized and the output window is set to full-screen size. The FASTFILL routine then darkens the window for the best contrast. Next, the title of the Output window is changed to ANIMATIC-.BAS using the WINDOWTITLE routine.

In the GETOLD routine, the three decimal intensity values for each palette (one red, one blue, and one green) are PEEKed and

SERVICE

Disk Drive.....35.00
8-Bit Repair.....44.95
ST 520 Repair.....129.00
ST 1040 Repair.....169.00
Monitor Repair.....CALL

PARTS AND DIAGNOSTICS INCLUDED
(FREE Return Freight in the USA,
APO-FPO, C3PO add $10)

(916) 635-3725

SECOND SOURCE ENG.

2664B MERCANTILE DR.
RANCHO CORDOVA, CA 95670

ATARI SERVICE
FACTORY AUTHORIZED NETWORK

MasterCard
VISA

stored in the array OLD% so you can restore the original colors when you're finished.

Next, the main dialog box appears and two options are presented: begin the demonstration or exit the program. The demonstration begins by turning off the mouse cursor and calling a VDI (Virtual Device Interface) routine to avoid interference between the cursor and the graphics. Several variables are randomly set and a series of ellipses is drawn, each in alternating colors (palettes 2–15). This produces a nice string-art picture. After a moment, however, the picture seems to vanish as each palette is changed to black (GOBLACK).

Alternating Ellipses

Now the animation begins. The duration of the display is controlled by the value of the variable LOOP in the WHILE–WEND loop. First, one set of ellipses becomes visible when the color of its palette is changed to green (GOGREEN). Then the palette is changed to black (GOBLACK) and the next palette is changed to green, causing a different set of ellipses to appear. Then *that* palette is changed to black, and the next palette is changed to green, and so on.

When the variable LOOP counts down to zero, all of the palettes are restored to their original colors using the SETOLD routine. Finally, the main dialog box reappears.

ANIMATIC.BAS gives only a sampling of the exciting techniques available in such powerful graphics-design systems as Quantel's *Digital Paint Box* or Aurora Imaging Systems' *Video Graphics*. Most of the paint programs now available for the ST don't offer color cycling, but this feature will be showing up in the more advanced programs soon to come. In time, software will take better advantage of the ST's power and provide more of the higher-level computer graphics techniques. **ST**

Word Count:
A Writer's Accessory

Tony Roberts
Production Director

Here's a compact, efficient desk accessory that quickly counts the number of words in a text file. It's a useful tool for writers and can be summoned in a flash from within any word processor. It works on any 520ST or 1040ST in any resolution mode: in low or medium resolution with a color monitor or in high resolution with a monochrome monitor.

One of a writer's traditional guideposts is the word count. It measures both what has been accomplished as well as what remains to be done. Writing assignments take shape when an editor specifies how many words are expected, and a writer working past deadline often relies on word count references to keep the anxious editor at bay.

Although most computers, the ST among them, can report the number of bytes in a file, writers and editors rarely bandy about byte counts. There are various mathematical methods for converting number of bytes to number of words, but they're rather haphazard. More precise counting methods are often painfully slow.

"Word Count" is a desktop accessory program that solves these problems. It provides a fast, convenient word count for all types of text files—including documents in *ST Writer* and *1ST Word* formats as well as plain ASCII files.

Using Word Count

You'll find Word Count on the magazine disk under the filename WRDCOUNT.AC. Don't try to run it from the disk menu. Instead, copy the file to one of your own boot disks and rename it WRDCOUNT-.ACC. The .ACC extender tells the ST's operating system to install the program as a desk accessory when the computer is switched on. (To in-

stall any desk accessory, you have to boot up the computer by turning on the power; merely pressing the reset button doesn't do it.) Once the computer has been started, it's not necessary to keep the WRDCOUNT .ACC disk in the drive.

After Word Count is installed, it's instantly available from the Desk menu on the GEM desktop. To activate it, simply drop down the Desk menu on the desktop or from within an application program (such as your word processor) and select it.

Once activated, Word Count provides a standard GEM item selector dialog box which lists the files on your disk. The dialog box displays the files in the current folder, but, as usual, you may switch to other folders to locate the desired file. This is accomplished by clicking the mouse on the directory line, editing the line to reflect the desired pathname, then moving the pointer down into the file area and clicking again to inform GEM that you want to view another directory.

Upon receiving a valid filename from the dialog box, Word Count reads through the specified file byte by byte, looking for space characters, tab characters, and line-ending characters, which it assumes are word delimiters. When it finds a delimiter, Word Count increments the counter and continues the search unless the previous character also was a word delimiter, in which case the counter is not incremented.

When the count is complete, another dialog box opens and displays the complete pathname of the file selected and the results of the count. After you click on OK or press Return, Word Count retreats back into the desktop and you can pick up where you left off.

It Has To Be Fast

Word Count works rapidly. The time it takes depends on the length of the text file, of course, but normally it finishes the job in only a few seconds. Because it's a memory-resident desk accessory and because it sets aside a large buffer for reading the text file, time-consuming disk accesses are kept to a minimum. Another advantage of Word Count as a desk accessory is that it's available whenever the Desk menu is displayed—while you are using your word processor, programming in BASIC, or even using a telecommunications program that supports GEM.

One shortcoming of Word Count is that it cannot analyze the document currently in memory until you've stored it on disk. But aside from that, you can ask it to count any disk file, including program (.PRG or .TOS) files—but with these, of course, the results are rather meaningless. It's also possible to count BASIC program files and Pascal or C source files, but because most source code is inconsistently spaced, the results are less than accurate.

Word Count ignores the format lines that are stored at the beginning of *1ST Word* and *ST Writer* document files. If you're using another word processor, you might have to experiment to see what effect (if any) its format lines have on the word count.

If you own an early Atari 520ST that has not yet been upgraded with the TOS operating system in ROM, you may have to rename WRDCOUNT.ACC to DESK5.ACC for it to work properly.

Word Count works with text files stored on floppy disks, hard disks, and RAM disks, but a caution is in order if you're using a single-drive floppy system. Because of an operating system bug, the computer may crash if you attempt to switch to Disk B from the file selector dialog to access a text file on another disk. To circumvent this problem, close the Disk A window, remove the disk from the drive, and insert the disk containing your text file before activating Word Count from the Desk menu.

Word Count was written using *Personal Pascal* from Optimized Systems Software. Portions of this program (the linked libraries) are copyright 1986 by OSS and CCD. Used by permission of OSS.

ST

It's Here!

NEW

BASIC COMPILER

For Atari 520/1040 ST

- Produces fast assembler code
- Atari interpreter compatible
- No line numbers needed
- Supports fully dynamic arrays
- Double-precision floating point numbers fully implemented
- EASY TO USE!

For the BEST professional BASIC Compiler on the market today send:

$119.95 personal check (wait 10 days to ship), money order, cashiers check, VISA and M/C, C.O.D. (add $2.00). (California residents add applicable sales tax.)

— Dealers Welcome —

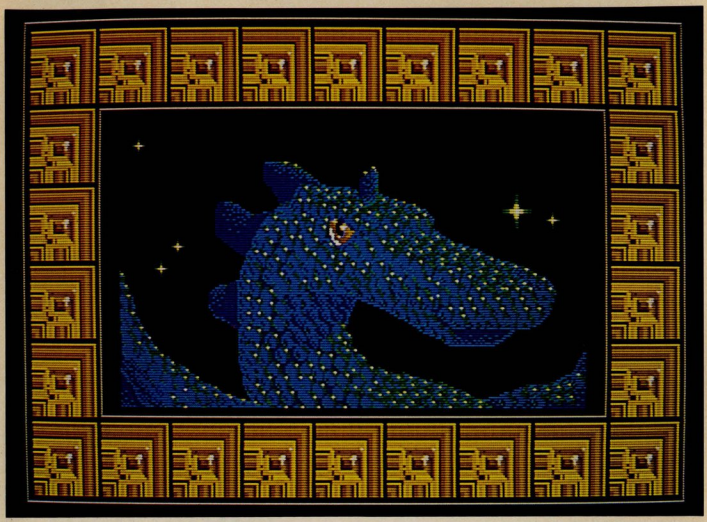
Logical Design Works, Inc.

780 Montague Expwy., Suite 205

San Jose, CA 95131, USA

Telex: 294526 LDW UR

(408) 435-1445



The Azure Dragon

Lee Noel, Jr.

Each issue, COMPUTE!'s Atari ST Disk & Magazine *will feature a screen of computer artwork contributed by an ST artist. The screen is stored on the magazine disk in* Neochrome *format under the filename ART.NEO. It can be loaded into any graphics design program compatible with* Neochrome *files. (Programs to convert* Neochrome *files to other formats are widely available in the public domain.) If you want to contribute a screen of your own, send the disk to* COMPUTE!'s Atari ST Disk & Magazine, *P.O. Box 5406, Greensboro, NC 27403. All artwork must be completely original and previously unpublished in any form. Screens should be drawn in the low-resolution color mode. If you wish, you may include several paragraphs of text describing the artwork and any special techniques employed. We pay $100 for each screen accepted for publication. Artwork accepted for publication becomes the property of* COMPUTE! Publications, Inc.

For this premiere issue of COMPUTE!'s Atari ST Disk & Magazine, *our featured artwork is "The Azure Dragon" by Lee Noel. Noel is the assistant editor for art and design at* COMPUTE! Publications, *and the coauthor (with Features Editor Selby Bateman) of the forthcoming book,* COMPUTE!'s ST Artist. *He created "The Azure Dragon" with* Neochrome *version 0.5, and his main goal (besides meeting an impending deadline) was to demonstrate some unusual techniques possible with the program.*

Although "The Azure Dragon" is not highly detailed or wondrously atmospheric, it does demonstrate a number of *Neochrome*'s impressive and useful features. These are probably best explained in the context of a brief look at the evolution of the artwork.

The dragon originally started out as a great blue reptile, destined to sprawl all over the screen. Initial plans called for a richly textured skin made up of many scales. *Neochrome*'s copy box tool would be used to cut down on the work involved in producing the desired level of skin detail.

To start the process, a small number of individual dragon scales were created—drawn pixel by pixel with the pencil tool. These were shaped and shaded in anticipation of their eventual placement in the overall design. Then, these few basic scales were grabbed by the copy box, repeated, and pasted all over an underlying sketch of the dragon.

For the most part, this went as planned. However, even using the copy box, it became apparent that building up the desired surface for a full-screen figure would still require a great deal of pixel-by-pixel editing. Time was restricted, so a less ambitious design was called for—one that could use the work already completed on the dragon. Naturally, there were already plenty of backup files on disk for the abandoned creature. It was safe, but what to do with it?

Since the dragon's head was virtually complete, one obvious idea was to use it as a central motif, and then to surround it with a decorative border. At this point, the dragon image was reloaded and the head section was placed in *Neochrome*'s buffer using the copy option.

The copy box is also an excellent tool for creating borders. Once a satisfactory element has been designed, all that's needed are a few duplicates placed in a rectangular configuration. That's easy, but how best to show off *Neochrome*?

Neochrome's extraordinary ramp lines provided the answer. Ramp lines enable the artist to effortlessly construct regions of smoothly gradated colors. (The copy box is used to multiply the original ramp line into a usable area.) In this case, a short ramp line was used as the basic building block of two thin, gold-colored rectangles. These were modified and shaded to give the appearance of golden cylinders—one lying horizontally, the other standing vertically. By cutting their ends at 45-degree angles and joining them, a perfect framelike corner was formed.

This was then sliced and recombined with the copy box to create the basic square element visible in the border. A glance at the dragon picture's palette still shows the golden ramp line colors in position between the ramp pointers. (By the way, try color cycling with this setup. The results on the border are fairly impressive.)

At this stage, the dragon's head was recalled from the buffer with the copy box's paste option. The head was carefully aligned and dropped into position, and the border element was tested around it. The element turned out to be a little too large to provide a tightly fitting border.

To fix this, the dragon was cut from the screen to provide working room, and the border segment was shaved down for a better fit. (The contents of the *Neochrome* buffer are preserved unless the cut or copy options are used to put new design work into it.) Once again, the head was dropped into place.

This time, the border worked out well. With a few last touches from the pencil tool, the picture was finished. **ST**

Why C?

Welcome to the C language programming column. In future issues, I plan to cover a variety of topics ranging from general C programming techniques to the ways in which C programmers can take maximum advantage of the many machine-specific features offered by the ST operating system and GEM (Graphics Environment Manager). But since many of you are still unfamiliar with C, I'm going to use this first column to discuss what C is like as a programming language, and why it has become so popular on "next-generation" microcomputers like the Atari ST series and Commodore Amiga.

If you're coming to the ST from the world of eight-bit Atari computers, you may feel that C has emerged from nowhere to become the dominant programming language on the ST. After all, on the older eight-bit machines, BASIC and machine language have been far and away the most popular programming languages, almost to the exclusion of any other languages. BASIC has filled the role of the beginner's language, easy to learn and easy to use because of its interactive nature. Since most home computers come with BASIC built into ROM (Read Only Memory), many people equate making the computer "do something" with giving it commands in BASIC.

Assembly or machine language (ML for short), on the other hand, has become the language of choice for commercial programs on eight-bit micros. Since ML is the only code that the computer really understands without the need for translation, ML programs can be smaller and faster than any other kind. They are, however, much more difficult to write than BASIC programs.

To many current Atari owners, these two languages have proven to be sufficient. So where did C come from, and why, all of a sudden, is it the language that you have to learn in order to program the ST effectively?

Sheldon Leemon

The Balance Of Power

Contrary to what the frustrated programmer might think, C is not a new language that somebody dreamed up to spring on the long-suffering microcomputer hobbyist who has finally mastered BASIC and grudgingly come to terms with ML. Rather, its recent popularity reflects a natural evolution in software that has paralleled the rapid transformation of low-cost computer hardware into high-powered gear like the ST.

Choosing the right programming language for any computer involves weighing tradeoffs between the level of performance that the finished program must meet, and the time required to develop and maintain that program. Programs that are written in ML execute quickly and take up relatively little memory, but they are relatively difficult to write and maintain. It's much easier to program in higher-level languages, but these languages generally don't offer the same performance as ML, and the programs are generally much larger in size than comparable ML programs.

On the older eight-bit computers, there isn't much weighing to do. After all, you can't run a 100K Pascal program on a computer that has only 64K of memory. These machines usually don't have enough memory or mass storage to obtain reasonable performance from programs written in higher-level languages, so ML is more or less a necessity.

Minicomputers and mainframes, on the other hand, have so much power that they can achieve excellent performance from programs written in higher-level languages like FORTRAN and Pascal. As a result, the emphasis is on writing a functional program in a reasonable amount of development time.

The ST computers fit somewhere between these two extremes. They have a much faster microprocessor than do the older micros, along with large mass-storage capacity and lots of main memory. Still, they're not quite in the same league as mainframes.

A Mid-Level Language

That's where C comes in. It represents a good compromise between high-level languages like FORTRAN that are used extensively on mainframes and the machine-level programming that is a must for micros. C supports most of the fundamental features of high-level languages, but it also supports lower-level functions like bit manipulation of memory. In addition, it interfaces easily with ML programs, so time-critical portions of a C program can be written in ML.

Although C may not offer features like sophisticated handling of text strings, it does generate

fairly compact code that executes rapidly enough to achieve good performance. In fact, most of GEM—which controls the windowing environment on the ST—is written not in ML, but in C. (This is another reason that C is the natural choice for those who wish to make use of GEM's features.)

Besides the strictly pragmatic considerations of the size and performance of C programs, the language itself has many features that make it well-suited for software development. It's a modern, *structured* language, the philosophy of which is based on the use of small subprograms called *functions*. Each function is a self-contained program that performs a particular task. The use of functions allows the programmer to break down the overall task into small, manageable pieces. Each piece can be independently tested and debugged, and then incorporated into larger functions that perform more complex tasks.

This modularity not only makes it easier to write and maintain properly working programs, but it also helps to eliminate duplication of effort. For instance, assume you're writing several programs, each of which requires the user to enter some specific kind of information, such as an amount expressed in dollars and cents. You can create one general module that prompts the user and accepts the input, and then include that same module in each of your programs. In fact, libraries of such commonly used modules are available commercially. In C programming, it is quite common to obtain the skeleton of an application like a telecommunications program or database manager from a commercial or public domain source, and then expand and customize the program to meet your own needs.

Portable Programming

Another aspect of C is the kind of output it generates. C compilers create machine language code, just like machine language assemblers. (In fact, some C compilers generate assembly source code that is later assembled.) Such programs generally run faster than those compiled by languages which generate semi-interpreted code or pseudo-code (*p-code*). Furthermore, compiled C programs can be executed without requiring any special support programs or runtime packages, so they're easy to operate and easy to distribute. The programs created by a C compiler can also be fairly compact, since they include only those parts of the language that are actually used in the program.

Finally, C offers a fair degree of portability—you can usually translate a C program to another computer without completely rewriting it. Although there isn't an official standard version of C, in practice, most implementations of the language are very similar.

Of course, programs that include any kind of graphics or windowing generally use very hardware-specific display methods, which makes it harder to convert them for use on computers with different types of display hardware. But by isolating these display routines into a small group of distinct functions, C programmers need to convert only these functions to enable their programs to operate on another machine. This makes it much easier to convert a C program written for the IBM PC to one for the Atari ST (particularly if it uses the GEM interface) than it would be to convert a similarly complex program from IBM BASIC to ST BASIC.

Delayed Gratification

Despite these many advantages, there are still several reasons that newcomers might feel put off or intimidated by C. For one thing, it's a compiled, rather than an interpreted, language. Using an interpreted language like BASIC is an interactive experience. The language has a built-in screen editor, and after you enter a line of code you need only type RUN to see the program execute. Some BASICs (such as eight-bit Atari BASIC) even provide syntax checking on entry, so you get instant feedback if you make a typing mistake. If you run the program and discover it doesn't function properly, you just list the lines that are to blame, make some changes, and run the program again.

Compiled languages such as C are not so easy to use. First, you must compose the source code using a separate text editor. Then you use the compiler program to convert the source code into object code. The compiler may be a single program, or it may be composed of two or more programs, each of which handles a different phase of the compilation process. Finally, you must use a linker program to combine the object code with portions of the C library and convert it into an executable format.

If an error occurs at any stage of this process (due to a typing mistake in the source code, for instance), the whole procedure must be repeated until the program compiles and links successfully. Only then can you actually run the program to determine whether or not it does what you want it to do. If it doesn't, you've got to load up your text editor and try again. This is a far cry from BASIC, where you type PRINT "HELLO", press RETURN, and the computer prints HELLO.

Fortunately, many products have appeared which make the process of generating C programs much less tedious. Very sophisticated text editors are now available, and some can even perform syntax checking so that you don't have to wait until compiling time to discover syntax errors. Many DOS shell programs are available for the ST which let you run batch programs to automate the process of compiling and linking

into a single step. Some will return you to your text editor if the compiling process fails and give you error messages that help you locate the problem.

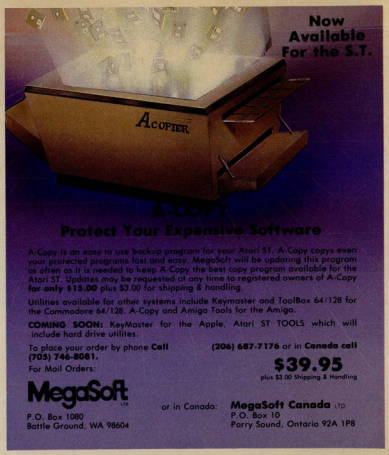
There are even some interpreted versions of C (though not yet for the Atari ST). These allow a programmer to develop programs in an interactive environment that is much more like BASIC. The difference is that once programs have been written using the interpreter, they can then be compiled and run with a speed that BASIC can't touch.

Memory Leverage

Thankfully, the power of the ST computers is a big help in compiling C programs. Even with compiled (rather than interpreted) C, the large amount of memory available in the 520ST and 1040ST provides a luxurious environment in which to work. Using one of many programs available either commercially or in the public domain, you can partition some of that memory as a RAM disk, then run your text editor, compiler, and linker from memory. The amount of time saved by not having to compile and link from floppy disk can dramatically increase your productivity. In fact, some compilers work so fast from a RAM disk that they hardly take longer than interpreters.

The main point to keep in mind is that deep down, a programming language is a programming language. Though C has its eccentricities of syntax and style, it still incorporates the basic concepts of conditional branching (IF-THEN-ELSE), loops (FOR-NEXT, DO-WHILE), and so on. If you've learned to program in BASIC, you can learn to program in C. And once you get used to it, you may really enjoy creating programs that run rings around eight-bit machine language without the tedium of machine language programming.

Now that's what I call "Power without the price." **ST**



THE WAIT IS OVER!

MT C-SHELL™ IS HERE.

MAIN FRAME PERFORMANCE FOR YOUR ST . . .

- Multiuser and Multitasking
- Electronic Mail
- Print Spooling
- Unix™ Like Environment
- And it Runs TOS Programs

$129.95
COMPLETE

Beckemeyer Development Tools

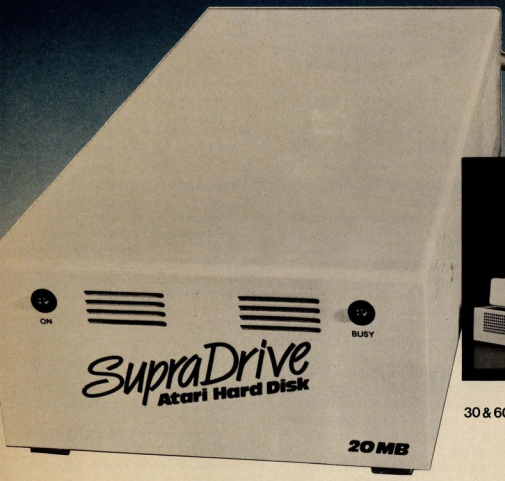
592 JEAN STREET, #304, OAKLAND, CA 94610
415/658-5318

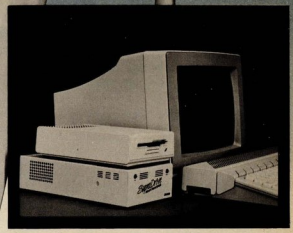
SupraDrive

Atari ST Hard Disk

- 20 MegaByte formatted capacity
- Improves the ST's overall performance (3-20X improvement on disk transfers)
- Atari™ compatible with software and other DMA Buss Devices
- Includes Hard Disk Utilities

Only $799





30 & 60 MB capacity drives are also available.

Supra Corporation

1133 Commercial Way / Albany, OR 97321 USA
Phone: (503) 967-9075 / Telex: 5106005236 (Supra Corp.)

New Products

BASIC Compiler For The ST

Logical Design Works (LDW) is marketing a BASIC compiler for the 520 and 1040 ST computers. The *LDW BASIC Compiler* supports VDIGEM and SYSGEM calls; symbolic labels; dynamic arrays; all file management calls; all basic graphics and sound commands such as WAVE, COLOR, and others; and CALL, BLOAD, and BSAVE functions.

The *LDW BASIC Compiler* recognizes almost all of the ST BASIC Interpreter syntax, with a few exceptions. As a result of compilation, *LDW BASIC* produces an M68000 assembler source file in ASCII format. This file can be assembled using any available assembler (the package includes AS68 assembler), and then linked with any other object files produced with other compilers of any other language. LINK68 linker is provided.

The suggested retail price is $119.95 ($2 postage and handling). For an additional $10, LDW will send an unprotected copy of the program.

Logical Design Works, Inc., 780 Montague Expwy, Suite 205, San Jose, CA 95131.

Circle Reader Service Number 200.

Abacus Personal Productivity Packages

Abacus Software has announced the release of several new productivity packages for the Atari ST.

TextPro is a professional word processor that features multi-column output, automatic indexing and table of contents, fast text input and scrolling, definable function keys, sideways output (to Epson printers), and flexible drivers for other printers. *Text Designer* is a page-making package for creating layouts from *ST Text Pro* or other ASCII word processing files. *DataPro* is a database management system that allows up to 64,000 records of unlimited length, and supports both RAM disk and floppy. *PaintPro* is a design and painting package that uses the GEM interface exclusively, supports three active windows, and has an extensive array of sophisticated drawing tools. And *AssemPro* is a complete assembly language development package; it includes a full screen editor, fast macro assembler and debugger that can also simulate 68020 mode. All packages retail for $49.95, with the exception of *AssemPro*, which is $59.95.

Abacus has also introduced another volume in its ST series of books. *Introduction To MIDI Programming For The Atari ST* is a $19.95 book explaining how to get the most from the ST's MIDI interface. The book also includes the source code for a MIDI editor, driver, and animated player for any of the Casio CZ-series synthesizers.

Abacus Software, P.O. Box 7219, Grand Rapids, MI 49510.

Circle Reader Service Number 201.

Interplanetary Combat Simulation

Starglider, a new fast-action arcade game from Firebird for the ST, puts you in the pilot's seat as you try to destroy an invading fleet from the planet Ergon and annihilate the flagship, Starglider. The game features air-to-air and air-to-ground combat flight simulation, and uses animated 3-D vector graphics with first-person perspective. The program also employs digitized sound and all 16 colors on the Atari ST. Mouse, joystick, or keyboard control is available.

Suggested retail price is $44.95.

Firebird, P.O. Box 49, Ramsey, NJ 07446.

Circle Reader Service Number 202.

New Epyx Entertainment Programs

Epyx has announced that three of its popular entertainment packages, *Temple of Apshai Trilogy*, *Rogue*, and *Winter Games*, are now available in ST versions.

Temple of Apshai Trilogy is actually the three Apshai graphic adventure games in one package. Included are the classic *Temple of Apshai*, and the two popular sequels, *Upper Reaches of Apshai* and *Curse of Ra*. *Rogue* is another popular adventure game in which you search for the Amulet of Yendor, trying to avoid trap doors, deadly darts, sleeping gas, and other insidious obstacles.

Winter Games, which has also been very popular in earlier versions, features seven action Olympic events—bobsledding, ski jumping, figure skating, free-style skating, speed skating, hot dogging, and the biathlon.

Epyx plans to release many of its new titles in Atari ST versions later this year, as well. Prices vary.

Epyx, Inc., 1043 Kiel Ct., Sunnyvale, CA 94089.

Circle Reader Service Number 203.

Ultima III For The Atari ST

Electronic Arts has released an Atari ST version of *Ultima III*, a popular fantasy role-playing game originally written by Lord British and published by Origin Systems. This new version takes advantage of the graphics, sound, and music capabilities of the ST and adds an enhanced user interface as well as full mouse control. The adventure game features 3-D animation such as waves washing against a shore, banners flapping in the breeze, and snarling monster attacks.

Suggested retail price for *Ultima III* is $59.95.

Electronic Arts, 1820 Gateway Dr., San Mateo, CA 94404.

Circle Reader Service Number 204.

ST Golf Simulation Game

Mean 18, a colorful golf simulation game, is the first ST product announced by Accolade Software. Play on one of three famous courses included on disk—Pebble Beach, St. Andrews, and Augusta National—or

$10,000.00

Atari ST Programming Contest!

First Prize $5,000.00 **Second Prize $2,500.00**

Third Prize $1,000.00

Three Honorable Mentions $500.00 each

COMPUTE! Publications, Inc. is looking for the very best original software for the Atari ST series computers. And to prove we're serious, we're offering a total of $10,000.00 in prize money to the top six winners. That's $5,000.00 for First Prize, $2,500.00 for Second Prize, $1,000.00 for Third Prize, and $500.00 each for three Honorable Mentions. In addition, the winners will receive our standard royalties when their programs are published. And even if your program doesn't win a prize, you can still earn purchase fees and royalties if we accept your entry for publication.

Interested? If so, read these rules:

1. Entries must be your original work, previously unpublished. All those whose programs are accepted will be required to affirm this in writing.

2. You can submit as many entries as you want, but we cannot consider programs which have been entered in other contests or submitted for publication elsewhere at the same time.

3. The deadline is October 1, 1986. All entries must be received at our offices by this date. Programs submitted after this date will still be considered for publication, but will not be entered in the contest.

4. Entries are allowed (and encouraged) in virtually all software categories: home and business applications, education, recreation, telecommunications, graphics, sound and music, utilities, and desk accessories.

5. Entries may be written in any programming language—including BASIC, Logo, C, machine language, Pascal, Modula-2, Forth, FORTRAN, and Prolog—as long as they meet two requirements. First, if you're using a compiled language, the compiled object or run-time code must be a self-standing program that can be run by someone who doesn't own a copy of the language. (Exceptions are ST BASIC and Logo. Since these languages come with the ST, it can be assumed that everyone owns a copy.) Second, we must be able to legally distribute the program without incurring licensing fees or other obligations to the maker of the language. If you're not sure whether a certain language qualifies, contact its maker for clarification.

6. Entries must be submitted on a single- or double-sided 3½-inch ST disk with both the run-time code and source code included.

7. Entries must be accompanied by an article which explains how to use the program, what it does, and so on. If your program employs any new or unusual techniques that you think will be of interest to other ST programmers, you can also describe how the program works.

8. Submissions which do not win a prize and are not accepted for publication will be returned only if accompanied by a self-addressed, stamped mailer.

9. All judging will be handled by the staff of COMPUTE! Publications, Inc. All decisions regarding contest entries and acceptances will be solely at the discretion of COMPUTE! Publications, Inc., and all decisions are final. This includes decisions regarding creativity, similarity among entries, and so forth.

10. Winners will be announced by COMPUTE! Publications, Inc. in late 1986.

11. This contest is void where prohibited by law. Full-time, part-time & previous employees of COMPUTE! Publications, Inc., and Capital Cities/American Broadcasting Corporation are ineligible for the contest, but may still submit work for publication at standard rates.

Every Contest Entry Must Contain This Form:

I warrant that the program presently entitled_____

_____ is my own original work and that the work has not been submitted for consideration elsewhere, nor has it been previously published. If my work is accepted by you, I understand that your decision as to the selection of winners and awarding of prizes is final and without recourse on my part. I agree, should you select my submission, to sign your standard contract, which includes assignment of the copyright of the program to COMPUTE!, and to allow you to use my name and image in promotional materials and other forms. (*If you are under age eighteen, your parent or legal guardian must sign for you.*)

Address entries to:
ATARI ST CONTEST
COMPUTE! Publications, Inc.
P.O. Box 5406
Greensboro, NC 27403

COMPUTE! Publications, Inc. abc

Part of ABC Consumer Magazines, Inc.
One of the ABC Publishing Companies

create your own course with the built-in course architect.

You must develop your timing and rhythm in order to hit the ball well. A practice tee is included to help you in those efforts. Different levels of play, including a Pro-mode, are also a part of *Mean 18*. Each tee has a yardage marker stating par and distance to the pin. Distance-to-hole is noted after each shot. And a bird's eye view shows you where you you've been and where you have to go.

Suggested retail price is $49.95.

Accolade, Inc., 20833 Stevens Creek Blvd., Cupertino, CA 95014.

Circle Reader Service Number 205.

New From MichTron

MichTron offers almost two dozen products for the ST, including the following new programs. *KISSED* is a complete debugger with full-screen editor, mini assembler/disassembler, help functions, and other features. *M-Disk* is a RAM-disk emulator that gives you the power of an extra disk drive.

Another utility package, *MichTron Utilities*, repairs damaged disks, restores killed files, modifies disk memory, and saves screens to disk.

The Animator is an applications program that animates graphic pictures for business presentations or for your own enjoyment. It can be used with *Degas* or *Neochrome*.

KISSED, M-Disk, and *The Animator* retail for $39.95. *MichTron Utilities* retails for $59.95.

MichTron, 576 S. Telegraph, Pontiac MI 48053.

Circle Reader Service Number 206.

New ST Products From Activision

Activision recently added several new programs to its Atari ST product line.

Paintworks is a comprehensive paint program and graphics editor that utilizes the sophisticated graphics and color capabilities of the ST. It contains an extensive list of professional-level graphics options, such as clip art, cut-and-paste, and freehand drawing. A multilevel magnification feature is also available for detailed graphics editing. *Paintworks* provides the user with a palette of hundreds of different colors and shades that can be mixed and blended for unique combinations. Designs may also be enhanced with pre-designed lettering and messages for use in things like posters, newsletters, and invitations. The program retails for $69.95.

Hacker II: The Doomsday Papers is a sequel to Activision's earlier hit, *Hacker*. In *Hacker II*, a diabolical plot exists to destroy the United States. This scheme has been uncovered by the government, and you are asked to help collect details on the project known only as "The Doomsday Papers." *Hacker II* retails for $49.95.

Designers at Activision, who had previously discovered little people living inside Commodore and Apple computers and provided "houses on a disk" to lure them out, have now discovered that little people also live inside Atari STs. The Activision *Little Computer People Discovery Kit* lets the user communicate with, play games with, and be entertained by the little person living in his or her computer. The program retails for $49.95.

Activision, Inc., 2350 Bayshore Frontage Rd., Mountain View, CA 94043.

Circle Reader Service Number 207.

Batteries Included Features Several New ST Programs

Batteries Included is shipping several new Atari ST programs. Included are the following:

Thunder! The Writer's Assistant offers three time-saving accessories that work within ST word processing programs. Its spelling checker comes with a 50,000-word dictionary, and allows you to add thousands more words of your choice. It works in realtime mode, to instantly identify errors as you type (the computer beeps to alert you to misspelled words), or lets you check an entire document after you've finished it. The abbreviations expander saves time by automatically expanding an abbreviation you enter to its full form. For instance, if you type *TV*, the program displays *Television*. The document analyzer displays a statistical report of a document you've written, including number of words, average sentence length, and other factors. *Thunder!* retails for $39.95.

Batteries Included has also announced ST conversions of several of its other titles. *I*S Talk* is a full-scale telecommunications package based on the GEM graphics interface. It includes a 50,000 word spelling checker and three levels of macros ($79.95). *HomePak* contains three integrated productivity programs: a word processor, database manager, and terminal software ($49.95). *I*S Degas Elite* is a sophisticated graphics program that offers flexible control over colors, multiple text fonts, and fill and brush patterns ($79.95). And the *Isgur Portfolio System*, designed by Wall Street analyst Lee Isgur, is a powerful investment management program ($199.95).

Batteries Included, 30 Mural St., Richmond Hill, Ontario, Canada L4B 1B5.

Circle Reader Service Number 208.

Timeworks' Personal Productivity Software

ST versions of some of Timeworks' most popular titles are now available. *Word Writer ST* is a powerful 80-column word processing system for home and business use. It offers an 85,000-word spelling checker, built in outliner and five-function calculator, thesaurus, and onscreen highlighting. *SwiftCalc ST* helps develop budgets and business forecasts accurately and effectively, incorporating colorful, high-resolution 3-D graphics and *Sideways* (a sideways printing option). *Data Manager ST* is a complete general information storage and retrieval system that helps with report-writing, business graphics, statistics, label-making, and more.

Each program retails for $89.95.

Timeworks, Inc., 444 Lake Cook Rd., Deerfield, IL 60015.

Circle Reader Service Number 209.

Color Printer For ST

Okidata now offers a cable that allows the Okimate 20 color thermal transfer printer to work with the Atari ST. The Plug 'n Print Module for the ST includes the cable, cartridge ribbons, paper, and instructions.

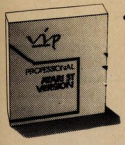
The Okimate 20's 24-element thermal printhead is capable of producing more than 100 colors, and prints text at 80 cps (characters per second) in draft mode or 40 cps in NLQ mode. The printer can facilitate expanded print, boldface, italic, fine print, underlining, superscripts, and subscripts. The Okimate 20 with Plug 'n Print module retails for $268.

Okidata, 532 Fellowship Rd., Mt. Laurel, NJ 08054.

Circle Reader Service Number 210.

ABBY'S DISCOUNT SOFTWARE

VIP PROFESSIONAL



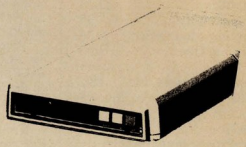
- This integrated program has features and commands identical to LOTUS 1-2-3, including Spreadsheet Analysis and Information Management. Extraordinary graphics with more characters per line, more color & different fonts.

LIMITED — TO THE FIRST TWO HUNDRED ORDERS

RETAIL $179.95

OUR PRICE $59.95

AVATEX MODEM



- 300-1200 Baud
- Standard RS232 Port
- Auto Answer-Auto Dial
- 8 LED Status Indicators
- Compuserve Kit
- Hayes Compatible

ONLY — $89.95

COPY II/ST

- Backup Protected Software Fast
- Copies Many Protected Programs Automatically
- Supports Single and Double Sided Drives
- Requires Atari 520 & 1040 ST Computers - One or Two Drives

ONLY $31.95

HARDWARE & ACCESSORIES

SF 314 3½" DS/DD Disk Drive $199.95

SF 354 3½" SS/DD Disk Drive $159.95

SH 204 20 Meg. Hard Drive $599.95

DISKS (Box of 10)

Sony 3½" DS/DD $29.00

Sony 3½" SS/DD $19.00

Bonus 5¼" SS/DD ..... $ 8.00

Bonus 5¼" DS/DD $10.00

DISKS (Bulk)

Gold Star 3½" DS/DD ..... ea. $1.79

Sony 3½" DS/DD ..... ea. $2.05

100 Quality 5¼" DS/DD ..... ea. $ .69

100 Quality 5¼" SS/DD ..... ea. $ .59

Less Than 100 5¼" DS/DD ..... ea. $ .79

Less Than 100 5¼" SS/DD ..... ea. $ .69

ALL TITLES ON DISK UNLESS MARKED (R) FOR ROM CARTRIDGE

WE CHECK FOR STOLEN VISA & MASTERCARD

Order Line
1-800-282-0333
ORDERS ONLY! Sales Personnel Do Not Have Information on Previous Orders or Product Specs.

M-F 10 a.m.-8 p.m.; Sat. 10-3 ET

37 S. Broad Street, Fairborn, OH 45324

Customer Service & Ohio Residents
1-513-879-9699

AD #CPST-096

VISA

SOFTWARE ONLY — Prepaid orders over $50 receive free shipping via UPS in continental U.S. Please add $2 orders under $50. HARDWARE and all orders requiring shipment via U.S. Post Office are subject to additional freight charges. Add $5 for COD orders. VISA/MasterCard orders add $2 service charge. Ohio residents add 5.5% sales tax. Personal checks require a three-week waiting period. No waiting when paid by credit card, certified check or money order. All items subject to availability and price change. PLEASE CITE AD NUMBER WHEN ORDERING. ALL RETURNS MUST BE ACCOMPANIED BY AN AUTHORIZATION NUMBER.

MasterCard

Advertisers Index

Reader Service Number/Advertiser	Page
102 Abacus Software	BC
103 Abacus Software	12
104 Abby's House of Discount Software	37
105 Abby's House of Discount Software	79
106 Action Software	27
107 Absoft Corporation	63
108 American People Link	11
109 Atari Corp.	43,45,47
110 Basic Bits 'n' Bytes	39
Batteries Included	2
111 Beckemeyer Development Tools	74
112 Central Point Software, Inc.	31
113 CompuServe	1
114 CompuServe	22,23
115 ComputAbility	35
116 Computer Mail Order	33
117 Electronic One	63
118 EPYX	15
119 Firebird Licensees, Inc.	IBC
Logical Design Works, Inc.	69
120 MegaSoft	74
121 MichTron	29
122 MicroProse Simulation Software	4
123 NRI Schools	65
124 Origin Systems Inc.	IFC
125 Proco Products	61
Second Source Eng.	67
126 Software Discounters of America	63
127 Supra Corporation	75

COMPUTE! Books' Atari ST Collection	9
COMPUTE!'s Atari ST Disk & Magazine Subscription	17
COMPUTE!'s Atari ST Programming Contest	77

How To Use The Disk

Every issue of COMPUTE!'s Atari ST Disk & Magazine includes a microfloppy disk bound inside a protective sleeve next to this page. If you experience a problem with the disk, please contact us at 919-275-9809 from 8:30 a.m. to 4:30 p.m. (Eastern Time), Monday through Friday.

To remove the disk, carefully cut the sleeve with a pair of scissors. The sleeve is designed to be tear-resistant; attempting to rip it open may damage the disk.

To use the disk, simply insert it in a drive and click on the appropriate file drawer icon to display the directory window. If you wish, you may boot up your ST with this disk by inserting it in drive A and switching on the computer, but normally it contains no active desk accessories.

There are two ways to access programs and files on the disk. You can simply run or examine the files from the GEM desktop. Or you can use the custom menu program on the disk that contains descriptions of each file as well as special instructions. To run the menu program, double-click on the file named DISKMENU.PRG. It works in all screen modes: low- and medium-resolution color, and high-resolution monochrome.

DISKMENU.PRG displays a directory of files on the disk, one screen at a time. Click on the lower buttons labeled *Prev* or *Next* to display the previous or next screens.

At the top of the disk menu are three buttons labeled *Description, QUIT,* and *Run program.*

The *Description* button calls up a screen which describes the program or file. At the bottom of this screen is the filename and two buttons labeled *MENU* and *RUN.* Clicking on the *MENU* button returns you to the disk menu. Clicking on the *RUN* button loads and runs the program. However, if this particular file is not an executable program (for example, a source code or data file), the *RUN* button is dimmed and disabled.

You can also run a program directly from the disk menu by clicking on the *Run program* button at the upper right. However, if this particular file is not an executable program, you'll be alerted to this fact.

Clicking on the *QUIT* button on the disk menu returns you to the GEM desktop.

Note that many files on the disk require special instructions or explanation; please refer to the corresponding article before attempting to run a program or access a file.

Our disk is not copy-protected. You are encouraged to make a backup of the disk as soon as possible. However, the contents of the disk are copyrighted and may not be used by anyone other than the owner of the magazine. Since the writers and programmers whose work appears on this disk are paid in part according to the volume of sales, we ask that you respect the copyright.

Special Notes

There are four files on the disk which are required for the disk menu program: DISKMENU.PRG, DISKMENU.RSC, MONOMENU.RSC, and CONTENTS.OCT. Do not delete these files if you intend to use the disk menu. If you plan to use the disk menu, be sure these files are copied when you back up the disk.

This issue's disk contains a bonus—a complete listing of all Atari user groups worldwide. It is an ASCII text file which can be viewed or printed from the GEM desktop, a word processor, or a text editor. To view the file from the desktop, double-click on the filename or icon. A dialog box should appear which offers three options: *Show, Print,* or *Cancel.* Click on *Cancel* to return to the desktop. Click on *Print* to make a printout. Click on *Show* to view the list on the screen. You can scroll through the list a line at a time by pressing the Return key, or a screen at a time by pressing the space bar. Press the Control and C keys together to abort the listing and return to the desktop.

The user group list was supplied by the Atari Corporation and updated by COMPUTE! Publications, Inc. Please send updates or additions to our User Group Coordinator, COMPUTE!'s Atari ST Disk & Magazine, P.O. Box 5406, Greensboro, NC 27403.

One final note: The article and disk menu description for "Encryptor" states that the program works in all screen resolution modes. However, in low-resolution color mode, the filename prompt is not completely visible due to the narrow 40-column screen. If you have a color monitor, we recommend switching to the medium-resolution mode before running Encryptor.

STARGLIDER

Starglider – is an all action combat simulation featuring fast moving, animated, 3D vector graphics. The animation and speed of the action is superior to anything yet seen on a home micro. The scenario places the user in the role of a pilot of the only surviving ground attack vehicle left to oppose the alien invaders of the planet Novenia. The mission is to destroy the aliens powerful flagship the Starglider. To achieve the final conflict requires flying skill and battle strategy.

The novella accompanying the product incorporates many clues necessary to survive the increasing levels of combat difficulty and sets the mood for the player as the protector of the human race.

DYNAMIC SPACE COMBAT SIMULATION FOR YOUR ATARI 520ST™

firebird

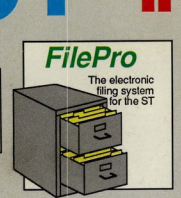
RAINBIRD 68000 RANGE

FACTSHEET/ORDER HOTLINE: 201.934.7373
FIREBIRD LICENSEES INC. P.O. BOX 49, RAMSEY, N.J. 07446
ATARI 520ST IS A TRADE MARK OF THE ATARI CORPORATION

BEST IN SOFTWARE IN BOOKS

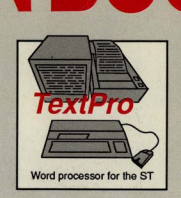
ST PaintPro

A GEM™ among ST drawing programs. Very friendly, but very powerful. A *must* for everyone's artistic or graphics needs. Use up to three windows. Free-form sketching; lines, circles, ellipses, boxes, text, fill, copy, move, zoom, spray, paint, erase, undo, help. **$49.95**



ST FilePro

Simple-to-use, versatile database manager. Help screens; lightning-fast operation; tailorable display using multiple fonts; user definable edit masks; RAM disk support for 1040ST; Multiple files; 64,000+ records. Complete search, sort & file subsetting. Interfaces to TextPro. **$49.95**

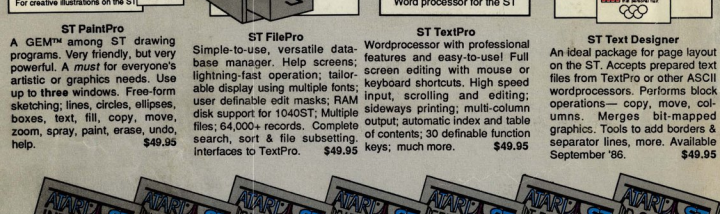


ST TextPro

Wordprocessor with professional features and easy-to-use! Full screen editing with mouse or keyboard shortcuts. High speed input, scrolling and editing; sideways printing; multi-column output; automatic index and table of contents; 30 definable function keys; much more. **$49.95**

ST Text Designer

An ideal package for page layout on the ST. Accepts prepared text files from TextPro or other ASCII wordprocessors. Performs block operations— copy, move, columns. Merges bit-mapped graphics. Tools to add borders & separator lines, more. Available September '86. **$49.95**



ST INTERNALS
Essential guide to learning inside information of the ST. Detailed descriptions of sound and graphics chips, internal hardware, I/O ports, using GEM. Commented BIOS listing. An indispensible reference for your ST library. 450pp. $19.95

GEM Programmer's Ref.
For serious programmers in need of detailed information on GEM. Written with an easy to understand format. All examples are written in C & assembly. Required library addition that no ST programmer should be without. 450pp. $19.95

TRICKS & TIPS
Fantastic collection of programs and info for the ST. Complete programs include: super-fast RAM disk; time-saving printer spooler; color print hardcopy; plotter hardcopy; creating accessories. Filled with money saving programs. 260pp. $19.95

GRAPHICS & SOUND
Detailed guide to understanding graphics & sound on the ST. 2D & 3D function plotters, Moiré patterns, various resolutions & graphic memory, fractals, waveform generation. Examples written in C, LOGO, BASIC and Modula2. $19.95

PEEKS & POKES
Enhance your programs with the examples found within this book. Explores using different languages BASIC, C, LOGO and machine language, using various interfaces, memory usage, reading and saving from and to disk, more. $16.95

ST BASIC to C
Move up from BASIC to C. If you're already a BASIC programer, learn C all that much faster. Parallel examples demonstrate the programing techniques & constructs in both languages. Variables, arrays, pointers, data structure. $19.95

3D GRAPHICS
Fantastic! Rotate, zoom, and shade 3D objects. All programs written in machine language for high speed. Learn the mathematics behind 3D graphics. Hidden line removal, shading. With 3D pattern maker and animator. $24.95

ST and 1040ST are trademarks of Atari Corp.
GEM is a trademark of Digital Research Inc.

Please send me a copy of your latest catalog:

Name:

Address:

City: State: Zip:

Send to: Abacus Software, P.O. Box 7219, Dept. S9, Grand Rapids, MI 49510

Optional diskettes are available for all book titles at **$14.95**

Call **now** for the name of your nearest dealer. Or order directly from ABACUS with your MasterCard, VISA, or Amex card. Add $4.00 per order for postage and handling. Foreign add $10.00 per item. **Other software and books coming soon.** Call or mail the coupon in for your **free** catalog. Dealer inquiries welcome–over 1400 dealers nationwide.

Telex 709-101 - Phone (616) 241-5510

You Can Count On **Abacus Software**

P.O. Box 7219 Dept. S9 Grand Rapids, MI 49510