Home Control

Protect Your Home ***Save On Utilities*** ***Regulate Appliances and More!!***

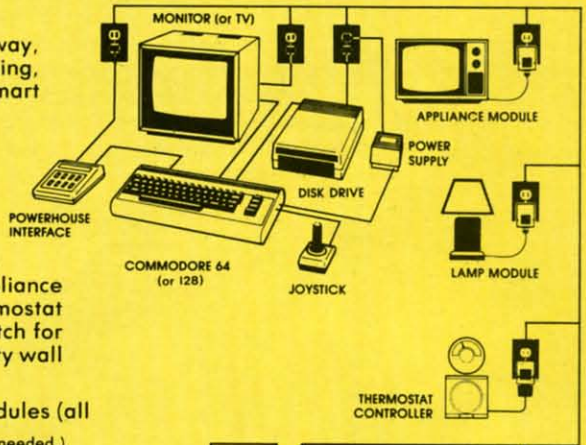
Controls Lights, Appliances, Thermostats, and More Throughout Your Home for Added Convenience, Security, Safety, and Energy Savings. Without Tying Up Your Computer

$49.95 **$49.95**

Free Lamp Module ($19.95 value) with purchase of X-10 computer interface.

The X-10 POWERHOUSE Computer Interface is programmed from your Commodore 64 or 128. You can then turn off your computer or use it for something else. It does not tie up your computer. The software shows a graphical representation of your home and allows you to select a room, then select a light or appliance to be controlled, all just by pointing with the joystick. The interface sends digital signals over your existing house wiring to control your lights and appliances which are connected to plug-in modules (sold separately) List $99.95. **Sale $49.95.**

It can wake you up to stereo or TV news. Light up your bedroom, hallway, bathroom. Start the coffee, start your central heating or air conditioning, warm up the curlers, all before you even get out of bed. But it's smart enough not to wake you up on weekends. At night it can lower the heat, play music or your favorite late night TV show for as long as you want, and can first dim and later turn out the lights automatically.



Many types of modules are available including appliance modules for TV's, hi fi's, coffee pots, etc. Lamp modules which contain a dimmer and can be used for incandescent lamps up to 300 watts. Wall switch modules which also contain a dimmer and can be used for incandescent outside lights and ceiling lights of up to 500 watts. 220V heavy duty appliance modules for 220V air conditioners and water heaters. The thermostat controller for central heating and air conditioning. The 3 way wall switch for controlling incandescent lights operated by two switches. The heavy duty wall receptable module to replace your existing wall receptable, etc.

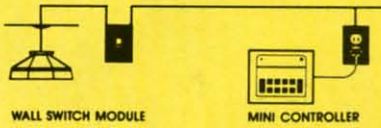
Plus, works with BSR, GE, Leviton, Radio Shack, and Sears Roebuck modules (all 256 codes addressable).

(Price does not include modules, please order modules as needed.)

Control Modules

	List	Sale
Lamp Module — 300 W incandescent	$19.95	**$14.95**
Wall Receptacle — replacement outlets	$24.95	**$19.95**
Wall Switch — turn on/off dim/brighten	$19.95	**$14.95**
3 Way Wall Switch — incandescent lights, for lights controlled by 2 or more wall switches	$24.95	**$19.95**
Thermostat Controller — controls your existing thermostat	$39.95	**$29.95**

C64 & C128



	List	Sale
2 Pin Appliance Module, up to 500W appliances	$19.95	**$14.95**
3 Pin Appliance Module, up to 500W appliances	$19.95	**$14.95**

Add $3.00 for shipping, handling and insurance. Illinois residents please add 6½% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES, EXCEPT CANADA. Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail! Prices & availability subject to change without notice.
VISA — MASTER CARD — C.O.D. C.O.D. on Phone orders only

PROTECTO

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5244 to order

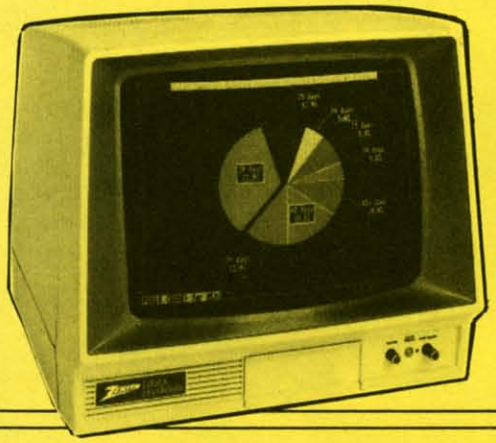
Reader Service No. 116

Computer Cleaners

Your Choice Reg. $19.95 **Sale $9.95**

½ PRICE SALE

Your Choice Reg. $19.95 **Sale $9.95**



TV/MONITOR SCREEN RESTORER & CLEANING KIT

Sale $9.95

Reduce eye fatigue by increasing the clarity of your TV or monitor screen. This kit contains a hard wax formulation to cover surface imperfections on TV screens and monitors. This restores maximum optical clarity, making what you see more distinct. Plus high absorbency cloths and an auto static spray cleaner allows you to clean your screen on a regular basis to keep your screen looking better than new. (This is a must for those who watch monitors or TVs for extended lengths of time.) List $19.95. **Sale $9.95.**

DISK DRIVE CLEANER

Reg. $19.95.
Sale $9.95

- 60% of all drive downtime is directly related to poorly maintained drives.
- Drives should be cleaned each week regardless of use.
- Drives are sensitive to smoke, dust & all micro patricles.
- Systematic operator performed maintenance is the best way of ensuring error free use of your computer system.

This unique twin slot jacket design gives twice as many "wet-dry" cleanings per rotation as other leading brands. Non abrasive, 100% lint free, random fiber cleaners capture dust, smoke particles and disk oxide build up which insures you against disk and data loss from dirty disk drive heads just like you must clean your albums and tape players you must clean your disk drive heads to keep your disk drive working well. (24 cleanings per kit.) List $19.95. **Sale $9.95.**

ANTI-STATIC KEYBOARD CLEANER

Sale $9.95

Now you can clean your computer keyboard fast, efficiently, and safely. The keyboard cleaning solution is exclusively formulated to remove skin oils, dust, and dirt that can destroy your equipment. Plus this non residue solution with anto-static properties will not build up like ordinary household cleaners so you can clean as much as you like without worry. Plus the lint free, high absorbency, non-abrasive cloths will not scratch or mar your equipment as they pick up dirt and grime in a matter of seconds. List $19.95. **Sale $9.95.**

Add $3.00 for shipping, handling and insurance. Illinois residents please add 6½% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES, EXCEPT CANADA. Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail! Prices & availability subject to change without notice.
VISA — MASTER CARD — C.O.D. C.O.D. on Phone orders only

PROTECTO

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5244 to order

Reader Service No. 116

Color Monitor Sale

Sale

16 COLORS

(Premium Quality)

- Built in Speaker & Audio
- For Video Recorders
- For Small Business Computers
- Apple - Commodore -Atari - Aplus 3000 -etc.
- One Year Warranty'

(Premium Quality)

- Beautiful Color Contrast
- High Resolution
- Sharp Clear Text
- Anti-Glare Screen
- 40 Columns x 24 Lines
- Front Panel Controls

List $329.00

Sale $139.95*

Add $14.50 Shipping

80 Columns **RGB** Super High Resolution

13" Color Computer Monitor'

*C64/Atari composite cable $9.95

* C128 RGB/Composite 80 column cable $19.95.

14" RGB & Composite Color Monitor

Allows use of C-128 and C64 mode - composite and 80 column RGB mode. Must be used to get 80 columns in color with 80 column computers. Specially designed for use with the C128's special composite video output, plus green screen only option switch. (add $14.50 shipping)

List $399.00 **Sale $239.95***

12" 80 Column Green/Amber Monitor

Super high resolution composite green or amber screen monitor. 80 columns x 24 lines, easy to read. Fantastic value. Limited Quantities.

List $129.00 **Sale $79.95***

9" Samsung Hi Res Green Screen Monitor

Super High Resolution 80 column monitor perfect for Apple & Laser 3000/128 computers. Fantastic Value. Very Limited Quantities.

List $129.95 **Sale $59.95***

Turn Your Monitor into a TV Set Without Moving Your Computer

Elegant TV Tuner with dual UHF/VHF selector switches goes between your computer and monitor. Includes mute, automatic fine tuning and computer-TV selector switches. Inputs included for 300 ohm, 75 ohm, and UHF. Can be used with cable TV and VCR's. Fantastic Value. Limited Quantities. (Includes loop antenna for UHF & RCA connecting cables) (Add $3.00 Shipping. Plus $3.00 for APO/FPO).

List $129.95 **Sale $49.95**

15 Day Free Trial - 90 Day Immediate Replacement Warranty

• LOWEST PRICES • BEST SERVICE IN U.S.A. • ONE DAY EXPRESS MAIL • OVER 500 PROGRAMS • FREE CATALOGS

Add $10.00 for shipping, handling and insurance. Illinois residents please add 6¼% tax. Add $20.00 for CANADA, PUERTO RICO, HAWAII, and ALASKA orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES, EXCEPT CANADA. Enclose Cashier Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail! Prices & Availability subject to change without notice.

VISA — MASTER CARD — C.O.D. C.O.D. on phone orders only

PROTECTO

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5244 to order

Reader Service No. 116

Rated No. 1 in Computer Education

DesignWare

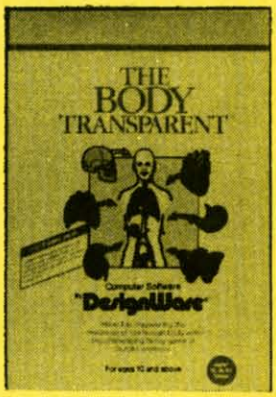
Commodore 64 SALE

Blue Chip

European Nations and Locations — An exploration of the continent that combines important facts and information in a challenging game format. The only European geography program that allows you to add questions of your own. If you miss a question during game play it is recycled to guarantee learning. Each game, you move nations to the proper location on a map of Europe. The map fills in with color as questions are correctly answered. Fast answers are rewarded with bonus points. Winner of the Parents Choice Award. (Disk)
List $44.95 **SALE $26.95**

Tycoon — It takes an individual with loads of confidence and driving ambition, to deal in the commodities marketplace. To foresee market trends and develop a plan of action before trading, are talents that can make or break you as you deal in precious metals, oil, grains and other items of necessity. Armed with charts and details from real-life situations of the past, you can research the market and make your move accordingly. The element of risk makes the game play a real challenge for all aspiring tycoons, or even established entrepreneurs. (Disk)
List $29.95 **SALE$16.95**

The Body Transparent — Learn basic anatomy as you play a game of facts and functions of the human body. Choose a male or female body and move bones and organs to their location. Then learn what the body parts perform. Find out where food is digested, the importance of kidneys and much much more. Add your own biology questions to help make homework fun or to challenge a friend. The game has multiple difficulty levels and bonus points for fast answers, plus color graphics. (Disk)
List $44.95 **SALE $26.95**



Baron — If you were given the opportunity to invest in real estate with a complimentary sum of money, say $35,000 and a time limit of 52 game -weeks to turn that into a million, how would you proceed? Baron is a program that offers you the chance to find out what your chances of success in this complicated venture will be. You must utilize the allocated funds wisely in buying, selling and developing land because real life influences like flunctuating mortage rates and other headaches that affect property owners will be apparent during play. It's challenging and risky to win the title of Baron (Disk)
List $29.95 **SALE $16.95**

States & Traits — A valuable learning tool for geography and history students. Learn basic geography and then go on to find out interesting facts about every region in the country. Find out where historical events happened on the map. As you play quick answers are rewardedwith bonus points. In the "States" part of the program you must move states into the right place on the map and the state lights up in bright color. In "Traits" you identify state capitals and landmarks. You can also design your own games (Disk)
List $44.95 **SALE $26.95**

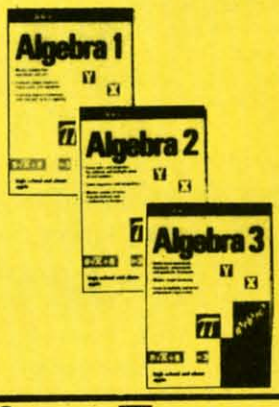
Millionaire — Put yourself in high gear and join the risk takers, because you are about to experience the New York Stock Exchange. To get you started in investments you are given $10,000 and 154 stocks from 5 industry categories, the rest is up to you. Learn the market as you are faced with influences based on true to life events, good judgement is essential to whether you buy or sell and the status of your personal portfolio depends on making the right choices. The play includes and introduction to buying-on-margin, call options, put options and borrowing against your net worth. (Disk)
List $29.95 **SALE $16.95**

Commodore 64 SALE

EduWare

Commodore 64 SALE

Algebra 1,2, or 3 — An introduction to Algebra concepts and definitions. Helps students master the fundamentals of first year Algebra, addition of real numbers, multiplication of real numbers, solving equations, solving inequalities and absolute value sets. Students can learn in 4 ways: They can work the sample problems, read discussions of concepts, watch equations being solved, or study rules that govern mathematic operations. The result is that the student creates a learning process that he or she is most comfortable with. (Disk)
List $39.95 **SALE $19.95**



Webster's Numbers — Introduce your young children to basic arithmetic concepts with this program that combines lively animation with four action games. For children with minimal reading skills, they will learn shape discrimination, eye-hand coordination, number recognition and sequential counting to name just a few. Children will delight in maneuvering a hot air balloon, leading a mosquito to safety and pushing blocks and triangles into different shapes. (Disk)
List $39.95 **SALE $19.95**

☆ ☆ Britannica Learning Companies ☆ ☆

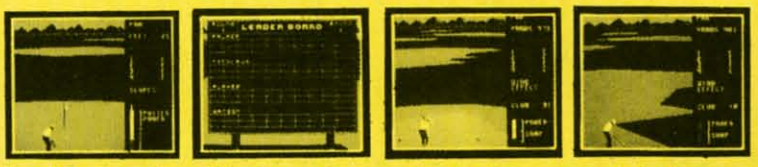
Reader Service No. 116

C64 Super Sale!! Save On C128

Access Software

Best Multi Level Graphics Games in the U.S.A.

No. 1 Golf Game



Leader Board — This is not just another golf program, it is the most realistic one on the market because it is really a simulator that allows true perspective of the game. Three levels of play take you from beginner to pro. The courses are difficult and require concentration to come in under par. (Disk) List $39.95. **Sale $23.95.** *Coupon $22.95.*

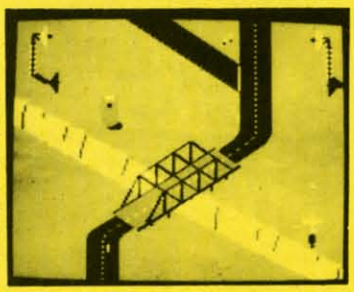
Disk No. 1 — Courses for Leader Board. Four different golf courses are on this disk of varying degrees of difficulty. List $19.95. **Sale $13.95.**

Disk Drive Speeder-Uppers

Mach 5 — Plug Mach 5 into the cartridge port and turn on the computer, the cartridge works automatically to load programs up to 5 times faster than normal. Loads 99% of all software, even copy protected games. Works with all popular printers. Will list directories without disturbing programs in memory. Commands such as LOAD, SAVE, OPEN etc. are executed with one keystroke. More features than most other "fast loaders". Does not disturb RAM. (Cart.) List $34.95. **Sale $19.95.**

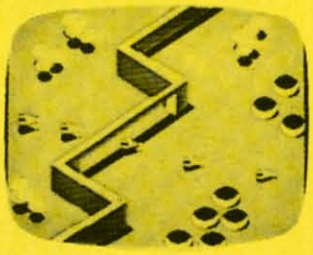
Mach 128 — This "fast loader" will load programs up to 500% faster on the 1541 disk drive or the 1571 drive in 1541 mode, plus it has features that make it one of the best on the market. A full help menu, screen dump, 95% compatibility, 10 second formatting, plus much, much more. (Disk) List $49.95. **Sale $28.95.**

Beachhead II — Fantastic hi-res scrolling screens will lead you to new levels of excitement. Three scenarios, each with fantastic scrolling screens, add great realism to the game. First you maneuver your helicopter past tanks and artillery to the fortress wall. Then you must time your moves quickly as you go through the opening and closing gates. Scene 2 starts after you drop your paratroopers and make your way to the dictators machine gun emplacement. You must attack the dictator and rescue his hostages. Finally in Scene 3 you have the final confrontation plus you can be the rescuer or the dictator and play against an opponent or the computer. Fantastic two player Head to Head action. Highly recommended. (Disk) List $49.95. **Sale $22.95**



Beach Head — Multi Screen computer graphics at its finest. First you must maneuver your fleet through the mine laden channel while torpedoes are fired at you from all sides. Next you must engage the enemy planes with your anti-aircraft guns (best scenario graphics anywhere). Then you must sink the enemy fleet before they sink you. Next you maneuver to the beach head where your tanks must get through an obstacle course and finally fight the ultimate battle at Headquarters. All scenes in stunning 3-D graphics. List $39.95. **Sale $17.95.** (Disk or Tape.)

Raid over Moscow — The ultimate in multi screen computer graphics has arrived. (over six screens) The Soviets have launched an attack. First you must locate and destroy the ICBM missiles from your Stealth Bomber, then you must lead your squadron through the Soviet terrain to the Kremlin itself. Destroy the defenses, then destroy the defense center and finally the attack operations themselves. Fly planes, launch grenades, shoot bazookas. Super Fantastic Graphics, Sound, Animation. List $39.95. **Sale $22.95.** (Disk.)



Add $3.00 for shipping, handling and insurance. Illinois residents please add 6½% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES, EXCEPT CANADA. Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail! Prices & availability subject to change without notice.
VISA — MASTER CARD — C.O.D. C.O.D. on Phone orders only

PROTECTO

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5244 to order

Reader Service No. 116

EPYX
COMPUTER SOFTWARE

Commodore 64

ORDER NOW

Fast Load Cartridge — Load, save and copy disks five times faster than normal. It plugs into the cartridge port ot the Commodore 64 and goes to work automatically loading disks with ease. And that's only the beginning. It can copy a single file, copy the whole disk, send disk commands, and even list directories without erasing programs stored in memory. (Cartridge). List $39.95. **Sale $22.95.**

EPYX
COMPUTER SOFTWARE

Commodore 64

ORDER NOW

The Best Arcade Games Around

World Karate Championship — Become a Karate master! You must prove your ability as you earn your belts. Travel to four different cities to fight a series of opponents. There are 14 controllable moves, including kicks, blocks and front flips. The opponents become progressively tougher as skills improve. Start as a "white belt" in this exciting martial arts challenge. The character on-screen is realistic right down to facial expressions. Excellent graphics and background music. (Disk) List $34.95. **Sale $18.95.** *Coupon $17.95.*

Super Cycle — Experience the excitement and danger of motorcycle racing. Lean into the turns and cruise through the scenic Grand Canyon or the Rocky Mountains, as you race cross country. Measure your speed with your tachometer and speedometer. All the instruments are realistic. Features eight different courses, day and night racing and obstacles to challenge the driver, like road barricades, oil slicks and ice. The competition is stimulating and the graphics are first rate. (Disk) List $39.95. **Sale $23.95.**

The World's Greatest Football Game — This one has it all — strategy, action, three views of the field. It's in a class all its own! Finally, a football game that not only puts you on the field, but also on the sidelines in the coach's shoes. Use the "Playbook" or design your own offensive and defensive plays. Then, grab the joystick and put your strategy to the test. You control key players to run a sweep, make a tackle, throw a pass and even kick a fieldgoal. All the action and all the strategy make this your favorite football game. (Disk) List $39.95. **Sale $22.95.**

Winter Games — Experience the challenge of six winter sporting events. In the Bobsled, you're right in the tube careening along the walls. At the SkiJump, you control your form in take-off, flight and landing. In Figure Skating, timing counts for the transitions, the jumps and landings. Choreograph your own routine in Free Style Skating. Hot Dog Aerials push your agility to new heights. The Biathlon challenges your endurance in cross-country skiing. There's even an opening ceremony, complete with national anthems. One to Eight Players.
(Disk) List $39.95. **Sale $23.95.**

Jet Combat Flight Simulator — Flight simulation programs are the most requested in the country. The key to a good one is realism, the sensation of being in the cockpit — guiding the plane through take-offs, landings and air to air — air to ground combat. You are an Air Force pilot and your mission is critical. Your success in completing your orders depends on how quickly and accurately you react. Very intense — Fantastic Graphics, animation and control elements. (Disk) List $39.95. **Sale $18.95.**

Summer Games II — The original Summer Games was last Summer's No. 1 seller. Carry on the tradition with another chance to "Go for the Gold!" Introducing Summer Games II with 8 new Olympic events including Kayaking, cycling, fencing, diving, track & field, gymnastics and equastrian. The excitement of Olympic competion is present in this new version as it was in the original. Great graphics and sound effects. This one's a winner! (1 to 8 players). (Disk)
List $39.95. **Sale $20.95.**

Movie Monster — Remember all those Saturday afternoons in front of the TV, with a big bowl of popcorn and an old Japanese monster movie? If you do then Epyx has created a program that will soon become a favorite in your software library. You choose the monster and control the outcome of the movie. The movies of the past always ended with the destruction of the monster but you can turn the tables by placing your creation in the hero role. Develop exciting plots for hours of fun. (Disk) List $39.95. **Sale $23.95.**

The New World's Greatest Baseball Game — One of the season's hottest offerings is this new enhanced baseball program. Fans can take over managing their favorite team, or create an All-Star team with players from the past or present. Two double sided disks are filled with teams, players and their stats. The graphics are improved and the team roster is the most complete that is currently available. A scoreboard will challenge your knowledge of baseball trivia. Two game modes allow you to play and manage or just manage. Fantastic. (Disk) List $34.95. **Sale $18.95.**

Super Graphics

Add $3.00 for shipping, handling and insurance. Illinois residents please add 6½% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES, EXCEPT CANADA. Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail! Prices & availability subject to change without notice.
VISA — MASTER CARD — C.O.D. C.O.D. on Phone orders only

PROTECTO

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5244 to order

Reader Service No. 116

Commodore 64 SALE

Weekly Reader

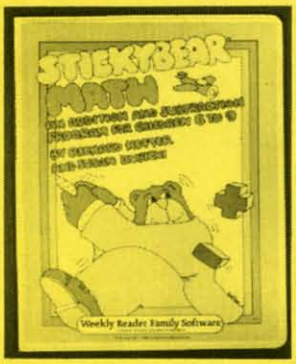
Commodore 64 SALE

☆ ☆ Educational Programs ☆ ☆

Stickybear Typing — Stickybear makes learning the keyboard both easy & fun for anyone. A 3 game format provides different levels so that as typing skills improve so does the level of difficulty. ***Stickybear Keypress*** is geared to the needs of the child or the beginner, the stress is on the keyboard. To increase typing speed and accuracy the fast-paced ***Stickybear Thump*** will provide a real challenge. ***Stickybear Stories*** allows further practice with real text for timed practice. Everything from jokes, stories and bright amusing graphics make this one of the most delightful typing programs ever, plus you can keep track of your progress and store on disk. (Disk) List $29.95. **Sale $14.95.**

C64 SALE

C64 SALE



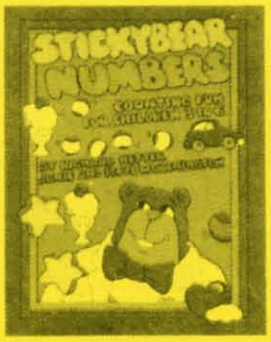
Stickybear Spellgrabber — 3 games in one: Picture Spell contains over 200 word/picture combinations. Word Spell is a scramble game, players recreate words shown on the screen. Bear Dunk involves guessing a word to save Stickybear from a dunking. (Disk) List $29.95. **Sale $14.95.**

Stickybear Math — Kids learn math skills as they help the Stickybear Family out of sticky jams. For every set of addition and subtraction problems youngsters get right, Stickybear gets a step closer to getting out of the jam. (Disk) List $29.95. **Sale $14.95.**

Stickybear Town Builder
Kids develop essential map skills as they build a town with roads, parks, airports, bridges, etc. and save on disk. That's just the first game. In Take A Drive, Compass-reading and directions are stressed. With Find The Keys, follow clues to locate mystery keys. (Disk)
List $29.95. **Sale $14.95.**

Stickybear Reading
3 fun activities that build reading comprehension skills. Match The Words: Kids practice vocabulary as they match up word and picture sets. Find The Word: Turns the words in a sentence into animated action. Build A Sentence: Children choose a subject, a verb and an object, then watch them turn into a picture. (Disk)
List $29.95. **Sale $14.95.**

Stickybear ABC — Beautifully animated pictures help children learn their ABC's and become familiar with computers. Two full-screen, full-color pictures with sound represent each letter in the alphabet. Besides helping boys and girls become familiar with computers, Stickybear ABC teaches younger children to recognize and name the letters. Older children begin to recognize words on sight. (Disk)
List $34.95. **Sale $14.95.**



Stickybear Numbers — There are colorful groups of big moving objects, such as ducks, birds and bears, that reinforce numbers and counting skills. You can use the space bar to learn to count up and down. The many possible picture combinations work to captivate youngsters with an endless variety of picture shows and give them plenty of hands on computer experience. (32 page book, disk, colorful Stickybear poster, Stickybear stickers.) (Disk) List $34.95. **Sale $14.95.**

Add $3.00 for shipping, handling and insurance. Illinois residents please add 6½% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES, EXCEPT CANADA. Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail! Prices & availability subject to change without notice.
VISA — MASTER CARD — C.O.D. C.O.D. on Phone orders only

PROTECTO

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5244 to order

Reader Service No. 116

HI TECH ♥ EXPRESSIONS™

Celebrate Every Occasion With Your Computer!

You've no idea just how much fun your computer can be! With Hi Tech Expressions creative software, your computer will come alive with musical animation, paper printouts, and computer greetings.

With CardWare and HeartWare you can personalize the animation to make your own greeting disks. Or express special wishes on full-page printouts and folded cards.

Throw a high-tech party with PartyWare. Create festive invitations, banners, hats, games . . . all kinds of party stuff. You can even make floppy disk invitations!

And the data base will make sure that no occasion or friend is forgotten.

Add an artistic flair to all of your Hi Tech Expressions with WareWithAll. It's the state-of-the-art supply kit with everything you need to jazz up your computer creations.

Your computer never had such fun!

CardWare™
Animated Birthday Greeting Disk And All-Occasion Card Maker
$7.95

PartyWare™
Card And Party Design Kit With Database Of Friends And Events
$12.95

Available for Commodore 64/128, Atari 400, 800, 800 XL, 1200 XL, 130 XE, Apple IIe, c,+ with 64K, and IBM PC, XT, AT and PC jr. with 256K and a color graphics adapter.

*Software not included in WareWithAll.

Special For The Holidays

JingleDisk
Holiday Card-Maker and Greeting Disk
$4.95

New and improved for 1986! Makes more cards. This card maker prints all kinds of holiday cards! Great selection of graphics!

What's more you'll see an animated musical story set to traditional Yuletide carols for a unique computer greeting!

Holiday PrinterPaper
Festive Designs to Brighten All Your Season's Greetings
$7.95

Includes 150 sheets of PrinterPaper in three festive holiday designs--making your holiday cards and print-outs look great! Also 24 holiday stickers for more holiday fun!

Our Holiday Gift To You--FREE HOLIDAY GRAPHICS DISK!
When you redeem the enclosed coupon you'll receive a special selection of *Holiday Graphics* to use with PartyWare. PartyWare is a print utility for making Greeting Cards, Banners, Ornaments and all kinds of party stuff to celebrate the holidays through the New Year!

Reader Service No. 116

HeartWare™
Animated Friendship Greeting Disk And Love Note Maker
$7.95

WareWithAll™
Colorfully Designed Paper, Envelopes, Stickers, Markers, Disk Labels, Disk*
$12.95

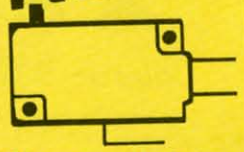
Hi Tech Expressions, Inc.
2699 South Bayshore Drive
Suite 1000A, Coconut Grove, Florida 33133
305/854-2318 1-800-848-9273
Telex 6712408 IMICO

JOYSTICK SALE

Arcade Quality Joysticks

for Atari® /Commodore®

NEW



Microswitches

5 Year Warranty

Rated No. 1

competition PRO™ 5000

List $24.95

Sale $16.95

for Atari® 5200

competition PRO™ 5000

List $29.95

Sale $24.95

Allows Keypad Hook Up

Flight Control Joystick for Atari® /Commodore®

competition PRO™ 3000

List $19.95

Sale $12.95

Trigger, Top and Base Fire Buttons

Economy Joystick for Atari® /Commodore®

competition PRO™ 1000

List $12.95

Sale $4.95

Single Fire Button

Spring Switches

All these quality competition PRO™ Joysticks feature:

- 2 or 5 year unconditional warranty
- Right or left hand operation
- Quality construction - Made in U.S.A.

PROTECTO

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5244 to order

Add $3.00 for shipping, handling, and insurance. Illinois residents please add 6½% sales tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. All orders must be in U.S. Dollars. WE DO NOT EXPORT TO OTHER COUNTRIES EXCEPT CANADA. Enclose Cashier Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail. Prices & Availability subject to change without notice.

VISA — MASTER CARD — C.O.D. C.O.D. on phone orders only.

Reader Service No. 116

Commodore 64

MODEM

Commodore 64

Sale

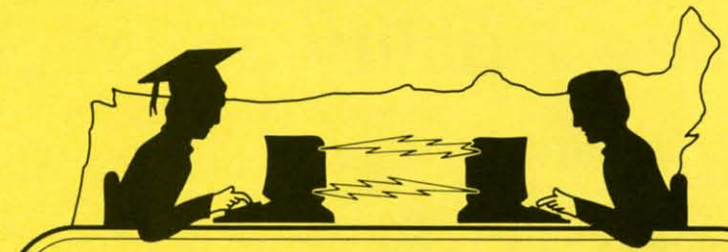
SAVE

List $99.00

Sale

$29.95

Coupon $24.95



Telecommunications

Lowest Price In the U.S.A.

List $99.00

Sale

$29.95

Coupon $24.95

FOR CHILDREN·ADULTS·BUSINESS

Complete Auto Dial Telecommunications Package

"The only telecommunications package you will ever need."

(Exclusive Easy To Use Features)

• 300 Baud Modem • Auto Dial • Auto Answer • Upload & Download

Reach Out and Access Someone

- Educational courses
- Financial Information
- Banking at Home
- Popular Games
- News Updates and Information
- Electronic Shopping
- Research and Reference Materials

The Complete Telecommunications Package offers you all this plus ...

- Auto Log-on
- Dialing from Keyboard
- On-line Clock
- Capture and Display High Resolution Characters
- Download Text, Program or Data Files
- Stores on Disk Downloaded Files
- Reads Files from Disk and Uploads Text or Program Files
- Select Any Protocol (access almost any computer or modem)
- Plus Much, Much More

List $99.00 **Sale** **$29.95** *Coupon $24.95*

We are so sure this is the only telecommunications package you will need we will give you 15 days Free Trial.

Add $3.00 for shipping, handling and insurance. Illinois residents please add 6½% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES, EXCEPT CANADA. Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail! Prices & availability subject to change without notice.
VISA — MASTER CARD — C.O.D. C.O.D. on Phone orders only

PROTECTO

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5244 to order

Reader Service No. 116

Complete Apple® Compatible
Word Processing System
Home • Business • Word Processing

Look At All You Get For Only $499.00

	List Price
• Laser 128 Apple Compatible Computer	$499.00
• Big Blue 8½" Letter Size 80 Column Printer	199.00
• 9" Samsung Green Screen Monitor	129.95
• Magic Window IIe Word Processor	150.00
• Magicalc Spreadsheet	150.00
• Magic Memory Data Base	60.00
• Laser 128 Printer Interface	19.95
• 2 Rolls of Paper	19.95
Comparable Apple System ($2495.00)	1227.85

Save Over $700.00

All 8 Pieces only $499.00

Magic Window IIe Word processing is easy and more efficient because of a simple-to-read menu system and editing commands. Many powerful features of professional quality programs and more. Supports virtually all major printer functions. Operates with most 80-column video cards or 40/70 column no-hardware modes.

Magicalc Rated as the best electronic spreadsheet on the market for Apple®. The speed with which Magicalc solves number problems allows you to accurately analyze decisions beforehand. Supports multiple RAM cards of most manufacturers in any combinations up to a full 512 K.

Magic Memory File anything, any way you want to: names, addresses, important numbers, dates and notes. Your information is organized simply, easy to maintain, and available instantly. Supports all popular printers and video cards.

Apple® is a registered trademark of Apple Computer Inc.

Add $30.00 for shipping, handling, and insurance. Illinois residents please add 6 1/2% sales tax. Add $60.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. All orders must be in U.S. Dollars. WE DO NOT EXPORT TO OTHER COUNTRIES EXCEPT CANADA. Enclose Cashier Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders. 1 day express mail. Prices & Availability subject to change without notice.

VISA — MASTER CARD — C.O.D. C.O.D. on phone orders only.

Computer Direct
We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5050 to order

Reader Service No. 116

COMMODARES
PROGRAMMING CHALLENGES

By Dale Rupert

Each month, we'll present several challenges designed to stimulate your synapses and toggle the bits in your cerebral random access memory. We invite you to send your solutions to:

Commodares, c/o *Ahoy!*
P.O. Box 723
Bethel, CT 06801

We will print and discuss the cleverest, simplest, shortest, most interesting and/or most unusual solutions. Be sure to identify the *name* and *number* of the problems you are solving. Also show sample runs if possible, where appropriate. Be sure to tell what makes your solutions unique or interesting, if they are.

Programs on diskette (1541 format only) are welcome, but they must be accompanied by listings. You must enclose a stamped, self-addressed envelope if you want any of your materials returned. Solutions received by the middle of the month shown on the magazine cover are most likely to be discussed, but you may send solutions and comments any time. Your original programming problems, suggestions, and ideas are equally welcome. The best ones will become *Commodares*!

PROBLEM #34-1: DISTANCE FINDER

This problem was inspired by a suggestion from Yuk Hung Mak (Curacao, Netherland Antilles). The user specifies the width (W) and height (H) of a grid (such as 320 and 200 for the C-64 graphics screen). Assume the points in the grid are numbered sequentially from 1 to (W times H). The user specifies two points A and B which are each between 1 and (W times H). The computer calculates the straight-line distance between those two points.

An example will clarify the problem. If W=320 and H=200, point number 1 is in the upper left corner of the grid. The upper right corner is point number 320, since the grid is 320 units wide. Point number 321 is at the left edge of the second row, and number 640 is at the right edge of that row. The bottom left and right corners are numbered 63681 and 64000 respectively. The distance from point 1 to point 63681 is 200 (top left corner to bottom left corner). The distance from point 1 to point 320 is 320 (top left corner to top right corner). The distance from point 1 to point 64000 is 377.36 with help from our friend Pythagoras.

PROBLEM #34-2: GREASED IGPAY

We are honored with this challenge from Dr. Petrie Curryfavor, CyberCryptographer and Professor of Obfuscation, as submitted by his colleague and alter ego Eddie Johnson (Albuquerque, NM). The user enters a word, and the program translates the word into Pig Latin. The good doctor is absolutely certain that no one can beat his 95-byte, 3-line Pig Latin translator program. Urelsay omesay ofway ouyay ancay oday etterbay anthay atthay, ichtnay ahrway?

PROBLEM #34-3: DIGIT DECOMPOSITION

The user enters any positive integer up to nine digits long. Write a program to print out the individual digits and their sum. If the user enters 1234, the computer displays "1 + 2 + 3 + 4 = 10." One slight catch: no strings (except the "+" and "=") and no string functions are allowed. Still an easy problem, isn't it?

PROBLEM #34-4: SIX COUPLES

Here's a good one from around the world, submitted by Necah Buyukdura (Ankara, Turkey). Using the first letters of the first names of twelve people (six couples), we are told:

C, F, P, Q, and T are YOUNGER;
A, D, E, F, and Q are TALLER;
D, P, Q, S, and T are DARKER;
and A, C, P, S, and U are SLIMMER than their spouses.

(B and R are not mentioned above.)

Write a program that will deduce and then print six pairs of letters, each pair giving the first name initials of each couple.

We received several good explanations for the "PRINT CHR$(34)" puzzle/bug mentioned at the end of the June *Commodares*. The problem was to explain why

```
PRINT MID$(CHR$(34),1);2;3
```

prints reverse-video right brackets after the two and the three. The explanation is that printing CHR$(34) puts the computer into the "Quote/insert" mode. In this mode, cursor movements show up as special symbols. In particular the cursor-right mode is a reverse-video right bracket. When the computer prints the number "2," it prints a space where the optional minus sign might be in front of the digit, and it also prints a cursor-right character after the digit so that there is a gap between a number and whatever follows it.

Try this example from Art Shipman (Westbrookville, NY). Move up to the line just above the READY on your

ABBY'S DISCOUNT SOFTWARE

AMIGA HARDWARE

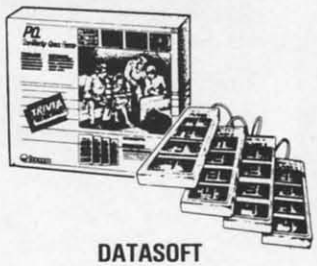
256 K. COLOR AMIGA SYSTEM
256 K. AMIGA CPU
A1010 3½" DISK DRIVE
A1080 COLOR MONITOR
A1050 256 K. EXPANSION

CALL FOR PRICES!!

INDIVIDUAL ITEMS OR PACKAGES

EARLY BIRD CHRISTMAS SPECIALS

SUNCOM
P.Q. PARTY QUIZ
ONLY
$14.95
COMMODORE (8-BIT)



TOP HITS

AMIGA SOFTWARE

Aegis Draw	$159
Aegis Animator/Images	$109
Analyze	$ 79
Amiga Assembler	$ 79
Amiga Pascal	$ 79
Arctic Fox	$ 30
Ballyhoo	$ 29
Brataccas	$ 39
Bulletin Board System	$ 79
Calcraft	$ 79
Deja Vu	$ 39
Deluxe Paint	$ 69
Art & Utility Disk #1	$ 24
Deluxe Print	$ 69
Art Disk #2	$ 24
Deluxe Video	$ 69
Driving Game	$ 39
Financial Cookbook	$ 35
French & Spanish Grammar	ea. $ 29
Halley Project	$ 35
Graph Craft	$ 49
Instant Music	$ 32
Lattice "C"	$169
Little Computer People	$ 39
Marauder (Copy)	$ 34
Marble Madness	CALL
Mind Shadow	$ 28
Mind Walker	$ 39
Music Studio	$ 39
One On One	$ 29
On Line	$ 49
Practical Composition	CALL
Practical Vocabulary	$ 44
Racter	$ 34
Robot War	$ 29
Rogue	$ 29
Scribble	$ 65
Seven Cities of Gold	$ 29
Skyfox	$ 29
Temple of Apshai Trilogy	$ 29
Text Craft	$ 49
Typing Tutor/ Word Invaders	$ 27
Winter Games	$ 29
World Geography Adventure	ea. $ 49

C-64/128 SOFTWARE

All ABACUS Titles	CALL
All MASTERTRONIC Titles	CALL
Bard's Tale	$28
Battle Group	$35
Bop 'N Wrestle	$21
Computer Quarterback	$24
Conflict in Vietnam	$27
Copy II 64/128	$39
Crusade in Europe	$27
CSM 1541 Disk Alignment	$39
Elite	$25
Fast Load (R)	$26
Fast Hack'em	$29
F-15 Strike Eagle	$25
Flight Simulator II	$39
Scenery Disk	ea. $16
Fontmaster II	$32
GEOS	$49
Gamemaker	$25
Gunship	$25
Hardball	$25
Hitchhiker's Guide to the Galaxy	$25
Jet	$29
Little Computer People	$23
Karateka	$22
Kampfgruppe	$35
Merlin 64/128	$39
Moviemaker	$24
Music Studio	$23
Newsroom	$39
Clip Art I & II	ea. $23
Leaderboard Golf	$29
Tournament Disk	$14
Phantasie I & II	ea. $24
Print Shop	$35
Graphic Library I, II & III	ea. $18
Print Master	$27
Art Gallery I & II	ea. $24
Print Shop Companion	$28
Snapshot	$39
Silent Service	$25
Rings of Zilfin	$28
Winter Games	$26
World Karate Championship	$29

COMMODORE HARDWARE

D 64 C Computer	$179
D 1541 C Disk Drive	$199
C-128 Computer	CALL
VIC-1571 Disk Drive	CALL
VIC-1581 Disk Drive	CALL
512K Ram Expansion	$179

PRINTERS

OTHERS AVAILABLE – PLEASE CALL

Star - NX-10	$279
Epson - FX-85	$399
Panasonic - 1091	$229

RIBBONS

Commodore 1525	$7.50
Commodore 1526/802	$7.00
Commodore MPS-801	$7.50
Commodore MPS-803	$9.00
Epson MX/RX/FX-80	$5.00
Okimate 10 - Color	$6.00
Star Powertype	$6.00
Star Gemini 10 - Black	$2.50
Star Gemini 10 - Color	$3.50

BULK DISKS

100 Quality DSDD	ea. $ .69
100 Quality SSDD	ea. $ .59
Less Than 100 – DSDD	ea. $ .79
Less Than 100 – SSDD	ea. $ .69
Gold Star DSDD 3½"	ea. $1.79
Sony DSDD 3½"	ea. $2.05

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

DATASOFT
COMMODORE 64/128

O'Riley Mine (D/C)	$5.95
Dallas Quest (D)	$5.95
Mancopter (D)	$5.95

TIMEWORKS
COMMODORE 64/128

Money Manager (D/C)	$5.95
Electronic Checkbook (D/C)	$5.95
Data Manager (D/C)	$5.95

MINDSCAPE
C-64/128 - EDUCATIONAL

Age 4 to 8

Tink's Adventure	$6.95
Tuk Goes to Town	$6.95
Tinka's Mazes	$6.95
Tonk in the Land of The Buddy Bots	$6.95

Age 8 to 12

Mr. Pixel's Paint Set	$9.95
Mr. Pixel's Cartoon Kit	$9.95
Show Director	$9.95
Bank Street Storybook	$9.95

TEDDY RUXPIN	**$65**
GRUBBY	**$49**
The Adventure Series	ea. $12
FOB Plush Hand Puppet	ea. $12
Teddy Ruxpin Clothing Starting at	$12

DUST COVERS

Amiga System Pack	$29
Amiga 1010 Disk Drive	$ 8
C-128 Computer	$ 8
1541 Disk Drive	$ 7
1571 Disk Drive	$ 8
1702 Monitor	$10
VIC 20/64 Computer	$ 7

ALL TITLES ON DISK UNLESS MARKED (R) FOR ROM CARTRIDGE

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

WE CHECK FOR STOLEN VISA & MASTERCARD
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 #AH-106

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

Reader Service No. 113

screen. Type in direct mode PRINT 1"X" and press <RETURN>. You now see " 1AXY." where "READY." was. The "RE" is replaced by the leading space and the digit. The "A" is skipped over by the cursor-right, and the "X" is printed.

Art sent the following program to provide further insight into how the Quote mode works.

```
·1 REM QUOTE MODE DEMO BY ART SHIPMAN
·2 PK=244 : IF LEN(DS$)=0 THEN PK=212
·3 Q$=CHR$(34)
·4 PRINT Q$;PEEK(PK);:GOTO 4
·5 PRINT Q$;:POKE PK,0:PRINT PEEK(PK);:GO
 TO 5
·6 PRINT Q$;:POKE PK,1:PRINT PEEK(PK);:GO
 TO 6
```

Line 1 sets PK to the address of the Quote mode flag. For the C-128 the address is 244. If DS$ has a length of zero, then the program is being run on the C-64, and PK is changed to the appropriate value of 212. Q$ is a quotation mark. Line 4 is executed repeatedly until you press the RUN STOP key. You will see the quotation mark followed by the current value of the Quote mode flag (1) and the right bracket discussed above. The next time line 4 is executed, another quotation mark is printed followed by the number zero (without the right bracket), indicating that the computer is no longer in the Quote mode.

Press RUN STOP and type GOTO 5. This line repeatedly takes the computer out of Quote mode by POKEing 0 into address PK. No more right brackets are displayed. Press RUN STOP again and type GOTO 6. Now the computer is consistently put into Quote mode, and the " 1]" is continuously displayed. Art even suggested a reason for the use of the right bracket. In the *1541 User's Manual* under the PRINT# section, it mentions that CHR$(29) is used as a delimiter between numbers in sequential files. A trailing space would be indistinguishable from a leading space, so a different character (the right bracket) is used. Change the semicolons after Q$ to commas in lines 4, 5, and 6 and you will see a string of right brackets corresponding to the number of cursor-rights inserted by the computer to move the cursor to the next tab column. Thanks to Art as well as to Harold Garter (Phoenix, AZ), Rick Rothstein (Trenton, NJ), Charles Grady (Cleveland, TN), and Jim Speers (Niles, MI) for all this information. After running the program above, predict or explain the output from PRINT Q$,C if you can. Hint: The 0 corresponds to the value of C.

Back to reclaim his title for the "World's Longest Listable Line" is James Borden (Carlisle, PA). Jim's program for the C-64 includes a machine language patch to the LIST code to allow lines longer than 256 characters. His program copies BASIC from ROM into RAM in order to change the LIST code. Then it fills memory with a one-line program consisting of a line number, three zero bytes at the end, and 38,904 "RESTORE" tokens (hex BC). A SYS call causes this Superline to be listed. By Jim's calculations, it is 272,337 characters in length and requires six minutes to list. If you are interested in Jim's machine language routine and BASIC program, send me a self-addressed, stamped envelope, with your request clearly stated, to *Commodares*.

Now on to the *Commodares* from the June issue. All of the solutions to *Problem #30-1: Functional Entry* used the dynamic keyboard technique. The problem suggested by Glen Bohusch (Erie, PA) was to allow the user to run a program, type an expression for a user-defined function, and have the program evaluate that function for various values. The solution from S. Simonetti (Glen Rose, TX) is listed below.

```
·1 REM COMMODARES PROBLEM #30-1:
·2 REM   FUNCTIONAL ENTRY
·3 REM SOLUTION BY
·4 REM      S. SIMONETTI
·5 REM
·8 K1=842 : CT=208 :REM  KBD BUFFER
·9 IF LEN(DS$)=0 THEN K1=631 : CT=198
·10 PRINT"DEFINE THE FUNCTION FNA(X) = "
·20 INPUT A$
·30 PRINT"[CLEAR][3"[DOWN]"]50 DEF FNA(X)
 ="A$ : PRINT"GOTO50[HOME]"
·40 POKE K1,13:POKE K1+1,13:POKE CT,2:END
·100 FOR X=1 TO 10 : PRINT FNA(X) : NEXT
```

This program runs on the C-64 and the C-128. The keyboard buffer and queue counter are in different locations in the two machines. Line 8 assumes a C-128 is being used. If the program is run on a C-64, line 9 changes the start of the keyboard buffer variable and the queue counter to their proper values. The variable DS$ is a special disk status variable for the C-128 only. On the C-64 it is normally a null string.

Line 30 positions the cursor and prints the function definition on the screen with the GOTO 50 statement below it. Line 40 stuffs the keyboard buffer with two Return codes and tells the queue counter that there are two characters waiting. When the program ends in line 40, the two Returns are processed. The first enters the new line 50, and second causes the program to GOTO 50 where the function is defined and execution continues.

Other readers went to the trouble of printing the added screen lines in the background color so that line 50 and the GOTO 50 were invisible. Jim Speers (Niles, MI) suggested adding these lines to do that for the C-64:

```
25 POKE251,PEEK(646):POKE646,PEEK(53281)
55 POKE646,PEEK(251)
```

For the C-128, replace the 646 with 241. Line 25 saves the current character color whose value is in location 646 (241 on the C-128) and replaces it with the background color whose value is in address 53281. Line 55 restores the original color. Location 251 is an available zero page RAM address. On the C-64, the original color value must be stored in RAM rather than in a variable, since variable values are lost when line 50 is added to the program.

Problem #30-2: What Environment turned out to be too much of a challenge for everyone except Jim Speers (Niles, MI). Jim's solution is printed below.

```
·1 REM COMMODARES PROBLEM #30-2:
·2 REM      WHAT ENVIRONMENT?
·3 REM SOLUTION BY
·4 REM      JIM SPEERS
·5 REM
·100 DIM C$(32):FORI=1 TO 32:READ C$(I):N
 EXT
·110 DATA BLACK,WHITE,RED,CYAN,PURPLE,GRE
 EN,BLUE,YELLOW,ORANGE,BROWN,LIGHT RED
·120 DATA DARK GRAY,MEDIUM GRAY,LIGHT GRE
 EN,LIGHT BLUE,LIGHT GRAY,BLACK,WHITE
·130 DATA DARK RED,LIGHT CYAN,LIGHT PURPL
 E,DARK GREEN,DARK BLUE,LIGHT YELLOW
·140 DATA DARK PURPLE,DARK YELLOW,LIGHT R
 ED,DARK CYAN,MEDIUM GRAY,LIGHT GREEN
·150 DATA LIGHT BLUE,LIGHT GRAY
·160 Z=FRE(1)-65536*(SGN(FRE(1)<0)):IFZ>4
 0000 THEN 190
·170 IF Z<37000 THEN 260
·180 GOTO 250
·190 PRINT"COMMODORE 128":X=RGR(0)
·200 IF X>=5 THEN PRINT"80 COLUMN MODE":P
 RINT"SCREEN COLOR= ";C$(16+RCLR(6)):GOTO
  300
·210 IF X=0 THEN PRINT"40 COLUMN MODE":PR
 INT"PRINT COLOR= "C$(RCLR(5)):GOTO 280
·220 IF X>0 AND X<5 THEN PRINT"BIT MAP MO
 DE":GOSUB 310:END
·230 PRINT"BORDER COLOR= ";C$(RCLR(4))
·240 PRINT"PRINT COLOR= ";C$(RCLR(5))
·250 PRINT"COMMODORE 64":PRINT"40-COLUMN
 MODE":GOTO 270
·260 PRINT"VIC 20":PRINT"20-COLUMN MODE"
·270 PRINT"PRINT COLOR= ";C$((PEEK(646)AN
 D15)+1)
·280 PRINT"SCREEN COLOR= ";C$((PEEK(53281
 )AND15)+1)
·290 PRINT"BORDER COLOR= ";C$((PEEK(53280
 )AND15)+1):IF X<>5 THEN END
·300 PRINT"PRINT COLOR= ";C$(RCLR(5)+16):
 END
·310 CHAR1,0,1,"COMMODORE 128":CHAR1,0,2,
 "BIT MAP MODE"
·320 X$="BORDER COLOR= "+C$(RCLR(4)):CHAR
 1,0,3,(X$)
·330 X$="SCREEN COLOR= "+C$(RCLR(0)):CHAR
 1,0,4,(X$)
·340 X$="PRINT COLOR= "+C$(RCLR(1)):CHAR1
 ,0,5,(X$):CHAR1,0,6,STR$(RGR(0))
```

The problem was to write a program which could determine the type of computer as well as the operating modes and screen colors of the computer on which it is being run. Since the program contains some of the new C-128 commands, it gives a SYNTAX ERROR when LISTed on a C-64, but it will run on a VIC 20 and a C-64 (as well as a C-128), since the offending lines are never executed on those machines. Notice how Jim determined the type of computer in lines 160 and 170. The amount of free memory is the key. If this is part of a long program, the numbers in those lines might have to be changed. Lines 310 through 340 allow this program to handle even the C-128 bit-map mode. If you are writing software that may be run on various machines, you should consider using some of the ideas of this program.

We received several good solutions to *Problem #30-3: Word Wrapper* which was originally submitted by Jim Speers. Of all the solutions, Jim's own solution is the most versatile. It is listed below.

```
·1 REM COMMODARES PROBLEM #30-3:
·2 REM      WORD WRAPPER
·3 REM SOLUTION BY
·4 REM      JIM SPEERS
·5 REM
·10 L$="[LEFT]":B$=" ":FORI=1TO79:LL$=LL$
 +L$:BL$=BL$+B$:NEXT:D$=CHR$(20)
·20 Q$=CHR$(34)+CHR$(20)+CHR$(34):WL=0:LL
 =0:C$="[c P]":PRINT"[CLEAR][DOWN]"C$;
·30 GETZ$:IFZ$="" THEN 30
·40 IF Z$="[BACKARROW]" THEN PRINT L$;B$:
 END
·50 X=ASC(Z$):IF X=32 THEN WL=0:W$="":LL=
 LL+1:PRINT L$Z$C$;:GOTO 140
·60 IF X=34 THEN PRINT L$Q$C$;:WL=WL+1:LL
 =LL+1:WL$=WL$+Q$:GOTO 140
·70 IF X=13 THEN PRINT L$B$:PRINT C$;:LL=
 0:WL=0:W$="":GOTO 30
·80 IF X<>20 THEN 120
·90 PRINT D$D$C$;:LL=LL-1:IF WL>0 THEN WL
 =WL-1:W$=LEFT$(W$,LEN(W$)-1)
·100 IF LL<0 THEN LL=LL+80
·110 GOTO 30
·120 IF (X>16ANDX<32) OR (X>127ANDX<160)
 THEN 30
·130 PRINT L$Z$C$;:W$=W$+Z$:WL=WL+1:LL=LL
 +1
·140 IF LL<81 THEN 30
·150 IF WL=0 THEN 30
·160 PRINT LEFT$(LL$,WL+1)LEFT$(BL$,WL-1)
 W$C$;:LL=WL:GOTO 30
```

The versatility comes from the fact that there are no PEEKs or POKEs, and no assumptions about screen memory locations. Consequently it is very easy to adapt Jim's program for screens of any width. To use this on the C-128 with an 80-column screen, just make these changes:

Line 10: change 39 to 79 (one less than screen width)
Line 100: change 40 to 80 (equal to screen width)
Line 140: change 41 to 81 (one more than screen width)

Jim uses the <BACK ARROW> as the escape key for

this program. Press it when you are done typing.

Other very good solutions to this problem were from Paul Sobolik (Pittsburgh, PA), Steven Steckler, and Ben Margolin (Detroit, MI). Congratulations to Ben and the rest of the computer team at Renaissance High School who are three-time city champs, and who made it to the international championships in Detroit this year.

It was easy to separate the correct solutions from the incorrect ones to *Problem #30-4: Color Bars*. The readers who felt that it was very likely for the screen to eventually be filled with one color generally misinterpreted the problem. The length of each character bar to be printed was determined by the greatest number of consecutive repetitions of any one color previously chosen, where colors were randomly chosen from 16 possibilities. Until two consecutive matching colors are randomly chosen, color bars are printed singly. After the first occurrence of two consecutive matching colors, color bars are each printed two characters in width. This continues until three consecutive matching colors are chosen, after which color bars are printed which are each three characters wide. And so forth.

The solution from Bing Perry (Monterey, CA) is listed below.

```
·1 REM COMMODARES PROBLEM #30-4:
·2 REM      COLOR BARS
·3 REM SOLUTION BY
·4 REM      BING PERRY
·5 REM
·10 CLMEM=241 : IF LEN(DS$)=0 THEN CLMEM=
 646
·20 PRINT CHR$(147);:HLD=0:REP=1:CTR=1
·30 RNR=INT(RND(0)*16)
·40 POKE CLMEM,RNR:FOR I=1 TO REP:PRINT C
 HR$(65+RNR);:NEXT
·50 IF RNR=HLD THEN CTR=CTR+1:GOTO 70
·60 CTR=1
·70 IF CTR>REP THEN REP=CTR
·80 HLD=RNR : GOTO 30
```

It is representative of most of the correct solutions. Bing's program displays colored letters, for easier interpretation. Most readers used "<RVS ON><SPACE><RVS OFF>" as the color bar character. In the program above, HLD is the last color selected; REP is the greatest number of repetitions so far; CTR holds the current number of repetitions of the selected color; and RNR is the current number from the random number generator.

Line 10 allows the program to be run on the C-128 and the C-64. The character color memory location is 241 on the C-128 and 646 on the C-64. C-128 users may want to add PRINT "<CONTROL-G>" at the end of line 70 so that the computer beeps whenever the color bar length changes.

True to form, Jim Speers did an interesting analysis of the likelihood of the screen ever being one color. The color bar must reach a length of 32 for the screen to be a solid color, since 32*32 is greater than the 1000 characters needed to fill the screen. That means that the same random number must occur 32 times in a row, and that's not very likely at all. Based upon some preliminary timing and the results of a simulation program he wrote, Jim concluded that it would take on the order of 10^{28} years (that's one with 28 zeros after it) on the average for the screen to be one color. Jim points out that this assumes the random number generator is truly random. In actuality it generates a very large but finite group of numbers. If in that group there is no occurrence of 32 consecutive numbers which are the same, the solid color screen will never happen.

Alain Goyette (Gander, Newfoundland) wrote an assembly language program to solve this problem. He and Scott Nicodemus used the random noise generator of the SID chip to generate random numbers as described in *Tips Ahoy!* in the June issue. In BASIC,

```
POKE 54287,255 : POKE 54290,128 : R=PE
EEK(54299)
```

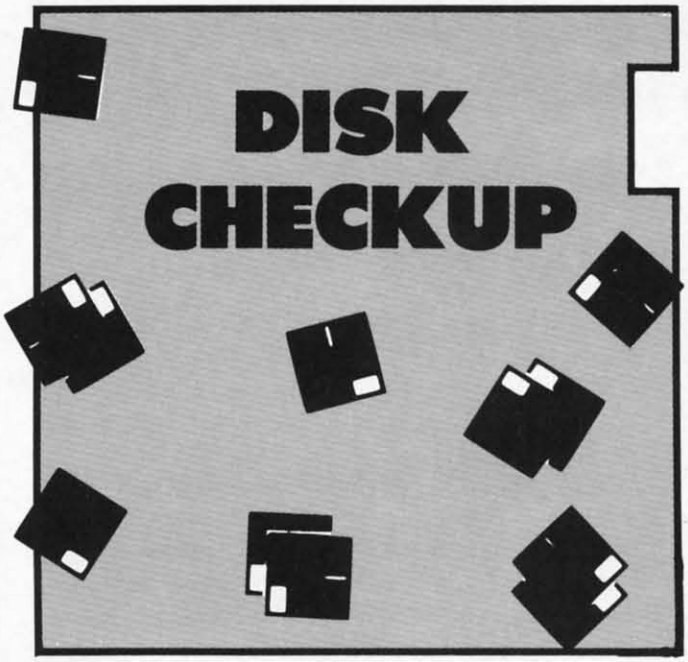
selects a random number R from 0 to 255. Scott used R=(PEEK(54299) AND 15) to choose his random numbers from 0 to 15. Alain ran his BASIC program for three days and never got a color bar of length greater than five. His assembly language program did better with a length of 13.

Jim Speers concluded that a VIC 20 using only two possible colors on its 22 by 23 screen should produce a solid-colored screen in only 29¼ hours. With the eight colors of the VIC, waiting for a color bar 23 characters long should take a mere 46,000,000 years.

Congratulations also to these readers not already mentioned this month: Dean Holmes (St. Petersburg, FL), Will Ludwigsen (Port Charlotte, FL), Karen Middaugh (San Diego, CA), Mark Breault (Brandon, MAN), Daniel Brumbaugh (Chambersburg, PA), Michael Jacknis (Dix Hills, NY), Craig Ewert (Crystal Lake, IL), Pete Baker (Rio Oso, CA), Tony Mannucci (Berkeley, CA), and Jimmy Wong (San Francisco, CA).

One final challenge. Paul Sobolik (Pittsburgh, PA) used the dynamic keyboard to solve *Problem #30-1: Functional Entry*, just as everyone else did. Paul said he worked on trying to devise a way to POKE the definition of the function into the program text, but he found the task of tokenizing too complex. He asked if there is a way to use BASIC's CRUNCH ROUTINE (at $A579 in the C-64) to do this. Or how about writing the string defining the function to a disk file and then merging it with the rest of the program? Any thoughts and examples you can come up with will be appreciated.

Don't wait around for your screens to fill with color bars. Get busy on this month's challenges. If these are too easy for you to bother with, send us some harder ones. If these are too hard for you to figure out, send us some easier ones that you have solved. See you next month. □



For the 1541 and the C-64, Plus/4, C-16, and VIC 20 (+8K)

By Donald Fulton

What with the bugs that seem to inhabit the Commodore DOS, it pays to periodically check the health of your files. It's not too well known that there is enough redundancy in the Commodore DOS to allow files to be checked very thoroughly. For example, the number of Blocks Free on the disk can be figured three ways: from the directory, from the BAM, and from the actual files. *Disk Checkup* displays all three. In a healthy disk all three should agree. An exception is some commercial disks that may use a nonstandard BAM or directory.

Disk Checkup, however, does much more than display three Blocks Free. It thoroughly checks each file in the directory. In testing this program on three of my heavily used working disks, which contained about 150 files, I found three previously undetected problem files. *Disk Checkup* first displays an expanded directory, including the starting track and sector of each file, and then checks that all files start at different track and sectors. This is a quick check that will detect the most common DOS file error: two directory entries pointing to the same file.

Disk Checkup then proceeds to trace each file. If the traced block count does not agree with the directory block count, the file is displayed in inverse video. The directory block count was the number of blocks in the file when it was saved, so if the block counts disagree, either the file is a different file than shown in the directory, or part of the file has been lost.

Disk Checkup will detect any case of file overlap, including partial overlap, because it checks each block for overlap during the trace. If a problem is found, the overlapping files are identified. Unlike Validate, *Disk Checkup* will not abort if it encounters an illegal file link, i.e. a link to a nonexistent track or sector. If it encounters an illegal link during a trace, it prints an illegal link message, and continues with the next file.

It is not necessary to monitor *Disk Checkup* while it works, since tracing all the files on a full disk can take a few minutes. On the final screen *Disk Checkup* displays a full status report on the disk files and three Blocks Free. To bring a problem disk back to health, a general guideline is to first Validate (Collect) the disk, delete any problem files, and Validate again.

Disk Checkup is safe to run on all disks because it does not write to the disk. If a disk contains REL files, they may be incorrectly identified as problem files, because REL files use a different directory format from PRG, SEQ, or USR files. *Disk Checkup* is written in 2.0 BASIC in a structured format, yet with an eye to speed. The program includes many REM statements that may be omitted if desired. □ *SEE PROGRAM LISTING ON PAGE 128*

S.O.S.

By Tim Little

My friends and I write BASIC programs, and we have heard of tricks and techniques that will allow our programs to run faster. Using these techniques, is it possible to write arcade-speed games without machine language? Also, could you tell us what kind of techniques we could use to speed up our BASIC programs?

Due to the way the BASIC language is translated by the C-64 and C-128, commercial quality animated action games are very difficult to implement successfully. BASIC is interpreted statement by statement, each statement taking possibly dozens of machine language steps to execute, not including the time or effort the BASIC interpreter spends just parsing (decoding) your program line into separate byte-sized pieces. The most efficient method of programming such a game would be to use machine language (actually assembly language would be the correct term, but the two are more or less interchangeable) which is not as difficult as you may think. So to answer your first question, arcade speed games are extremely difficult if not impossible without at least some machine language routines.

As for your second question, I will just list the techniques, as an explanation would be far too lengthy to print on this page:

1) Use variables instead of numbers whenever possible. For example:

```
AD = PI * RD
```

will execute much faster than

```
AD = 3.14 * 15
```

2) Put all commonly used subroutines in the beginning of your program. This is probably the opposite of what you were taught, and will require extensive rewriting of your program, but if you are trying to write fast code this is a must.

3) Replace the number 0 (zero) with a period (.).

4) Integers are supposed to run faster than floating point numbers, but timing program execution proves this to be false. So use floating point, even though the latter may use more memory.

5) While using FOR/NEXT loops, don't name the variable of the loop after the NEXT—it was intended to be optional on most home micros. An example should clear up any problems:

```
FOR X = 1 TO 1500 : NEXT
```

will run faster than

```
FOR X = 1 TO 1500 : NEXT X
```

That last X slows it down.

6) Lastly, remove all REMs and extra spaces, and put as much text on one line as possible.

All of the above is technically known as bad programming technique, but it works.

Seeing so many programs in all the Commodore magazines, I wondered what the formula for converting the SYS, PEEK, and POKE addresses from the C-64 to the C-128 would be.

Unfortunately, the simple answer is that no such formula exists. Although from a logical standpoint, it might seem that one would just multiply every address by two and let the program go at that, this just won't work.

The C-64 has ROM from $A000 to $BFFF and from $E000 up. Saved within the ROM is the operating system, the program that controls every function of the C-64 from checking the keyboard for recent input to interpreting your BASIC programs. The ROM is a machine language program and even has variables that can be easily changed by a programmer. Since the C-64 and the C-128 don't have identical operating systems, and because of the myriad of differences between the two machines, a formula is impossible.

What you can do is consult a memory map, a book that describes the function of most of the addresses in a specific machine, for both machines and look for similar labels (a label is a six letter name given to an address or routine) or functions for the given address.

ERRATUM

ESCAPE FROM SKULL CASTLE (July '86)

Five @ symbols were missing from the end of line 618 as printed. The error will cause an incorrect *Bug Repellent* code to be printed for that line, but will cause no other problems. To generate the correct *Bug Repellent* code, enter the line as follows:

```
618 PRINTTAB(21)"[RVSON][BLACK]NOP[RVSOF
F] [RVSON]QR[BLUE][3"@"][BLACK]@[BLUE][6
"@"]
```

Also, several readers reported incorrect *Bug Repellent* codes for all the data statement lines (1000-1178). This will occur if a user types in any line within the program with an incorrect number of quotation marks.

VAULT OF TERROR

Continued from page 61

Wand: battles magic with magic. Good for those invisible walls you keep slamming into, or hitting spirits.

Key: unlocks doors. You must be facing a door to unlock it. If the door opens, you will hear a hiss, or a beep if it is still locked.

Gold: heavy stuff. Need it to win the game, but you may have to drop it to get everything up that ladder.

Crown: you have to be clever indeed to find the hidden location of this treasure.

Not long after you enter the dungeon, you're sure to meet one of its intelligent, nasty inhabitants—either by running into one of them, or one of them coming looking for you. Your human scent is a dinner bell for the creatures in the dungeon. They are as follows:

Brown Recluse: this spider is a skinny little varmint who is usually carrying a torch, a remnant of the adventurer he had for breakfast. A stroke of the sword or a few raps with the torch will usually kill him, but don't underestimate him. He might get a lucky shot.

Red Tarantula: a deadlier cousin of the Recluse. Carries a sword. A wee bit faster on the draw.

Blue Crypt Rat: carries a shield in his junkpack. Possibly lethal teeth and claws.

Brown Carrion Rat: carries the ring, but doesn't know how to use it; he pulled it off a skeleton. Bigger, badder.

Purple Goblins: the original inhabitants of the dungeon before the Naga arrived. There are only two left alive; the rest perished at the hands of the serpents. Each carries a rope on his backpack and is armed with a scythe and a short sword. Don't fight these guys until you are ready. They wander around on the first level, looking for overzealous newcomers to carve up.

Light Blue Revenant: ghastly undead spirit, a corpse brought to life by the Emperor Naga's sorcery. He has the power of the "Cold touch"—he changes inorganic and organic material into the energy he needs to retain his existence on this plane. May not be fooled by the ring! These horrid friends will take your possessions if they succeed in touching you, and hoard them in a secret and usually well-guarded area of the dungeon.

Green Naga: giant cobra. Fast and powerful. A single hit from this snake can kill you if you are weak.

Emperor Naga (silver): this guy is the worst. He strikes as fast as you do and does more damage. You will have to be either lucky or smart to defeat him. (Hit and run is best.) He will be carrying a sack of gold, but he has hidden his favorite treasure, the crown, somewhere in the dungeon maze.

All of these monsters can be defeated in mortal combat, even the Revenant, but you will find that the wand is a better weapon against the Revenant than the sword.

You can fight and move in the dark, but you will find it much to your advantage to use the torch or the ring for visibility rather than stumble around in blackness. Torches burn for quite some time, but after a while you will see the outline of the dungeon growing dimmer, turning from light grey to medium grey to dark grey to blue and finally black. To replenish the torch you are carrying, drop it, walk away from it until you are in darkness, and turn around, go back, and pick it up again. When you use it the torch will burn brightly again. If the torch should go out on you, your situation is pretty hopeless unless you have placed other torches in strategic locations or you find one accidentally. (The ring could help you get out of a tight situation like this.)

When you do decide to engage a monster in combat, you and he must occupy the same square. You can move forward past a monster, and he can sometimes do the same to you. When you hear the creature's cry, he has entered the same block as you. You can now engage him in battle with your bare hands, torch, or the sword if you have it. The sword is a superb weapon, doing four times the damage of a torch or your bare hands. If you are empty-handed, you will still be able to do slight damage to monsters by pressing the joystick button, even enough to kill one if you are fortunate to hit him enough times!

Merely possessing the shield gives you its protection, so you will never need to use it, only have it, for it to protect you. A flash of red will let you know when you have been wounded. The screen flashes pink when you are lightly struck, orange when you are hit soundly, and dark red when you are gravely wounded. You will know when you strike your opponent by the flash of green and the sound of your slashing blade.

If you are hale and well, the border is black. When you are severely wounded, the border turns yellow. When you are within a single stroke of death, the border turns red. However, the greatest advantage you have over the monsters is that your wounds can heal, while theirs are cumulative until they are killed. Good strategy can entail hitting a creature several times, running away and resting for a while (until the border turns to yellow or black again), and coming back to deliver the death blow.

You do not have to hit the joystick button repeatedly to keep slashing with the sword, simply hold it down to keep raining blows—but a friend of mine claims you can time the strokes better if you keep popping the button.

You grow stronger by spending time in the dungeon and by killing other creatures. Don't take on a creature outside your capabilities, or descend to the maelstrom of the second level, until you are strong enough!

The first level is rather easy—you may be killed several times before you get the hang of it, but eventually you should be able to kill anything with a few blows on the first level. When you are strong enough and you have the rope, you can go down to the serpent's lair, the second level. The second level is the stronghold of the Emperor Naga and his evil servants.

Stand on a block that has an open shaft leading down, HOLD the rope (should be lit in yellow), and press the button. You will hear your descent, and the next screen that appears will be a view with a ceiling exit above you somewhere on the second level. To go back up, you stand on a block that has both a ladder and a ceiling exit, holding the rope, and press the button. You will hear your ascent, and you will be back on the first level. You cannot go up when there is a ceiling exit only, with the exception of the first one you saw when the game started, which is the one you leave through to win the game!

You will notice that the game has true three dimensional graphics, with the creatures closest to you obscuring the ones behind them—and if you look you will even see creatures behind ladders.

In your search for the crown, you will see rooms, coffins, ladders, shafts, and ceiling exits. The coffins cannot be opened. Ladders that stand vacant in corridors are exits to the surface that were sealed off by the Naga. You cannot climb up them without an exit above them.

The rooms are the most puzzling of obstacles. All rooms have four doors. Although a door can be locked from the outside, keeping you out, you can always leave from the inside of a room in any direction if there is a corridor outside that way. The doors are self-latching, however, and will remain locked behind you until you unlock them with either the key or the wand. Then you can pass freely through the door both ways. Because the view of a door from the inside of a room looks identical no matter which direction you face, the only way you can tell you have turned inside a room is the flash of the command bar and a change in your direction indicators. If you are inside a room and you hear a beep when you try to go through a door, indicating it is impassable, it is for one of two reasons: there is a stone wall on the other side, or the door is rusted shut and cannot be opened by (hint) earthly means. When you are outside and hear a beep, the door is merely locked. (Use the key.)

The most confusing obstacle in the dungeon is the invisible barrier, conjured up by the Emperor Naga to protect something of value to him. Some of these are one-way obstacles—you can walk through them but not back. The only thing that can smash these barriers is the wand, carried by the Revenants. It is not possible to win the game without passing through at least one of the barriers. When you are walking around and you suddenly hear a bump, you have walked into an invisible barrier.

There are multiple partitions on each floor, and you may have to go down, then come up again, or even go down again to get to them. It can get very confusing, and it is helpful to map the dungeon extensively in your travels. A friend can do this while you battle the monsters, or you can do it yourself.

Avoid the spirits at all cost! The Revenants can steal all your possessions with a single touch, even when you have the ring on. Sooner or later, however, you will have to kill one to get a wand. Do it wisely, by slashing and then moving back. If you stand toe to toe and slug it out he is sure to get your equipment before he expires.

When you get the crown and have brought all the objects on the command bar to the first floor, go to the original space you started in, under the ceiling exit you dropped down into the vault through, and USE the rope to win the game! Only when you have everything (all objects on the command bar lit in light red) can you exit and win the game.

There is a great deal more I could hint about, but I think half the fun of the game is solving the riddles in it and discovering the secret location of the crown. I leave the heavy exploration up to you. The game is seriously addictive, which I noted when I let several of my friends play it, and most of them yearned to break into the source listing and deduce the secrets of the *Vault*. For this reason I disabled the RUN STOP and RESTORE keys, as well as LIST, while the game is running.

I challenge you to win the game fairly. I think you will find it every bit as simultaneously frustrating and exhilarating as *Zork* itself! The save feature of the game allows you to set aside a game and come back to it, so you can play it for weeks before winning it, and I think even then you will return to the *Vault* for regular sojourns.

To use the load feature, hit f2 at the prompt on the title screen and enter the name of the file you wish to load. The filename must be 11 letters or less. If you hit RETURN with no input, the load will default to "LASTDUNGEON". The last game will be loaded, if you saved a game the last time you played. To save a game, go to the command bar at any time during the course of a game and hit f8. The current game will be saved to disk for you under the name "LASTDUNGEON". The reason the load feature asks you for the name of the file is that I plan to create new dungeon files if this game meets with reader approval. In that event, you will be able to enter the name of the new file to load in, instead of "LASTDUNGEON", and play a whole new version of the game with a new floorplan and new puzzles to solve! □ ***SEE PROGRAM LISTING ON PAGE 120***

PROGRAM LISTINGS

Attention new *Ahoy!* readers! You must read the following information very carefully prior to typing in programs listed in *Ahoy!* Certain Commodore characters, commands, and strings of characters and commands will appear in a special format. Follow the instructions and listing guide on this page.

On the following pages you'll find several programs that you can enter on your Commodore computer. But before doing so, read this entire page carefully.

To insure clear reproductions, *Ahoy!*'s program listings are generated on a daisy wheel printer, incapable of printing the commands and graphic characters used in Commodore programs. These are therefore represented by various codes enclosed in brackets [ ]. For example: the SHIFT CLR/HOME command is represented onscreen by a heart ♥. The code we use in our listings is [CLEAR]. The chart below lists all such codes which you'll encounter in our listings, except for one other special case.

The other special case is the COMMODORE and SHIFT characters. On the front of most keys are two symbols. The symbol on the left is obtained by pressing that key while holding down the COMMODORE key; the symbol on the right, by pressing that key while holding down the SHIFT key. COMMODORE and SHIFT characters are represented in our listings by a lower-case "s" or "c" followed by the symbol of the key you must hit. COMMODORE J, for example, is represented by [c J], and SHIFT J by [s J].

Additionally, any character that occurs more than two times in a row will be displayed by a coded listing. For example, [3 "[LEFT]"] would be 3 CuRSoR left commands in a row, [5 "[s EP]"] would be 5 SHIFTed English Pounds, and so on. Multiple blank spaces will be noted in similar fashion: e.g., 22 spaces as [22 " "].

Sometimes you'll find a program line that's too long for the computer to accept (C-64 lines are a maximum of 80 characters, or 2 screen lines long; VIC 20 lines, a maximum of 88 characters, or 4 screen lines). To enter these lines, refer to the *BASIC Command Abbreviations Appendix* in your User Manual.

On the next page you'll find our *Bug Repellent* programs for the C-128 and C-64. The version appropriate for your machine will help you proofread our programs after you type them. (Please note: the *Bug Repellent* line codes that follow each program line, in the whited-out area, should *not* be typed in. See the instructions preceding each program.)

Also on the following page you will find *Flankspeed*, our ML entry program, and instructions on its use. □

Call *Ahoy!* at 212-239-0855 with any problems.

When You See	It Means	You Type		You Will See
[CLEAR]	Screen Clear	SHIFT	CLR/HOME	♥
[HOME]	Home		CLR/HOME	
[UP]	Cursor Up	SHIFT	↑ CRSR ↑	
[DOWN]	Cursor Down		↑ CRSR ↑	
[LEFT]	Cursor Left	SHIFT	←CRSR←	
[RIGHT]	Cursor Right		←CRSR←	
[SS]	Shifted Space	SHIFT	Space	
[INSERT]	Insert	SHIFT	INST/DEL	
[DEL]	Delete		INST/DEL	
[RVSON]	Reverse On	CNTRL	9	
[RVSOFF]	Reverse Off	CNTRL	0	
[UPARROW]	Up Arrow		↑	
[BACKARROW]	Back Arrow		←	
[PI]	PI		π	
[EP]	English Pound		£	

When You See	It Means	You Type		You Will See
[BLACK]	Black	CNTRL	1	
[WHITE]	White	CNTRL	2	
[RED]	Red	CNTRL	3	
[CYAN]	Cyan	CNTRL	4	
[PURPLE]	Purple	CNTRL	5	
[GREEN]	Green	CNTRL	6	
[BLUE]	Blue	CNTRL	7	
[YELLOW]	Yellow	CNTRL	8	
[F1]	Function 1		F1	
[F2]	Function 2	SHIFT	F1	
[F3]	Function 3		F3	
[F4]	Function 4	SHIFT	F3	
[F5]	Function 5		F5	
[F6]	Function 6	SHIFT	F5	
[F7]	Function 7		F7	
[F8]	Function 8	SHIFT	F7	

BUG REPELLENT By MICHAEL KLEINERT and DAVID BARRON

Bug Repellent is a checksum program used for proofreading BASIC listings typed in from *Ahoy!* magazine. For each program line you enter, *Bug Repellent* will produce a two-letter code that should match the code listed beside that line in the magazine.

Type in, save, and run the *Bug Repellent.* (If you have a C-64, type in the C-64 version. If you have a C-128, you will need to type in the C-64 version for use with C-64 programs, and the C-128 version for use with C-128 programs.) If you have typed in *Bug Repellent* properly, you will get the message BUG REPELLENT INSTALLED; otherwise you will get an error message. If you get an error message, double check the *Bug Repellent* program for typing mistakes. Type NEW and hit RETURN. Then type in and save, or load, the *Ahoy!* program you wish to check. Type in SYS 49152 for the C-64 version or SYS 3072 for the C-128 version and hit RETURN (this will begin execution of *Bug Repellent*). You will see the prompt SCREEN OR PRINTER ? Hit S if you want the codes listed on the screen, or P if you want them listed on the printer. To pause the listing depress and hold the SHIFT key.

Compare the codes your machine generates to those listed to the right of the corresponding program lines. If you spot a difference, that line contains an error. Write down the numbers of the lines where the contradictions occur. LIST each line, locate the errors, and correct them.

COMMODORE 64 VERSION

```
·100 FOR X = 49152 TO 49488:READY:S=S+Y                    AB
·110 IF Y<0 OR Y>255 THEN 130                              EA
·120 POKE X,Y:NEXT:GOTO140                                 ID
·130 PRINT"[CLEAR][DOWN]**ERROR**":PRINT"[DOWN
]PLEASE CHECK LINE"PEEK(64)*256+PEEK(63):END               ID
·140 IF S<>44677 THEN PRINT"[CLEAR][DOWN]**ERR
OR**":PRINT"[DOWN]PLEASE CHECK DATA LINES 170
-500":END                                                  HJ
·150 PRINT"[CLEAR]":POKE53280,0:POKE53281,6:PO
KE646,1                                                    NP
·160 PRINT"[RVSON][6" "]C-64 BUG REPELLENT INS
TALLED[6" "]"                                              LF
·170 DATA32,161,192,165,43,133,251,165,44,133              DL
·180 DATA252,160,0,132,254,32,37,193,234,177               DB
·190 DATA251,208,3,76,138,192,230,251,208,2                OF
·200 DATA230,252,76,43,192,76,73,78,69,32                  KN
·210 DATA35,32,0,169,35,160,192,32,30,171                  CA
·220 DATA160,0,177,251,170,230,251,208,2,230               CE
·230 DATA252,177,251,32,205,189,169,58,32,210              JE
·240 DATA255,169,0,133,253,230,254,32,37,193               CL
·250 DATA234,165,253,160,0,76,13,193,133,253               NB
·260 DATA177,251,208,237,165,253,41,240,74,74              MB
·270 DATA74,74,24,105,65,32,210,255,165,253                EP
·280 DATA 41,15,24,105,65,32,210,255,169,13                GH
·290 DATA32,220,192,230,63,208,2,230,64,230                AN
·300 DATA251,208,2,230,252,76,11,192,169,153               NG
·310 DATA160,192,32,30,171,166,63,165,64,76                BF
·320 DATA231,192,96,76,73,78,69,83,58,32                   EP
·330 DATA0,169,247,160,192,32,30,171,169,3                 PJ
·340 DATA133,254,32,228,255,201,83,240,6,201               FK
·350 DATA80,208,245,230,254,32,210,255,169,4               FL
·360 DATA166,254,160,255,32,186,255,169,0,133              CL
·370 DATA63,133,64,133,2,32,189,255,32,192                 GC
·380 DATA255,166,254,32,201,255,76,73,193,96               NN
·390 DATA32,210,255,173,141,2,41,1,208,249                 NH
·400 DATA96,32,205,189,169,13,32,210,255,32                IM
·410 DATA204,255,169,4,76,195,255,147,83,67                KC
·420 DATA82,69,69,78,32,79,82,32,80,82                     DC
·430 DATA 73,78,84,69,82,32,63,32,0,76                     ML
·440 DATA44,193,234,177,251,201,32,240,6,138               GN
·450 DATA113,251,69,254,170,138,76,88,192,0                JK
·460 DATA0,0,0,230,251,208,2,230,252,96                    NA
·470 DATA170,177,251,201,34,208,6,165,2,73                 DM
·480 DATA255,133,2,165,2,208,218,177,251,201               JA
·490 DATA32,208,212,198,254,76,29,193,0,169                FM
·500 DATA13,76,210,255,0,0,0                               PA
```

COMMODORE 128 VERSION

```
·100 FAST:FOR X = 3072 TO 3520:READ Y:POKE X,Y
:S=S+Y:TRAP110:NEXT:SLOW                                   IH
·110 SLOW:IF S<>49057 THEN PRINT"[CLEAR][DOWN]
**ERROR**":PRINT"[DOWN]PLEASE CHECK DATA LINE
S 140-390":END                                             JA
·120 PRINT"[CLEAR][DOWN]  C-128 BUG REPELLENT
INSTALLED"                                                 II
·130 PRINT"[4" "]TYPE SYS 3072 TO ACTIVATE"                IN
·140 DATA 32,161,12,165,45,133,251,165,46,133,
252,160,0,132,254,32,37                                    OF
·150 DATA 13,234,177,251,208,3,76,138,12,230,2
51,208,2,230,252,76,43                                     NC
·160 DATA 12,76,73,78,69,32,35,32,0,169,35,160
,12,32,80,13,160,0,177                                     OL
·170 DATA 251,170,230,251,208,2,230,252,177,25
1,32,89,13,169,58,32,98                                    EF
·180 DATA 13,169,0,133,253,230,254,32,37,13,23
4,165,253,160,0,76,13                                      JO
·190 DATA 13,133,253,177,251,208,237,165,253,4
1,240,74,74,74,74,24                                       LC
·200 DATA 105,65,32,98,13,165,253,41,15,24,105
,65,32,98,13,169,13,32                                     DE
·210 DATA 220,12,230,65,208,2,230,66,230,251,2
08,2,230,252,76,11,12                                      GM
·220 DATA 169,153,160,12,32,80,13,166,65,165,6
6,76,231,12,96,76,73,78                                    CP
·230 DATA 69,83,58,32,0,169,247,160,12,32,80,1
3,169,3,133,254,32,107                                     HC
·240 DATA 13,201,83,240,6,201,80,208,245,230,2
54,32,98,13,169,4,166                                      GK
·250 DATA 254,160,255,32,116,13,169,0,133,65,1
33,66,133,250,32,125,13                                    LB
·260 DATA 32,134,13,166,254,32,143,13,76,73,13
,96,32,98,13,165,211                                       JF
·270 DATA 234,41,1,208,249,96,32,89,13,169,13,
32,98,13,32,152,13,169,4                                   GD
·280 DATA 76,161,13,147,83,67,82,69,69,78,32,7
9,82,32,80,82,73,78,84,69                                  PL
·290 DATA 82,32,63,32,0,76,44,13,234,177,251,2
01,32,240,6,138,113,251,69                                 OK
·300 DATA 254,170,138,76,88,12,0,0,0,0,230,251
,208,2,230,252,96,170,177                                  FJ
·310 DATA 251,201,34,208,6,165,250,73,255,133,
250,165,250,208,218,177                                    GA
·320 DATA 251,201,32,208,212,198,254,76,29,13,
0,169,13,76,98,13,0,0,32                                   FI
·330 DATA 170,13,32,226,85,76,180,13,32,170,13
,32,50,142,76,180,13,32                                    OF
·340 DATA 170,13,32,210,255,76,180,13,32,170,1
3,32,228,255,76,180,13,32                                  AK
·350 DATA 170,13,32,186,255,76,180,13,32,170,1
3,32,189,255                                               BP
·360 DATA 76,180,13,32,170,13,32,192,255,76,18
0,13,32,170,13                                             FP
·370 DATA 32,201,255,76,180,13,32,170,13,32,20
4,255,76,180,13,32,170                                     ID
·380 DATA 13,32,195,255,76,180,13,133,67,169,0
,141,0,255,165,67,96                                       BJ
·390 DATA 133,67,169,0,141,1,255,165,67,96,0,0
,0                                                         IF
```

FLANKSPEED FOR THE C-64 By GORDON F. WHEAT

Flankspeed will allow you to enter machine language *Ahoy!* programs without any mistakes. Once you have typed the program in, save it for future use. While entering an ML program with *Flankspeed* there is no need to enter spaces or hit the carriage return. This is all done automatically. If you make an error in a line a bell will ring and you will be asked to enter it again. To LOAD in a program Saved with *Flankspeed* use LOAD "name",1,1 for tape, or LOAD "name",8,1 for disk. The function keys may be used after the starting and ending addresses have been entered.

f1 – SAVEs what you have entered so far.

f3 – LOADs in a program worked on previously.

f5 – To continue on a line you stopped on after LOADing in the previous saved work.

f7 – Scans through the program to locate a particular line, or to find out where you stopped the last time you entered the program. f7 temporarily freezes the output as well.

```
LL ·100 POKE53280,12:POKE53281,11
ED ·105 PRINT"[CLEAR][c 8][RVSON][15" "]FLANKSPEED[15" "]";
·110 PRINT"[RVSON][5" "]MISTAKEPROOF ML ENTRY PROGRAM[6" "
   ]"
·115 PRINT"[RVSON][9" "]CREATED BY G. F. WHEAT[9" "]"
DM ·120 PRINT"[RVSON][3" "]COPR. 1984, ION INTERNATIONAL INC.
   [3" "]"
DH ·125 FORA=54272TO54296:POKEA,0:NEXT
IM ·130 POKE54272,4:POKE54273,48:POKE54277,0:POKE54278,249:PO
   KE54296,15
NH ·135 FORA=680TO699:READB:POKEA,B:NEXT
KO ·140 DATA169,251,166,253,164,254,32,216,255,96
HJ ·145 DATA169,0,166,251,164,252,32,213,255,96
JB ·150 B$="STARTING ADDRESS IN HEX":GOSUB430:AD=B:SR=B
KA ·155 GOSUB480:IFB=0THEN150
GN ·160 POKE251,T(4)+T(3)*16:POKE252,T(2)+T(1)*16
KE ·165 B$="ENDING ADDRESS IN HEX":GOSUB430:EN=B
LO ·170 GOSUB470:IFB=0THEN150
EE ·175 POKE254,T(2)+T(1)*16:B=T(4)+1+T(3)*16
MN ·180 IFB>255THENB=B-255:POKE254,PEEK(254)+1
GE ·185 POKE253,B:PRINT
HN ·190 REM  GET HEX LINE
IL ·195 GOSUB495:PRINT": [c P][LEFT]";:FORA=0TO8
NH ·200 FORB=0TO1:GOTO250
MP ·205 NEXTB
ME ·210 A%(A)=T(1)+T(0)*16:IFAD+A-1=ENTHEN340
LE ·215 PRINT" [c P][LEFT]";
IK ·220 NEXTA:T=AD-(INT(AD/256)*256):PRINT" "
PD ·225 FORA=0TO7:T=T+A%(A):IFT>255THENT=T-255
LK ·230 NEXT
IA ·235 IFA%(8)<>TTHENGOSUB375:GOTO195
LE ·240 FORA=0TO7:POKEAD+A,A%(A):NEXT:AD=AD+8:GOTO195
BI ·245 REM    GET HEX INPUT
AB ·250 GETA$:IFA$=""THEN250
HK ·255 IFA$=CHR$(20)THEN305
HF ·260 IFA$=CHR$(133)THEN535
KH ·265 IFA$=CHR$(134)THEN560
JM ·270 IFA$=CHR$(135)THENPRINT" ":GOTO620
EG ·275 IFA$=CHR$(136)THENPRINT" ":GOTO635
AB ·280 IFA$>"@"ANDA$<"G"THENT(B)=ASC(A$)-55:GOTO295
DL ·285 IFA$>"/"ANDA$<":"THENT(B)=ASC(A$)-48:GOTO295
MD ·290 GOSUB415:GOTO250
JJ ·295 PRINTA$"[c P][LEFT]";
OA ·300 GOTO205
CF ·305 IFA>0THEN320
PG ·310 A=-1:IFB=1THEN330
OI ·315 GOTO220
BM ·320 IFB=0THENPRINTCHR$(20);CHR$(20);:A=A-1
HG ·325 A=A-1
BE ·330 PRINTCHR$(20);:GOTO220
LK ·335 REM LAST LINE
AD ·340 PRINT" ":T=AD-(INT(AD/256)*256)
GJ ·345 FORB=0TOA-1:T=T+A%(B):IFT>255THENT=T-255
PL ·350 NEXT
IA ·355 IFA%(A)<>TTHENGOSUB375:GOTO195
NF ·360 FORB=0TOA-1:POKEAD+B,A%(B):NEXT
HN ·365 PRINT:PRINT"YOU ARE FINISHED!":GOTO535
JA ·370 REM BELL AND ERROR MESSAGES
FL ·375 PRINT:PRINT"LINE ENTERED INCORRECTLY":PRINT:GOTO415
DA ·380 PRINT:PRINT"INPUT A 4 DIGIT HEX VALUE!":GOTO415
FF ·385 PRINT:PRINT"ENDING IS LESS THAN STARTING!":B=0:GOTO41
   5                                                          HD
   ·390 PRINT:PRINT"ADDRESS NOT WITHIN SPECIFIED RANGE!":B=0:
   GOTO415                                                    OK
MC ·395 PRINT:PRINT"NOT ZERO PAGE OR ROM!":B=0:GOTO415        FN
DM ·400 PRINT"?ERROR IN SAVE":GOTO415                         PP
   ·405 PRINT"?ERROR IN LOAD":GOTO415                         PO
DH ·410 PRINT:PRINT:PRINT"END OF ML AREA":PRINT               PG
IM ·415 POKE54276,17:POKE54276,16:RETURN                      BH
   ·420 OPEN15,8,15:INPUT#15,A,A$:CLOSE15:PRINTA$:RETURN      IM
NH ·425 REM GET FOUR DIGIT HEX                                PC
KO ·430 PRINT:PRINTB$;:INPUTT$                                GM
HJ ·435 IFLEN(T$)<>4THENGOSUB380:GOTO430                      NP
JB ·440 FORA=1TO4:A$=MID$(T$,A,1):GOSUB450:IFT(A)=16THENGOSUB
   380:GOTO430                                                FJ
GN ·445 NEXT:B=(T(1)*4096)+(T(2)*256)+(T(3)*16)+T(4):RETURN   GF
KE ·450 IFA$>"@"ANDA$<"G"THENT(A)=ASC(A$)-55:RETURN           EH
LO ·455 IFA$>"/"ANDA$<":"THENT(A)=ASC(A$)-48:RETURN           KP
EE ·460 T(A)=16:RETURN                                        NP
MN ·465 REM ADRESS CHECK                                      LI
GE ·470 IFAD>ENTHEN385                                        LB
HN ·475 IFB<SRORB>ENTHEN390                                   KC
IL ·480 IFB<256OR(B>40960ANDB<49152)ORB>53247THEN395          MG
NH ·485 RETURN                                                IM
MP ·490 REM ADDRESS TO HEX                                    EB
ME ·495 AC=AD:A=4096:GOSUB520                                 FD
LE ·500 A=256:GOSUB520                                        PE
IK ·505 A=16:GOSUB520                                         MI
PD ·510 A=1:GOSUB520                                          IL
LK ·515 RETURN                                                IM
IA ·520 T=INT(AC/A):IFT>9THENA$=CHR$(T+55):GOTO530            PE
LE ·525 A$=CHR$(T+48)                                         JP
BI ·530 PRINTA$;:AC=AC-A*T:RETURN                             AC
AB ·535 A$="**SAVE**":GOSUB585                                LH
HK ·540 OPEN1,T,1,A$:SYS680:CLOSE1                            LH
HF ·545 IFST=0THENEND                                         EO
KH ·550 GOSUB400:IFT=8THENGOSUB420                            CM
JM ·555 GOTO535                                               CL
EG ·560 A$="**LOAD**":GOSUB585                                NE
AB ·565 OPEN1,T,0,A$:SYS690:CLOSE1                            MF
DL ·570 IFST=64THEN195                                        LC
MD ·575 GOSUB405:IFT=8THENGOSUB420                            AN
JJ ·580 GOTO560                                               CL
OA ·585 PRINT" ":PRINTTAB(14)A$                               FG
CF ·590 PRINT:A$="":INPUT"FILENAME";A$                        OM
PG ·595 IFA$=""THEN590                                        DD
OI ·600 PRINT:PRINT"TAPE OR DISK?":PRINT                      DF
BM ·605 GETB$:T=1:IFB$="D"THENT=8:A$="@0:"+A$:RETURN          IG
HG ·610 IFB$<>"T"THEN605                                      BO
BE ·615 RETURN                                                IM
LK ·620 B$="CONTINUE FROM ADDRESS":GOSUB430:AD=B              OH
AD ·625 GOSUB475:IFB=0THEN620                                 GH
GJ ·630 PRINT:GOTO195                                         PH
PL ·635 B$="BEGIN SCAN AT ADDRESS":GOSUB430:AD=B              FA
IA ·640 GOSUB475:IFB=0THEN635                                 IB
NF ·645 PRINT:GOTO670                                         PP
HN ·650 FORB=0TO7:AC=PEEK(AD+B):GOSUB505:IFAD+B=ENTHENAD=SR:G
   OSUB410:GOTO195                                            NK
FL ·655 PRINT" ";:NEXTB                                       EC
DA ·660 PRINT:AD=AD+8                                         GN
FF ·665 GETB$:IFB$=CHR$(136)THEN195                           LI
   ·670 GOSUB495:PRINT": ";:GOTO650                           IB
```

THE FASTEST DRAW IN THE INDUSTRY
FROM PAGE 27

Assembler required for program entry! See introductory article.

RECTANGLE.S

```
 1 *
 2 * RECTANGLE.S
 3 *
 4           ORG   $1300
 5 *
 6 TEMPA     EQU   $FA
 7 TEMPB     EQU   TEMPA+2
 8 *
 9 TABPTR    EQU   TEMPA
10 *
11 COLOR     EQU   $10
12 BMFLG     EQU   $D8
13 BMPTR     EQU   $A2D
14 COLMAP    EQU   $1C00
15 SCROLY    EQU   $D011
16 *
17 HMAX      EQU   320
18 *
19 SCRBAS    EQU   $2000
20 PTRL      EQU   $8000
21 PTRH      EQU   $8100
22 *
23 MAPLEN    EQU   1000
24 SCRLEN    EQU   8000
25 *
26 HSTART    EQU   $0C00
27 HEND      EQU   $0C02
28 VSTART    EQU   $0C04
29 VEND      EQU   $0C05
30 *
31 TABSIZ    EQU   VEND+1
32 HPSN      EQU   TABSIZ+2
33 VPSN      EQU   HPSN+2
34 CHAR      EQU   VPSN+1
35 ROW       EQU   CHAR+1
36 LINE      EQU   ROW+1
37 BYTE      EQU   LINE+1
38 BITT      EQU   BYTE+2
39 *
40 MPRL      EQU   BITT+1
41 MPRH      EQU   MPRL+1
42 MPDL      EQU   MPRH+1
43 MPDH      EQU   MPDL+1
44 PRODL     EQU   MPDH+1
45 PRODH     EQU   PRODL+1
46 *
47 FILVAL    EQU   PRODH+1
48 HPOS      EQU   FILVAL+1
49 *
50           JMP   START
51 *
52 BITPSN    HEX   80,40,20,10
53           HEX   08,04,02,01
54 * BLOCK FILL ROUTINE
55 *
56 BLKFIL    LDA   FILVAL
57           LDX   TABSIZ+1
58           BEQ   PARTPG
59           LDY   #0
60 FULLPG    STA   (TABPTR),Y
61           INY
62           BNE   FULLPG
63           INC   TABPTR+1
64           DEX
65           BNE   FULLPG
66 PARTPG    LDX   TABSIZ
67           BEQ   FINI
68           LDY   #0
69 PARTLP    STA   (TABPTR),Y
70           INY
71           DEX
72           BNE   PARTLP
73 FINI      RTS
74 *
75 * 16-BIT MULTIPLICATION
76 *
77 MULT16    LDA   #0
78           STA   PRODL
79           STA   PRODH
80           LDX   #16
81 SHIFT     ASL   PRODL
82           ROL   PRODH
83           ASL   MPRL
84           ROL   MPRH
85           BCC   NOADD
86           CLC
87           LDA   MPDL
88           ADC   PRODL
89           STA   PRODL
90           LDA   MPDH
91           ADC   PRODH
92           STA   PRODH
93 NOADD     DEX
94           BNE   SHIFT
95           RTS
96 *
97 * CREATE Y-LOOKUP TABLE
98 *
99 MAKTAB    LDY   #0
100 YLOOP    CPY   #200
101          BCC   CONT
102          JMP   EXIT
103 *
104 * DIVIDE Y BY 8
105 *
106 CONT     TYA
107          LSR   A
108          LSR   A
109          LSR   A
110          STA   ROW
111 *
112 * MULTIPLY ROW * HMAX
113 *
114          LDA   ROW
115          STA   MPRL
116          LDA   #0
117          STA   MPRH
118          LDA   #<HMAX
119          STA   MPDL
120          LDA   #>HMAX
121          STA   MPDH
122          JSR   MULT16
123          LDA   PRODL
124          STA   TEMPA
125          LDA   PRODH
126          STA   TEMPA+1
127 *
128 * ADD PRODUCT TO SCRBAS
129 *
130          CLC
131          LDA   #<SCRBAS
132          ADC   TEMPA
133          STA   PTRL,Y
134          LDA   #>SCRBAS
135          ADC   TEMPA+1
136          STA   PTRH,Y
137 *
138          INY
139          JMP   YLOOP
140 *
141 EXIT      RTS
142 *
143 * MAIN ROUTINE
144 *
145 * DEFINE BIT MAP AND
146 * ENABLE HI-RES GRAPHICS
147 *
148 START     JSR   MAKTAB
149 *
150          STA   $FF01
151          LDA   #$78
152          STA   BMPTR
153 *
154          LDA   #$20
155          STA   BMFLG
156 *
157          LDA   #0
158          STA   $FF00
159          LDA   SCROLY
160          ORA   #$20
161          STA   SCROLY
162          STA   $FF01
163 *
164 * CLEAR BIT MAP
165 *
166          LDA   #0
```

IMPORTANT! Letters on white background are **Bug Repellent** line codes. **Do not enter them!** Pages 101 and 102 explain these codes and provide other essential information on entering ***Ahoy!*** programs. Refer to these pages **before** entering any programs!

```
167             STA   FILVAL
168             LDA   #<SCRBAS
169             STA   TABPTR
170             LDA   #>SCRBAS
171             STA   TABPTR+1
172             LDA   #<SCRLEN
173             STA   TABSIZ
174             LDA   #>SCRLEN
175             STA   TABSIZ+1
176             JSR   BLKFIL
177 *
178 * SET BKG AND LINE COLORS
179 *
180             LDA   #COLOR
181             STA   FILVAL
182             LDA   #<COLMAP
183             STA   TABPTR
184             LDA   #>COLMAP
185             STA   TABPTR+1
186             LDA   #<MAPLEN
187             STA   TABSIZ
188             LDA   #>MAPLEN
189             STA   TABSIZ+1
190             JSR   BLKFIL
191 *
192 * DRAW HORIZONTAL LINES
193 *
194             LDA   VSTART
195             STA   VPSN
196             LDA   HSTART
197             STA   HPSN
198             LDA   HSTART+1
199             STA   HPSN+1
200             JSR   HDRAW
201 *
202             LDA   VEND
203             STA   VPSN
204             LDA   HSTART
205             STA   HPSN
206             LDA   HSTART+1
207             STA   HPSN+1
208             JSR   HDRAW
209 *
210 * DRAW VERTICAL LINES
211 *
212             LDA   VSTART
213             STA   VPSN
214             LDA   HSTART
215             STA   HPOS
216             LDA   HSTART+1
217             STA   HPOS+1
218             JSR   VDRAW
219 *
220             LDA   VSTART
221             STA   VPSN
222             LDA   HEND
223             STA   HPOS
224             LDA   HEND+1
225             STA   HPOS+1
226             JSR   VDRAW
227 *
228 INF         JMP   INF
229 *
230 HDRAW        JSR   PLOT
231             INC   HPSN
232             BNE   NEXT
233             INC   HPSN+1
234 NEXT        LDA   HPSN+1
235             CMP   HEND+1
236             BCC   HDRAW
237             LDA   HPSN
238             CMP   HEND
239             BCC   HDRAW
240             RTS
241 *
242 VDRAW        LDA   HPOS
243             STA   HPSN
244             LDA   HPOS+1
245             STA   HPSN+1
246             JSR   PLOT
247             INC   HPSN
248             BNE   SKIP
249             INC   HPSN+1
250 SKIP        JSR   PLOT
251             LDX   VPSN
252             INX
253             STX   VPSN
254             CPX   VEND
255             BCC   VDRAW
256             RTS
257 *
258 * CHAR=HPSN/8
259 *
260 PLOT        LDA   HPSN
261             LSR   HPSN+1
262             ROR
263             LSR   HPSN+1
264             ROR
265             LSR   HPSN+1
266             ROR
267             STA   CHAR
268 *
269 * MULTIPLY 8 * CHAR
270 *
271             LDA   #0
272             ASL   CHAR
273             ROL
274             ASL   CHAR
275             ROL
276             ASL   CHAR
277             ROL
278             STA   TEMPB+1
279 *
280 * ADD LINE
281 *
282             CLC
283             LDA   VPSN
284             AND   #7
285             ADC   CHAR
286             STA   TEMPB
287             LDA   TEMPB+1
288             ADC   #0
289             STA   TEMPB+1
290 *
291 * BYTE = TEMPA + TEMPB
292 *
293             CLC
294             LDY   VPSN
295             LDA   PTRL,Y
296             ADC   TEMPB
297             STA   TEMPB
298             LDA   PTRH,Y
299             ADC   TEMPB+1
300             STA   TEMPB+1
301 *
302 * BYTE=BYTE OR 2^BIT
303 *
304             LDA   HPSN
305             AND   #$07
306             TAX
307             LDY   #0
308             LDA   (TEMPB),Y
309             ORA   BITPSN,X
310             STA   (TEMPB),Y
311             RTS
312 *
```

RECTANGLE.BAS

```
BD  ·10 REM *** RECTANGLE.BAS ***
DI  ·20 :
EO  ·30 HST = 105
CH  ·40 HND = 211
CB  ·50 VST = 66
EF  ·60 VND = 132
DI  ·70 :
BO  ·80 IF A=0 THEN A=1:BLOAD "RECTANGLE.0"
KM  ·90 HI=INT(HST/256):LO=HST-HI*256
    ·100 POKE DEC("0C00"),LO:POKE DEC("0C01")
DO    ,HI
KK  ·110 HI=INT(HND/256):LO=HND-HI*256
    ·120 POKE DEC("0C02"),LO:POKE DEC("0C03")
```

```
,HI                                                    OC
·130 POKE DEC("0C04"),VST:POKE DEC("0C05"
),VND                                                  NO
·140 SYS DEC("1300")                                   NG
```

ANALOG TO DIGITAL ADVENTURES
FROM PAGE 20

DIGITAL SCOPE

```
·1 REM                                                 JD
·2 REM       -- DIGITAL SCOPE --                       OL
·3 REM      PROGRAM FOR THE C-128                      IJ
·4 REM        RUPERT REPORT #34                        OK
·5 REM                                                 JD
·6 REM  *-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*      CD
·7 REM   USE A/D CONVERTER TO CREATE                   CI
·8 REM     A DIGITAL OSCILLOSCOPE                      GN
·9 REM  KEYS:                                          BJ
·10 REM  <CRSR UP> - SHIFT DISPLAY UP                  JB
·11 REM  <CRSR DN> - SHIFT DISPLAY DOWN                OA
·12 REM  "+" - MAGNIFY DISPLAY                         HH
·13 REM  "-" - DEMAGNIFY DISPLAY                       NH
·14 REM  "C" - CLEAR DISPLAY                           LK
·15 REM  "P" - PAUSE DISPLAY                           PK
·16 REM        (ANY KEY TO RESTART)                    GN
·17 REM  "Z" - REINITIALIZE PARAMETERS                 CI
·18 REM *-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*      CD
·19 REM                                                JD
·20 TRAP 1000                                          JM
·25 REM >=>=>=>=>=>=> DEFINE CONSTANTS >=>             CJ
·30 D$="V" : U$="[UPARROW]"                            NL
·40 MAX=10      :REM  RUNNING AVERAGE QTY.             EC
·50 PN=1        :REM  POT NUMBER                       AD
·60 REM >=>=>=>=>=>=>  INITIALIZATION >=>              KB
·70 GRAPHIC 1,1                                        GP
·80 G=190/255 :REM  GAIN (MAGNIFICATION)               FM
·90 VY=5       :REM  VERTICAL OFFSET                   FJ
·100 PRINT POT(PN)*G+VY,    :REM  INITIAL
  Y SCREEN VALUE                                       LI
·110 REM >=>=>=>=>=>=>=>=>=> MAIN LOOP >=>             ID
·120 FOR X=1 TO 320                                    PC
·130 Y=200-(POT(PN)*G+VY)                              JO
·140 DRAW ,X,Y                                         PE
·150 REM >=>=>=>=>=>=> RUNNING AVERAGE >=>             FL
·160 PTR=PTR+1 : IF PTR>MAX THEN PTR=0                 OB
·170 SUM=SUM-YV(PTR)+Y                                 CE
·180 YV(PTR)=Y        :REM  REPLACE OLDEST
  Y VALUE WITH NEWEST                                  MO
·190 AVG=SUM/MAX       :REM  CALC. AVERAGE             CB
·200 DRAW ,X,AVG-50    :REM  PLOT IT                   GI
·210 REM -=-=- LOCATE OFFSCREEN IMAGE -=-              PD
·220 IF Y<200 AND Y>0 THEN CHAR ,1,24," "
  : CHAR ,1,0," " : GOTO 240                           AH
·230 IF Y>200 THEN CHAR ,1,24,D$ :ELSE IF
  Y<0 THEN CHAR ,1,0,U$                                DN
·240 GET A$ : IF A$>"" THEN 290                        JN
·250 NEXT X                                            NK
·260 GRAPHIC 1,1       :REM  CLEAR SCREEN              DE
·270 GOTO 120          :REM  START AGAIN               OA
·280 REM >=>=>=>=>=>=>=>=> KEY PRESSED >=>             MI
·290 K=INSTR("[UP][DOWN]+-CPZ",A$)                     AL
·300 IF K=0 THEN 250    :REM INVALID KEY               NI
·310 ON K GOSUB 330,340,350,360,370,380,3
  90                                                   KE
·320 ON K GOTO 250,250,250,250,120,250,70              CA
·330 VY=VY+5 : RETURN  :REM CURSOR UP                  NM
·340 VY=VY-5 : RETURN  :REM CURSOR DOWN                KD
·350 G=G*1.1 : RETURN  :REM  +(GAIN UP)                BI
·360 G=G*.9 : RETURN   :REM -(GAIN DOWN)               AF
·370 GRAPHIC 1,1 :RETURN :REM C (CLEAR)                MP
·380 GETKEY Q$ : RETURN  :REM P (PAUSE)                PG
·390 RETURN        :REM  Z (ZERO/RESTART)              HI
·1000 IF ER=30 THEN END  :REM RUN/STOP                 JD
·1010 RESUME NEXT                                      FI
```

ATTACK FORCE
FROM PAGE 72

```
·10 POKE52,55:POKE56,55:CLR:GOTO800                    OO
·20 POKE00,1:POKEPS,0:POKEHT,0:POKETH,0:F
  C=0:TC=10:LC=18:CC=12:Y=25                           KJ
·30 GOTO100                                            CF
·50 GOSUB660                                           CI
·55 PRINT"[HOME][YELLOW] AGAIN[3"."]":SYS
  SR:SYSSR                                             HM
·60 PRINT"[HOME] YOUR PEOPLE ARE FREE ONC
  E":SYSSR:SYSSR:SYSSR:SYSSR                           KD
·65 PRINT"[HOME][RED] OF THE ENEMY INSTAL
  LATIONS.":SYSSR:SYSSR                                DH
·70 PRINT"[HOME] YOU HAVE DESTROYED ALL":
  SYSSR:SYSSR                                          BF
·75 FORI=1TO6:SYSSR:NEXT:POKES+5,128:POKE
  S+6,248:POKES+12,0:POKES+13,240                      MC
·80 P(0)=34:P(1)=34:P(2)=45:D(0)=200:D(1)
  =100:D(2)=450                                        OB
·85 FORI=0TO2:POKES+1,P(I):POKES+8,P(I)-1
  5:POKES,0:POKES+7,0                                  JH
·86 POKES+4,17:POKES+11,17                             CP
·90 FORD=1TOD(I):NEXT:POKES+4,16:POKES+11
  ,16:FORJ=1TO25:NEXT:NEXT                             CK
·95 SYSXY,7,23:PRINT"[BLUE]PRESS ANY KEY"
  :POKE198,0                                           FP
·98 GETA$:IFA$=""THEN98                                GC
·99 MS=9:GOTO700                                       AD
·100 IFPEEK(PS)THEN200                                 IK
·110 IFPEEK(TH)THENGOSUB400                            BF
·115 IFPEEK(HT)THENGOSUB450                            AE
·120 IFINT(RND(1)*30)=1ANDBS=0THENGOSUB46
  0                                                    IL
·130 IFBS=1THENGOSUB470                                NA
```

```
·140 FC=FC+1:IFFC=40+MS*2THENFC=0:GOTO325 LE
·150 IFPEEK(653)AND1THEN190 GF
·180 GOSUB300 CD
·185 GOTO100 CF
·190 IFPEEK(653)THENPOKE00,0:POKES+24,0:G
OTO190 LN
·195 POKE00,1:POKES+24,47:GOTO180 ME
·200 POKE2040,253:POKES+1,8:POKES+5,0:POK
ES+6,13:POKES+4,129:POKES+4,128 FO
·205 POKES+11,0:POKES+18,0 BM
·210 POKEV+28,PEEK(V+28)OR1:FORI=1TO100:P
OKEV+39,8:POKEV+39,2:NEXT MB
·215 BS=0:SYSXY,32,21:PRINT"[5" "]" NJ
·220 FORI=1TO1500:NEXT:SH=SH-1:IFSH=-1THE
N250 NB
·225 SYSXY,31,14:PRINT"[BLUE]"SH:POKEV+21
,192:FC=0:GOTO500 EG
·250 POKE00,0:POKES+4,0:POKES+11,0:POKES+
18,0:POKEV+21,192 PK
·255 SYSXY,7,10:PRINT"[YELLOW]][11"[UPARR
OW]"]?" KE
·260 PRINTTAB(7)"[ [WHITE]GAME OVER[YELLO
W] @" PK
·265 PRINTTAB(7)"[EP][11"[UPARROW]"][BACK
ARROW]" KO
·270 FORI=1TO3000:NEXT:POKEV+21,0:GOTO700 OJ
·300 SYSSR:Y=Y+1:IFY=35-MSTHENY=0:GOSUB34
0 BA
·305 TC=TC+1:IFTC=35-MSTHENTC=0:GOSUB350 AI
·310 CC=CC+1:IFCC=35-MSTHENCC=0:GOSUB380 JJ
·315 LC=LC+1:IFLC=35-MSTHENLC=0:GOSUB390 CE
·320 RETURN IM
·325 F=F-1:IFF=0THEN200 EE
·330 SYSXY,30,17:PRINT"[9" "]" JE
·335 SYSXY,30,17:PRINTLEFT$(F$,F*2):GOTO1
80 CO
·340 SYSXY,0,0:POKE646,RND(1)*6+10:PRINTW
L$(RND(1)*9+1):RETURN BJ
·350 POKEV+6,RND(1)*220+30:POKEV+7,28:POK
EV+21,PEEK(V+21)OR8:POKETH,0 JK
·352 POKE2043,251:POKEV+42,11:RETURN MI
·380 POKEV+4,RND(1)*220+30:POKEV+5,28:POK
EV+21,PEEK(V+21)OR4:POKEHT,0 KC
·385 POKE2042,252:POKEV+41,8:RETURN HF
·390 SYSXY,0,0:PRINTLZ$(RND(1)*8+1):RETUR
N CK
·400 IFPEEK(2043)=253THENRETURN AM
·405 POKEV+42,2:POKE2043,253 AH
·410 SC=SC+250:GOSUB680 IA
·415 T=T+1:IFT=MS+4THEN570 OH
·420 RETURN IM
·450 IFPEEK(2042)=253THENRETURN PF
·455 POKEV+41,2:POKE2042,253 BB
·458 SC=SC+10+10*MS:GOSUB680:RETURN HJ
·460 BS=1:BF=150:POKES+12,0:POKES+13,240:
POKES+8,BF HD
·465 X=INT(RND(1)*26)+1:XX=INT(X/6)+1 KI
·466 SYSXY,XX+31,21:PRINT"[RED].":POKES+1
1,17 MB
·468 RETURN IM
·470 BF=BF-5:POKES+8,BF:IFBF>90THENRETURN CN
·475 Z=INT(PEEK(V+1)/8):SYSXY,X,Z-7:PRINT
"[RED][3"&"][UP][LEFT][LEFT]&[DOWN][DOWN
][LEFT]&":POKES+11,16:BS=0 MK
·480 SYSXY,32,21:PRINT"[5" "]" BJ
·485 POKES+11,16:POKES+8,8:POKES+12,0:POK
ES+13,11:POKES+11,129:POKES+11,128 FE
·490 RETURN IM
·500 POKEV+21,192:GOSUB660:POKE2040,255:P
OKE2041,254:POKE2042,252:POKE2043,251 FJ
·505 POKE V,124:POKEV+1,210:POKEV+39,6:PO
KEV+40,2:SYSXY,31,11:PRINT"[BLUE]"MS BO
·510 POKEV+41,8:POKEV+42,11:POKEV+28,12:P
OKEV+37,7:POKEV+38,12 LK
·515 POKE53270,PEEK(53270)OR16:POKE53282,
1 BC
·520 POKEV+23,0:POKEV+29,0:POKEV+21,193 DH
·522 POKES+23,244:POKES+22,0:POKES+24,47:
POKES+19,208:POKES+20,240 EO
·524 POKES+14,0:POKES+15,6:POKES+18,129 DE
·525 SYSXY,3,11:PRINT"[RED]PREPARE FOR MI
SSION"MS CE
·528 SYSXY,5,13:PRINT"DESTROY"MS+4-T"TOWE
R";:IFMS+4-T>1THENPRINT"S" GE
·530 POKES+5,0:POKES+6,240:POKES+4,33:FOR
I=1TO3 FJ
·535 POKES+1,8:POKES,97:FORJ=1TO350:NEXT:
POKES+1,12:POKES,32:FORJ=1TO350:NEXT GL
·540 NEXT:POKES+4,32 AL
·545 GOSUB660:TC=0:Y=0 GB
·550 00=53245:SS=52992:TS=52993:PS=52994:
HT=52995:TH=52996:SR=50432 AF
·555 POKES+5,0:POKES+6,9 IK
·560 F$="[RED]'[GREEN]'[GREEN]'[GREEN]'[G
REEN]'[GREEN]'[GREEN]'[GREEN]'[GREEN]'":
SYSXY,30,17:PRINTF$:F=9 CG
·565 Z=PEEK(V+30):Z=PEEK(V+31):GOTO20 FB
·570 POKEV+21,192:POKES+4,0:POKES+11,0:PO
KES+18,0:GOSUB660 PA
·575 SYSXY,4,11:PRINT"[WHITE]MISSION"MS"C
OMPLETED":POKE00,0 KB
·580 FORI=1TO500:NEXT:MS=MS+1:IFMS=10THEN
50 EC
·585 SYSXY,7,13:PRINT"[YELLOW]BONUS:"F*10
"PTS.":GOSUB680 MH
·590 FORI=1TO2500:NEXT:T=0:GOTO500 CL
·600 FORI=1093TO1973STEP40:POKEI,27:POKEI
+10,0:NEXT NN
·610 XY=832:SYSXY,29,0:PRINT"[WHITE]][9"[
UPARROW]"]?" EG
·615 SYSXY,29,24:PRINT"[EP][9"[UPARROW]"]
";:POKE2023,31 EA
·620 FORI=3TO18STEP3:SYSXY,30,I:PRINT"[9"
[UPARROW]"]":NEXT EI
·625 SYSXY,31,4:PRINT"[CYAN]SCORE":SYSXY,
31,7:PRINT"HIGH":SYSXY,31,10:PRINT"MISSI
```

```
ON"                                                    KE
·630 SYSXY,31,13:PRINT"SHIPS":SYSXY,31,16
 :PRINT"FUEL"                                          KB
·632 PRINTTAB(30)"[RED]'[GREEN][8"'"][CYA
N]"                                                    BB
·635 SYSXY,31,19:PRINT"SCANNER":PRINTTAB(
31)"[BLUE]][5"[UPARROW]"]?"                            FE
·640 PRINTTAB(31)"[[5" "]@":PRINTTAB(31)"
[  [BLUE]*[BLUE]  @"                                   BA
·650 PRINTTAB(31)"[EP][5"[UPARROW]"][BACK
ARROW]"                                                LG
·655 RETURN                                            IM
·660 FORI=0TO12:SYSXY,0,I:PRINT"[29" "]"               CA
·670 SYSXY,0,24-I:PRINT"[29" "]";:NEXT:RE
TURN                                                   FF
·680 SYSXY,31,5:PRINT"[BLUE][8" "]":SYSXY
,31,5:PRINTSC                                          BI
·685 IFSC>HSTHENHS=SC:SYSXY,31,8:PRINT"[8
" "]":SYSXY,31,8:PRINTHS                               MO
·690 RETURN                                            IM
·700 POKE53281,0:POKE53280,11:POKEV+28,0:
POKE53270,PEEK(53270)AND239                            LD
·705 POKEV+23,1:POKEV+29,1:POKEV+16,192:P
OKEV+12,25:POKEV+14,49                                 EL
·710 POKEV+13,57:POKEV+15,57:POKE2046,249
:POKE2047,250                                          BF
·712 POKEV,122:POKEV+1,85:POKEV+39,6:POKE
2040,255                                               BK
·715 POKEV+45,7:POKEV+46,7:POKEV+21,193:P
RINT"[WHITE][CLEAR]":GOSUB600                          JO
·716 SYSXY,31,5:PRINT"[BLUE]"SC:SYSXY,31,
8:PRINTHS:SYSXY,31,11:PRINTMS                          JA
·718 SH=3:SYSXY,31,14:PRINTSH                          ME
·720 SYSXY,2,15:PRINT"[RED]PRESS [WHITE]F
1[RED] TO SELECT MISSION"                              LL
·725 SYSXY,4,17:PRINT"[c 1]PRESS [WHITE]F
3[c 1] TO START GAME"                                  LP
·730 SYSXY,7,19:PRINT"[c 3]PRESS [WHITE]F
7[c 3] TO EXIT"                                        KK
·740 GETA$:IFA$=""THEN740                              JO
·750 IFA$="[F1]"THENMS=MS+1:IFMS=10THENMS
=1                                                     JA
·755 SYSXY,31,11:PRINT"[BLUE]"MS                       ED
·760 IFA$="[F3]"THEN780                                ML
·770 IFA$="[F7]"THENPOKEV+21,0:PRINT"[WHI
TE][CLEAR]":END                                        OM
·775 GOTO740                                           CH
·780 SC=0:F=9:T=0:GOSUB680:GOTO500                     EG
·800 POKE53281,11:POKE53280,11:POKE646,7:
V=53248:S=54272:PRINT"[CLEAR]":POKEV+21,
0                                                      OL
·805 FORI=STOS+23:POKEI,0:NEXT:POKES+24,1
5                                                      JL
·806 POKES+5,0:POKES+6,14:POKES+1,150                  DE
·808 POKE53270,PEEK(53270)AND239:POKEV+28
,0                                                     AN
·810 FORI=249TO255:FORJ=0TO63:READK:POKE6
4*I+J,K:NEXT:NEXT                                      BD
·815 POKE2040,249:POKE2041,250:POKEV+29,3
 :POKEV+23,3                                           NE
·820 POKEV+39,0:POKEV+40,0:POKES+4,129:PO
KES+4,128                                              IB
·825 POKEV,136:POKEV+2,184:POKEV+1,110:PO
KEV+3,110:POKEV+21,3                                   MK
·828 FORI=0TO240:POKES+1,I:POKEV+39,I:POK
EV+40,I:NEXT                                           CI
·830 PRINT"[17"[DOWN]"][YELLOW]"SPC(7)"PL
UG JOYSTICK INTO PORT 2"                               BL
·835 PRINT"[DOWN][DOWN]"SPC(13)"[c 7]LOAD
ING DATA"                                              EC
·845 POKE56334,PEEK(56334)AND254:POKE1,PE
EK(1)AND251                                            BE
·850 FORI=0TO511:POKEI+14336,PEEK(I+53248
):NEXT                                                 GC
·855 POKE1,PEEK(1)OR4:POKE56334,PEEK(5633
4)OR1                                                  IE
·860 POKE53272,(PEEK(53272)AND240)+14                  LK
·865 READK:IFK=-1THEN875                               KI
·870 FORI=0TO7:READJ:POKE14336+K*8+I,J:NE
XT:GOTO865                                             DD
·875 FORI=832TO860:READJ:POKEI,J:NEXT                  FD
·878 FORI=1TO9:READWL$(I):NEXT:FORI=1TO8:
READLZ$(I):NEXT                                        PM
·880 FORI=49152TO49577:READJ:POKEI,J:NEXT               MJ
·885 FORI=50432TO50633:READJ:POKEI,J:NEXT               IB
·890 MS=1:SH=3:POKE53245,0:SYS49553:GOTO7
00                                                     KN
·900 DATA1,223,124,1,68,17,2,68                        MC
·901 DATA17,3,196,17,4,68,17,4                         FG
·902 DATA68,17,8,68,17,8,0,0                           IC
·903 DATA16,0,0,31,156,56,32,34                        ME
·904 DATA68,60,34,68,64,66,121,64                      BA
·905 DATA68,145,128,68,144,128,56,136                  BF
·906 DATA0,0,0,0,0,0,0,0                               FG
·907 DATA0,0,0,0,0,0,0,0                               FG
·908 DATA227,145,0,20,82,0,20,20                       FD
·909 DATA0,244,24,0,20,20,0,20                         CN
·910 DATA82,0,19,145,0,0,0,0                           KI
·911 DATA0,0,0,112,240,0,137,0                         KA
·912 DATA0,129,0,0,3,192,0,2                           GH
·913 DATA0,0,148,0,0,103,192,0                         GE
·914 DATA0,0,0,0,0,0,0,0                               FG
·915 DATA0,0,0,0,0,0,0,2                               FI
·916 DATA10,160,0,42,168,0,234,171                     KM
·917 DATA0,186,175,0,171,255,0,170                     DA
·918 DATA255,0,170,255,0,170,255,0                     IJ
·919 DATA170,255,0,170,255,0,170,255                   IA
·920 DATA0,170,255,0,170,255,0,170                     DH
·921 DATA255,0,170,255,0,170,255,0                     IJ
·922 DATA170,255,0,170,255,0,42,252                    GP
·923 DATA0,10,240,0,0,0,0,0                            MJ
·924 DATA0,0,0,0,0,0,0,0                               FG
·925 DATA0,0,0,0,0,0,0,0                               FG
·926 DATA0,0,0,168,0,10,170,128                        PF
·927 DATA42,170,160,86,170,84,169,85                   CC
·928 DATA168,42,170,160,10,170,128,0                   BK
```

```
NE ·929 DATA168,0,0,0,0,0,0,0                    AE
   ·930 DATA0,0,0,0,0,0,0,0                      FG
   ·931 DATA0,0,0,0,0,0,0,18                     IC
IB ·932 DATA0,0,0,2,10,0,0,0                     IG
   ·933 DATA0,2,8,8,32,34,0,0                    OL
MK ·934 DATA138,128,10,10,0,2,168,162            HK
   ·935 DATA138,42,162,34,136,128,40,170         EO
CI ·936 DATA162,10,170,32,34,170,136,10          MJ
   ·937 DATA40,168,136,170,32,0,34,136           IF
BL ·938 DATA2,138,128,32,40,32,8,32              OO
   ·939 DATA128,0,2,8,0,32,0,255                 PN
EC ·940 DATA0,0,0,0,0,0,0,0                      FG
   ·941 DATA0,0,0,0,128,0,32,128                 LE
BE ·942 DATA0,32,128,0,32,128,0,32               OL
   ·943 DATA128,0,32,128,0,032,0,0               AA
GC ·944 DATA0,0,0,0,0,0,0,0                      FG
   ·945 DATA0,0,0,0,0,0,0,0                      FG
IE ·946 DATA0,0,0,0,0,0,0,0                      FG
LK ·947 DATA0,0,0,0,0,0,0,0                      FG
KI ·948 DATA0,64,0,0,64,0,0,64                   CI
   ·949 DATA0,0,224,0,0,224,0,1                  NF
DD ·950 DATA176,0,1,16,0,1,16,0                  DA
FD ·951 DATA1,176,0,1,240,0,1,240                IG
   ·952 DATA0,6,236,0,14,238,0,14                EF
PM ·953 DATA174,0,29,183,0,189,183,160           DM
MJ ·954 DATA253,183,224,253,183,224,7,28         PI
IB ·955 DATA0,0,0,0,0,0,0,0                      FG
   ·956 DATA0,216,216,216,216,216,216,216,21
KN 6                                             PP
MC ·957 DATA27,27,27,27,27,27,27,27,27           PF
FG ·958 DATA28,27,27,27,24,15,0,0,0              PN
IC ·959 DATA29,0,0,15,24,27,27,27,27             FE
ME ·960 DATA30,0,0,255,0,255,0,0,0               EL
BA ·961 DATA31,216,216,216,24,240,0,0,0          HJ
BF ·962 DATA33,0,1,15,15,29,60,124,126           GK
FG ·963 DATA35,0,128,240,240,184,60,62,126       EJ
FG ·965 DATA38,8,90,52,191,90,52,74,16           KI
FD ·966 DATA45,85,85,255,255,255,255,255,255     EA
CN ·967 DATA44,85,93,245,245,245,245,244,240     EM
KI ·968 DATA47,21,85,255,255,255,255,255,255     EM
KA ·969 DATA63,0,0,240,24,216,216,216,216        EK
GH ·970 DATA39,255,255,255,255,255,255,255,2
GE 55,-1                                         ED
FG ·971 DATA32,253,174,32,138,173,32,247,183     BO
FI ·972 DATA152,72,32,253,174,32,138,173,32      AD
KM ·973 DATA247,183,152,170,104,168,24,32        KF
DA ·974 DATA240,255,96                           GL
IJ ·980 DATA"--,[3" "]/[22"-"]"                  FE
IA ·981 DATA"[6"-"],[3" "]/[18"-"]"              GM
DH ·982 DATA"[10"-"],[3" "]/[14"-"]"             FE
IJ ·983 DATA"[14"-"],[3" "]/[10"-"]"             DM
GP ·984 DATA"[18"-"],[3" "]/[6"-"]"              DE
MJ ·985 DATA"[22"-"],[3" "]/--"                  GM
FG ·986 DATA"--,[3" "]/[5"-"],[3" "]/[5"-"],
FG [3" "][3"-"]"                                 BC
PF ·987 DATA"[3" "]/[21"-"],[3" "]"              OG
CC ·988 DATA"--,[3" "]/[15"-"],[3" "]/--"        IB
BK ·990 DATA"  [WHITE]![RED][11"%"][WHITE]#"     DG
   ·991 DATA"  [WHITE]![RED][6"%"][WHITE]#[4
   " "]![RED][6"%"][WHITE]#  "                   KI
   ·992 DATA"[7" "][WHITE]![RED][13"%"][WHIT
   E]#"                                          NK
   ·993 DATA"[13" "][WHITE]![RED][12"%"][WHI
   TE]#"                                         IC
   ·994 DATA"  [WHITE]![RED]%%[WHITE]#[3" "]
   ![RED]%%[WHITE]#[3" "]![RED]%%[WHITE]#[3
   " "]![RED]%%[WHITE]#"                         CP
   ·995 DATA"[3" "][WHITE]![RED][21"%"][WHIT
   E]#"                                          IC
   ·996 DATA"[WHITE]![RED][5"%"][WHITE]#[4"
   "]![RED][5"%"][WHITE]#[4" "]![RED][5"%"]
   [WHITE]#"                                     CN
   ·997 DATA" [WHITE]![RED]%[WHITE]#[3" "]![
   RED]%%[WHITE]#[3" "]![RED][3"%"][WHITE]#
   [3" "]![RED][4"%"][WHITE]#"                   MG
   ·1000 DATA173,253,207,240,37,173,0,220,41
   ,16,240,5,169,0,141,249,207,173               NH
   ·1001 DATA2,207,208,6,32,45,192,32,112,19
   2,173,3,207,208,3,32,3,193,32                 BN
   ·1002 DATA198,192,32,74,193,108,254,207,1
   73,0,220,174,0,208,41,8,240,10                IM
   ·1003 DATA224,26,240,23,206,0,208,206,0,2
   08,173,0,220,41,4,240,10,224                  GA
   ·1004 DATA238,240,6,238,0,208,238,0,208,1
   73,0,220,41,1,208,12,174,1,208                GP
   ·1005 DATA224,165,144,15,206,1,208,208,10
   ,174,1,208,224,210,240,3,238,1                JN
   ·1006 DATA208,96,173,252,207,240,23,174,3
   ,208,224,35,144,60,206,3,208,206              KE
   ·1007 DATA3,208,206,3,208,173,3,208,141,1
   ,212,96,173,249,207,208,39,173                GG
   ·1008 DATA0,220,41,16,208,32,169,1,141,24
   9,207,141,252,207,169,2,13,21                 KG
   ·1009 DATA208,141,21,208,173,1,208,141,3,
   208,32,64,193,173,0,208,141,2                 EK
   ·1010 DATA208,96,169,253,45,21,208,141,21
   ,208,169,0,141,252,207,96,173,251             IF
   ·1011 DATA207,201,7,208,11,162,7,189,243,
   192,157,39,57,202,208,247,173                 LC
   ·1012 DATA251,207,201,12,208,16,169,0,141
   ,251,207,162,7,189,251,192,157                EF
   ·1013 DATA39,57,202,208,247,238,251,207,9
   6,0,12,63,243,192,0,0,0,0,192                 GC
   ·1014 DATA243,63,12,0,0,0,173,21,208,41,4
   ,201,4,240,1,96,173,250,207,240               NO
   ·1015 DATA23,173,4,208,201,231,176,10,238
   ,4,208,238,4,208,238,4,208,96                 FL
   ·1016 DATA169,0,141,250,207,96,173,4,208,
   201,27,144,10,206,4,208,206,4                 CJ
   ·1017 DATA208,206,4,208,96,169,1,141,250,
   207,96,162,129,142,4,212,202,142              EN
   ·1018 DATA4,212,96,173,30,208,141,16,207,
   41,1,240,3,141,2,207,173,16                   HM
   ·1019 DATA207,41,6,201,6,208,8,169,1,141,
   3,207,32,184,192,173,16,207,41                GB
   ·1020 DATA10,201,10,208,8,169,1,141,4,207
```

```
,32,184,192,173,31,208,141,17                LJ
•1021 DATA207,41,1,240,3,141,2,207,173,17
,207,41,2,240,3,32,184,192,96                MH
•1022 DATA173,20,3,141,254,207,173,21,3,1
41,255,207,120,169,0,141,20,3                MH
•1023 DATA169,192,141,21,3,88,96             PN
•1040 DATA169,152,133,252,169,7,133,253,1
69,152,133,254,169,219,133,255,162           FB
•1041 DATA23,160,28,177,252,141,0,207,177
,254,141,1,207,24,165,252,105                EJ
•1042 DATA40,133,252,165,253,105,0,133,25
3,165,254,105,40,133,254,165,255             NM
•1043 DATA105,0,133,255,173,0,207,145,252
,173,1,207,145,254,56,165,252                AB
•1044 DATA233,40,133,252,165,253,233,0,13
3,253,165,254,233,40,133,254,165             CM
•1045 DATA255,233,0,133,255,136,192,255,2
08,181,160,28,202,224,255,240,28             MI
•1046 DATA56,165,252,233,40,133,252,165,2
53,233,0,133,253,165,254,233,40              IC
•1047 DATA133,254,165,255,233,0,133,255,7
6,20,197,160,28,169,32,145,252               OH
•1048 DATA169,0,145,254,136,192,255,208,2
43,173,21,208,41,4,240,21,24,173             BH
•1049 DATA5,208,105,8,141,5,208,201,240,1
44,8,173,21,208,41,251,141,21                NP
•1050 DATA208,173,21,208,41,8,240,21,24,1
73,7,208,105,8,141,7,208,201                 LM
•1051 DATA240,144,8,173,21,208,41,247,141
,21,208,96                                   JO
```

PENGUINS
FROM PAGE 16

Starting address in hex: C000
Ending address in hex: CB72
SYS to start: 49152

***Flankspeed* required for entry! See page 103.**

```
C000: 20 76 C8 20 4D C2 AD 1F 5C
C008: D0 D0 FB AD 1E D0 D0 FB 0F
C010: AD 11 D0 10 FB AE 1F D0 4A
C018: AC 1E D0 AD 00 DC C9 7F 87
C020: D0 03 AD B5 02 8D B5 02 9E
C028: C9 7B D0 3F AD 10 D0 29 35
C030: 01 D0 0A AD 00 D0 C9 16 6A
C038: D0 03 4C 80 C0 AD 00 D0 18
C040: 38 E9 01 8D 00 D0 B0 76 E8
C048: AD 10 D0 49 01 8D 10 D0 8F
C050: AD AB 02 29 FE 8D AB 02 0F
C058: AD AC 02 29 FE 8D AC 02 19
C060: AD AD 02 29 FE 8D AD 02 23
C068: 4C BE C0 C9 77 D0 37 AD 2B
C070: 10 D0 29 01 F0 0A AD 00 24
C078: D0 C9 41 D0 03 4C 3D C0 72
C080: AD 00 D0 18 69 01 8D 00 0F
C088: D0 90 33 AD AB 02 09 01 82
C090: 8D AB 02 AD AC 02 09 01 32
C098: 8D AC 02 AD AD 02 09 01 3C
C0A0: 8D AD 02 4C BE C0 C9 7E F1
C0A8: D0 06 CE 01 D0 4C BE C0 EB
C0B0: C9 7D D0 0A AD 01 D0 C9 1C
C0B8: DE F0 03 EE 01 D0 8A 29 FF
C0C0: 01 F0 0C A9 F0 18 6D B0 8F
C0C8: 02 8D F8 07 4C DC C0 A9 EB
C0D0: F5 8D F8 07 98 29 54 D0 3B
C0D8: 03 CE 01 D0 98 29 AA F0 D9
C0E0: 5C 20 D6 C4 20 D7 C1 20 D2
C0E8: 04 C2 A0 00 A2 03 18 20 2E
C0F0: F0 FF A9 0F A0 C7 20 1E 41
C0F8: AB 20 DA C2 AD B6 02 D0 99
C100: 39 A9 00 85 FB 85 FC A2 89
C108: 18 A0 0E 18 20 F0 FF A9 A1
C110: 9B A0 C7 20 1E AB A5 C5 69
C118: C9 04 D0 FA A9 03 8D B6 A2
C120: 02 A9 0F 8D B7 02 A9 9F 6B
C128: 20 D2 FF A2 18 A0 00 18 8E
C130: 20 F0 FF A9 64 A0 C7 20 D7
C138: 1E AB 4C 3D C1 A9 FE 85 7B
C140: A2 A5 A2 D0 FC 20 97 E0 91
C148: A5 8D C9 FF D0 03 20 A6 DF
C150: C6 A9 12 20 D2 FF A9 9F 0F
C158: 20 D2 FF A0 22 A2 18 18 E0
C160: 20 F0 FF A5 FC A6 FB 20 D6
C168: CD BD A0 06 A2 18 18 20 8D
C170: F0 FF A5 FE A6 FD 20 CD 98
C178: BD CE BA 02 D0 1E A9 19 73
C180: 8D BA 02 A2 03 AC B9 02 D8
C188: 18 20 F0 FF A9 97 A0 C7 5B
C190: 20 1E AB AE B9 02 F0 04 D9
C198: CA 8E B9 02 AD 01 D0 38 65
C1A0: C9 36 B0 30 A9 F0 8D F8 A2
C1A8: 07 A9 F0 85 A2 A5 A2 D0 8B
C1B0: FC AD 1F D0 29 01 F0 03 69
C1B8: 4C E1 C0 20 12 C5 20 C4 84
C1C0: C5 A2 03 A0 00 18 20 F0 F5
C1C8: FF AE B7 02 F0 03 CE B7 AB
C1D0: 02 4C F2 C0 4C 10 C0 A9 99
C1D8: 00 8D 1C D0 A9 FA 8D F8 7E
C1E0: 07 CE 01 D0 A9 FF 85 A2 5A
C1E8: A5 A2 D0 FC AD 01 D0 8D 0C
C1F0: 27 D0 D0 ED A9 01 8D 1C FB
C1F8: D0 A9 F0 8D F8 07 AD 1E BD
C200: D0 D0 FB 60 A9 12 20 D2 AC
C208: FF A9 9F 20 D2 FF AE B6 A9
C210: 02 CA 8E B6 02 F0 28 E0 1E
C218: 02 D0 10 A2 18 A0 19 18 87
C220: 20 F0 FF A9 20 20 D2 FF ED
C228: 4C 4C C2 E0 01 D0 10 A2 E8
C230: 18 A0 18 18 20 F0 FF A9 D3
C238: 20 20 D2 FF 4C 4C C2 A2 49
C240: 18 A0 17 18 20 F0 FF A9 E2
C248: 20 20 D2 FF 60 78 A9 16 F3
C250: 8D 14 03 A9 C3 8D 15 03 08
C258: A9 4A 8D 12 D0 AD 11 D0 4C
C260: 29 7F 8D 11 D0 A9 81 8D 31
C268: 1A D0 58 A9 0A 8D B8 02 A7
```

```
C270: 8D BA 02 A9 0F 8D B7 02 BA
C278: A9 00 8D B0 02 8D B2 02 A4
C280: 8D 25 D0 85 FB 85 FC 85 8D
C288: FD 85 FE 8D 10 D0 8D 17 1E
C290: D0 8D 20 D0 A9 AA 8D 1D DE
C298: D0 A9 01 8D 1C D0 8D B4 D0
C2A0: 02 8D B1 02 8D 26 D0 8D F5
C2A8: B0 02 8D AF 02 A9 E0 8D B2
C2B0: AB 02 8D AC 02 8D AD 02 D7
C2B8: A2 01 BD FC C7 9D 27 D0 74
C2C0: E8 E0 08 D0 F5 A9 03 8D 93
C2C8: B6 02 20 5A C6 A9 FF 8D F9
C2D0: 15 D0 20 93 C4 A9 05 8D 6B
C2D8: B3 02 A9 F0 8D F8 07 A9 60
C2E0: DE 8D 01 D0 A9 A0 8D 00 F6
C2E8: D0 A9 07 8D 27 D0 AD 1E BB
C2F0: D0 D0 FB A9 7D 8D B5 02 FA
C2F8: AD AB 02 29 FE 8D AB 02 B7
C300: AD AC 02 29 FE 8D AC 02 C0
C308: AD AD 02 29 FE 8D AD 02 CA
C310: A9 27 8D B9 02 60 AD 19 51
C318: D0 8D 19 D0 29 01 D0 07 62
C320: AD 0D DC 58 4C 31 EA AD 26
C328: 12 D0 C9 EA D0 03 4C BC 9C
C330: C3 C9 95 F0 58 C9 6E F0 C5
C338: 2F A2 02 BD B4 C7 9D 00 E3
C340: D0 E8 E0 10 D0 F5 A2 01 55
C348: BD E4 C7 9D F8 07 E8 E0 1A
C350: 08 D0 F5 AD AB 02 8D 10 18
C358: D0 A9 06 8D 21 D0 A9 00 02
C360: 8D 2E D0 A9 6E 4C C3 C3 D8
C368: A2 02 BD C4 C7 9D 00 D0 C5
C370: E8 E0 10 D0 F5 A2 01 BD 72
C378: EC C7 9D F8 07 E8 E0 08 9C
C380: D0 F5 AD AC 02 8D 10 D0 12
C388: A9 95 4C C3 C3 A2 02 BD FD
C390: D4 C7 9D 00 D0 E8 E0 10 75
C398: D0 F5 A2 01 BD F4 C7 9D 1B
C3A0: F8 07 E8 E0 08 D0 F5 AD E6
C3A8: AD 02 8D 10 D0 A9 0E 8D 0C
C3B0: 21 D0 A9 09 8D 2E D0 A9 8B
C3B8: EA 4C C3 C3 A9 0B 8D 21 DA
C3C0: D0 A9 4A 8D 12 D0 CE AE 73
C3C8: 02 D0 0C 20 DA C3 AD B7 CB
C3D0: 02 8D AE 02 20 AD C4 4C EF
C3D8: BC FE A2 01 8A 18 0A A8 8D
C3E0: B9 B4 C7 18 69 02 99 B4 E8
C3E8: C7 90 09 BD AC C7 4D AB 75
C3F0: 02 8D AB 02 E8 E0 08 D0 D0
C3F8: E3 A2 01 8A 18 0A A8 B9 8F
C400: C4 C7 38 E9 01 99 C4 C7 D5
C408: B0 09 BD AC C7 4D AC 02 EF
C410: 8D AC 02 E8 E0 08 D0 E3 D2
C418: A2 01 8A 18 0A A8 B9 D4 9F
C420: C7 18 69 01 99 D4 C7 90 31
C428: 09 BD AC C7 4D AD 02 8D ED
C430: AD 02 E8 E0 04 D0 E3 8A EC
C438: 18 0A A8 B9 D4 C7 38 E9 7B
C440: 01 99 D4 C7 B0 09 BD AC 9B
C448: C7 4D AD 02 8D AD 02 E8 33
C450: E0 07 D0 E3 8A 18 0A A8 42
C458: CE AF 02 D0 19 A9 02 8D FB
C460: AF 02 B9 D4 C7 38 E9 01 8B
C468: 99 D4 C7 B0 09 BD AC C7 8A
C470: 4D AD 02 8D AD 02 CE B1 2B
C478: 02 D0 17 A9 05 8D B1 02 52
C480: A9 01 38 ED B0 02 8D B0 42
C488: 02 AD B0 02 18 69 F8 9D 03
C490: F4 C7 60 A2 18 A9 00 9D AF
C498: 00 D4 CA 10 FA A9 F0 8D 6B
C4A0: 06 D4 A9 11 8D 04 D4 A9 46
C4A8: 0F 8D 18 D4 60 CE B3 02 17
C4B0: D0 23 A9 04 8D B3 02 AE 44
C4B8: B2 02 38 E0 5A 90 05 A2 19
C4C0: 00 8E B2 02 BD 04 C8 8D 1C
C4C8: 01 D4 E8 BD 04 C8 8D 00 9F
C4D0: D4 E8 8E B2 02 60 A2 05 D9
C4D8: A9 00 8D 04 D4 A9 0F 8D 2F
C4E0: 07 D4 A9 0A 8D 08 D4 A9 84
C4E8: F0 8D 0C D4 A9 00 8D 0D 8C
C4F0: D4 A9 21 8D 0B D4 A9 EB 93
C4F8: 85 A2 A5 A2 8D 08 D4 8D 61
C500: 0D D4 D0 F6 A9 00 8D 0B EB
C508: D4 A9 11 8D 04 D4 CA D0 99
C510: C7 60 A9 F6 8D F8 07 20 86
C518: 31 C5 A2 0A A9 F6 18 6D E1
C520: B0 02 8D F8 07 A9 EB 85 7B
C528: A2 A5 A2 D0 FC CA D0 EC 69
C530: 60 A9 00 8D 04 D4 A0 03 44
C538: A9 55 8D 0C D4 8D 0D D4 15
C540: 8D 13 D4 8D 14 D4 A9 21 F6
C548: 8D 0B D4 A9 11 8D 12 D4 E4
C550: A9 60 85 02 A9 C8 85 03 DC
C558: A2 06 8A 48 A2 00 A1 02 1A
C560: 8D 08 D4 8D BB 02 E8 A1 A0
C568: 02 8D 07 D4 E8 A1 02 8D ED
C570: 0F D4 E8 A1 02 8D 0E D4 51
C578: AD BB 02 C9 32 D0 08 A9 62
C580: F0 85 A2 A5 A2 D0 FC A9 59
C588: FA 85 A2 A5 A2 D0 FC A5 67
C590: 02 18 69 04 85 02 A5 03 48
C598: 69 00 85 03 68 AA CA F0 59
C5A0: 05 8A 48 4C 5C C5 A9 20 B0
C5A8: 8D 0B D4 A9 10 8D 12 D4 44
C5B0: A9 E6 85 A2 A5 A2 D0 FC 7F
C5B8: 88 F0 03 4C 38 C5 A9 11 3A
C5C0: 8D 04 D4 60 A2 00 AD B9 91
C5C8: 02 0A A8 8A 2A AA 98 0A 7F
C5D0: A8 8A 2A AA 98 0A A8 8A AE
C5D8: 2A AA 98 0A A8 8A 2A AA 58
C5E0: 98 0A A8 8A 2A AA 98 0A 2E
C5E8: A8 8A 2A AA 98 18 65 FB 03
C5F0: 85 FB 8A 65 FC 85 FC A2 84
C5F8: 00 AD B9 02 0A A8 8A 2A C9
C600: AA 98 0A A8 8A 2A AA 98 ED
C608: 0A A8 8A 2A AA 98 0A A8 65
```

```
C610: 8A 2A AA 98 0A A8 8A 2A 6F
C618: AA 98 18 65 FB 85 FB 8A E0
C620: 65 FC 85 FC A2 00 AD B9 0F
C628: 02 0A A8 8A 2A AA 98 0A DE
C630: A8 8A 2A AA 98 18 65 FB 4A
C638: 85 FB 8A 65 FC 85 FC A5 CE
C640: FE 38 C5 FC F0 04 B0 11 F0
C648: 90 07 A5 FD 38 C5 FB B0 2E
C650: 08 A5 FB 85 FD A5 FC 85 A5
C658: FE 60 A9 93 20 D2 FF A9 91
C660: 05 20 D2 FF A2 00 8A 48 CD
C668: A9 DD A0 C6 20 1E AB 68 A9
C670: AA E8 8A 48 E0 03 D0 F0 7C
C678: 68 A9 0F A0 C7 20 1E AB EB
C680: A2 0D A0 00 18 20 F0 FF F9
C688: A9 39 A0 C7 20 1E AB A2 60
C690: 17 A0 00 18 20 F0 FF A9 1B
C698: 39 A0 C7 20 1E AB A9 64 32
C6A0: A0 C7 20 1E AB 60 A9 92 8F
C6A8: 20 D2 FF 20 97 E0 A5 8D 67
C6B0: 29 03 F0 28 48 AC B8 02 A5
C6B8: A2 01 18 20 F0 FF A9 20 4F
C6C0: 20 D2 FF 20 D2 FF 68 AA B9
C6C8: BD A8 C7 8D B8 02 A8 A2 8A
C6D0: 01 18 20 F0 FF A9 92 A0 D7
C6D8: C7 20 1E AB 60 08 0E 12 13
C6E0: DC DC DC DC DC DC DC DC C7
C6E8: 92 A8 20 20 20 20 A8 12 5F
C6F0: DC DC DC 92 A8 20 20 20 23
C6F8: 20 A8 12 DC DC DC 92 A8 A5
C700: 20 20 20 20 A8 12 DC DC F4
C708: DC DC DC DC DC DC 00 99 CE
C710: A8 A8 A8 A8 A8 A8 A8 A8 55
C718: A8 A8 A8 A8 A8 A8 A8 A8 5D
C720: A8 A8 A8 A8 A8 A8 A8 A8 65
C728: A8 A8 A8 A8 A8 A8 A8 A8 6D
C730: A8 A8 A8 A8 A8 A8 A8 A8 75
C738: 00 12 05 20 20 20 20 20 EF
C740: 20 20 20 20 20 20 20 20 41
C748: 20 20 20 20 20 20 20 20 49
C750: 20 20 20 20 20 20 20 20 51
C758: 20 20 20 20 20 20 20 20 59
C760: 20 20 20 00 12 9F C8 49 84
C768: 47 48 3A 20 20 20 20 20 D2
C770: 20 20 20 20 D0 45 4E 47 9C
C778: 55 49 4E 53 20 BA BA BA 09
C780: 20 20 D3 43 4F 52 45 3A F8
C788: 20 20 20 20 20 9D 94 20 7B
C790: 13 00 92 96 4F 4F 00 12 7D
C798: 1C A8 00 12 05 C7 41 4D CA
C7A0: 45 20 CF 56 45 52 21 00 E4
C7A8: 00 0A 13 1C 01 02 04 08 F0
C7B0: 10 20 40 80 00 00 46 55 3D
C7B8: 8C 55 C3 55 FA 55 64 55 BD
C7C0: AA 55 E6 55 00 00 46 75 B8
C7C8: 8C 75 D2 75 1E 75 64 75 80
C7D0: AA 75 DC 75 00 00 64 A5 4D
C7D8: C8 A5 00 A5 64 C8 78 C8 5B
C7E0: FA C8 19 9C 00 F2 F4 F2 35
C7E8: F4 F2 F4 F2 00 F3 F4 F3 95
C7F0: F4 F3 F4 F3 00 F2 F4 F2 9D
C7F8: F4 F3 F4 F8 07 00 03 00 D9
C800: 03 00 01 00 19 1E 15 1F 6F
C808: 19 1E 16 60 00 00 19 1E EC
C810: 16 60 00 00 16 60 10 C3 D0
C818: 10 C3 00 00 10 C3 00 00 BF
C820: 10 C3 00 00 10 C3 00 00 C7
C828: 12 D1 15 1F 16 60 19 1E ED
C830: 00 00 19 1E 19 1E 12 D1 82
C838: 19 1E 12 D1 16 60 12 D1 AD
C840: 19 1E 15 1F 19 1E 10 C3 B6
C848: 00 00 10 C3 00 00 10 C3 EF
C850: 00 00 12 D1 15 1F 16 60 DE
C858: 19 1E 00 00 19 1E 01 01 C8
C860: 19 1E 12 D1 21 87 19 1E 5B
C868: 2A 3E 1F A5 32 3C 25 A2 CB
C870: 2A 3E 1F A5 32 3C A9 B1 67
C878: 85 02 A9 C8 85 03 A9 00 A4
C880: 85 04 A9 3C 85 05 A2 0B 28
C888: A0 00 B1 02 91 04 C8 C0 FB
C890: 40 D0 F7 A5 02 18 69 40 03
C898: 85 02 A5 03 69 00 85 03 BA
C8A0: A5 04 18 69 40 85 04 A5 3B
C8A8: 05 69 00 85 05 CA D0 D8 16
C8B0: 60 00 00 00 00 00 00 00 11
C8B8: 00 00 00 08 00 00 08 00 C8
C8C0: 00 26 00 00 15 00 00 15 11
C8C8: 00 00 55 41 01 55 54 05 0F
C8D0: 55 54 11 55 50 01 55 50 D7
C8D8: 01 55 50 01 55 50 01 55 7C
C8E0: 50 01 55 50 00 55 40 00 6D
C8E8: 15 00 01 51 00 00 01 50 A1
C8F0: 00 00 00 00 00 00 00 00 F0
C8F8: 00 00 00 08 00 00 08 00 09
C900: 00 26 00 00 15 00 00 15 50
C908: 00 10 55 40 05 55 50 05 5D
C910: 55 54 01 55 51 01 55 50 08
C918: 01 55 50 01 55 50 01 55 BB
C920: 50 01 55 50 00 55 40 00 AC
C928: 15 00 00 11 50 01 50 00 EF
C930: 0A 00 00 00 00 00 00 00 3A
C938: 00 00 00 00 00 00 00 00 38
C940: 00 70 00 00 38 00 00 1C 05
C948: 00 00 1F F0 80 7F FC 81 D6
C950: FF 2F C7 FF FF FF FF FC 44
C958: 7F FF F0 00 04 20 00 02 EE
C960: 40 01 55 80 00 2A 00 00 A1
C968: 00 00 00 00 00 00 00 00 68
C970: 00 00 00 00 00 00 00 00 70
C978: 00 00 00 00 00 00 00 00 78
C980: 00 0E 00 00 1C 00 00 38 E2
C988: 00 0F F8 00 3F FE 01 F4 C4
C990: FF 81 FF FF E3 3F FF FF 35
C998: 0F FF FE 04 20 00 02 40 0D
C9A0: 00 01 AA 80 00 54 00 00 21
C9A8: 00 00 00 00 00 00 00 00 A8
```

```
C9B0: 00 00 00 00 00 00 00 00 B0
C9B8: 00 00 00 00 00 00 00 00 B8
C9C0: 00 20 00 01 32 00 03 FF 17
C9C8: 20 07 FF 60 1F FF F0 3F 9F
C9D0: FF FC FF FF FF 00 00 00 CD
C9D8: 00 00 00 00 00 00 00 00 D8
C9E0: 00 00 00 00 00 00 00 00 E0
C9E8: 00 00 00 00 00 00 00 00 E8
C9F0: FF 00 00 00 00 00 00 00 F0
C9F8: 00 00 00 08 00 00 08 00 09
CA00: 00 26 00 00 15 00 00 15 50
CA08: 00 05 55 54 11 55 51 11 7F
CA10: 55 51 01 55 50 00 55 40 F2
CA18: 00 00 0C 0C C3 00 00 00 F3
CA20: C0 0C 0C 0C 00 C0 C0 00 86
CA28: 00 00 00 00 00 00 00 00 28
CA30: 00 00 26 00 00 19 00 00 6F
CA38: 3B 00 10 7F 41 05 FF D4 1E
CA40: 05 FF D4 01 FF D0 01 FF EC
CA48: D0 01 FF D0 01 FF D0 01 BD
CA50: FF D0 01 7F 50 00 5D 40 8F
CA58: 00 15 00 00 11 00 00 51 CF
CA60: 40 01 51 50 01 40 50 00 D4
CA68: 00 00 00 00 00 00 00 00 68
CA70: 0A 00 00 00 00 00 00 00 7A
CA78: 00 00 00 00 00 00 00 00 78
CA80: 00 26 00 00 19 00 00 3B FA
CA88: 00 00 7F 40 01 FF D0 05 1F
CA90: FF D4 11 FF D1 11 FF D1 2B
CA98: 01 FF D0 01 FF D0 01 FF 3D
CAA0: D0 01 7F 50 00 5D 40 00 DF
CAA8: 15 00 00 11 00 01 51 50 71
CAB0: 01 04 3E 00 1C FF 80 2F BF
CAB8: FF C0 F3 FF E2 6C 20 9C 79
CAC0: 00 C0 C0 00 00 00 00 00 42
CAC8: 00 00 00 00 00 00 00 00 C8
CAD0: 00 00 00 00 00 00 00 00 D0
CAD8: 00 00 00 00 00 00 00 00 D8
CAE0: 00 00 00 00 00 00 00 00 E0
CAE8: 00 00 00 00 00 00 00 00 E8
CAF0: 00 04 3E 00 1C FF 80 2F FE
CAF8: FF C8 7B FF E4 E4 40 98 DF
CB00: 00 33 00 00 00 00 00 00 33
CB08: 00 00 00 00 00 00 00 00 08
CB10: 00 00 00 00 00 00 00 00 10
CB18: 00 00 00 00 00 00 00 00 18
CB20: 00 00 00 00 00 00 00 00 20
CB28: 00 00 00 00 00 00 00 00 28
CB30: 00 00 3E 00 00 41 00 18 C7
CB38: 3E 0C 24 00 12 42 1C 21 38
CB40: 81 36 41 80 EB 81 BE 08 ED
CB48: 3D A2 42 45 A5 42 A5 A9 E6
CB50: 42 95 B1 24 8D A1 18 85 CA
CB58: A1 00 85 A1 00 85 60 81 88
CB60: 0A 30 7E 0C 18 42 18 00 97
CB68: C3 00 03 81 C0 01 00 80 F2
CB70: 00 00 FF 70
```

QUICK CHANGE

FROM PAGE 52

```
·5 REM *** QUICK CHANGE *** BUCK CHILDRES
  S ***                                          IE
·10 REM *** P.O. BOX 13575 SALEM, OR 9730
  9 ***                                          DC
·15 PRINTCHR$(147)"LOADING AND CHECKING D
  ATA LINE:":J=50000:L=50:C=11                   HF
·20 PRINTCHR$(19)TAB(31)L:PRINT:FORB=0TOC
  :READA:IFA<0ORA>255THEN30                      JJ
·25 POKEJ+B,A:X=X+A:NEXTB:READA:IFA=XTHEN
  35                                             BO
·30 PRINT"ERROR IN DATA LINE:"L:END              HL
·35 X=0:J=J+12:L=L+5:IFL<671THEN20               DH
·40 PRINT"THE DATA IS OK AND LOADED.":PRI
  NT                                             HH
·45 PRINT"SYS 5[4"0"] TO SWITCH ON AND OF
  F.":END                                        OG
·50 DATA169,106,162,132,160,195,204,1,3,2
  08,6,169,1515                                  NE
·55 DATA139,162,135,160,227,141,0,3,140,1
  ,3,76,1187                                     HP
·60 DATA251,199,173,0,2,201,65,144,4,201,
  91,144,1475                                    HL
·65 DATA3,76,139,227,32,237,199,32,217,19
  9,133,253,1747                                 AI
·70 DATA162,96,134,251,142,138,2,157,0,20
  1,232,208,1723                                 AP
·75 DATA250,232,169,39,141,248,201,32,79,
  199,169,91,1850                                LM
·80 DATA133,252,141,247,201,162,10,32,251
  ,199,133,254,2015                              MC
·85 DATA166,252,169,100,32,171,199,32,66,
  199,32,7,1425                                  PE
·90 DATA200,32,228,255,240,245,133,212,20
  1,13,208,44,2011                               CI
·95 DATA173,96,201,240,227,169,32,166,252
  ,157,0,4,1717                                  CO
·100 DATA166,251,48,13,169,192,162,24,160
  ,171,133,251,1740                              GG
·105 DATA132,252,76,159,195,173,192,201,2
  08,86,162,139,1975                             NE
·110 DATA134,199,32,251,199,76,52,196,201
  ,20,208,39,1607                                BO
·115 DATA174,247,201,48,3,76,120,195,169,
  0,162,32,1427                                  DL
·120 DATA157,192,201,202,16,250,164,254,3
  2,162,199,132,1961                             DM
·125 DATA254,140,247,201,169,192,162,171,
  133,251,134,252,2306                           MA
·130 DATA76,164,195,166,254,224,26,144,3,
  76,171,195,1694                                HK
·135 DATA230,254,166,251,157,0,201,230,25
```

```
1,230,252,32,2254                                                    OF
·140 DATA210,255,165,252,16,3,141,247,201
,76,164,195,1925                                                     JH
·145 DATA32,240,199,162,105,32,251,199,16
9,0,162,9,1560                                                       AD
·150 DATA157,0,2,172,233,201,208,3,157,23
0,201,202,1766                                                       EE
·155 DATA16,242,174,233,201,169,100,157,2
51,4,173,134,1854                                                    OO
·160 DATA2,157,251,216,32,66,199,32,7,200
,32,228,1422                                                         AN
·165 DATA255,240,245,201,13,208,39,173,23
3,201,240,20,2068                                                    HN
·170 DATA173,235,201,201,2,144,229,32,181
,199,173,238,2008                                                    NI
·175 DATA201,208,107,173,239,201,208,102,
169,255,141,238,2242                                                 DC
·180 DATA201,141,239,201,141,230,201,76,2
34,196,201,20,2081                                                   EC
·185 DATA208,30,172,233,201,240,9,32,162,
199,140,233,1859                                                     NJ
·190 DATA201,76,60,196,169,32,141,251,4,1
60,79,32,1401                                                        IO
·195 DATA162,199,200,132,254,76,244,195,2
01,45,208,20,1936                                                    BP
·200 DATA174,231,201,208,159,141,231,201,
32,210,255,238,2281                                                  OH
·205 DATA233,201,32,181,199,76,60,196,201
,48,144,140,1711                                                     GE
·210 DATA201,58,176,136,174,232,201,224,5
,176,129,157,1869                                                    LH
·215 DATA0,2,32,210,255,238,232,201,238,2
33,201,76,1918                                                       IL
·220 DATA78,196,32,237,199,32,217,199,173
,248,201,32,1844                                                     GE
·225 DATA79,199,172,192,201,208,8,162,77,
32,140,199,1669                                                      KG
·230 DATA76,18,197,162,87,32,140,199,162,
96,32,251,1452                                                       JH
·235 DATA199,162,96,32,145,199,162,119,32
,251,199,174,1770                                                    AD
·240 DATA236,201,173,237,201,32,205,189,1
69,45,32,210,1930                                                    DP
·245 DATA255,173,230,201,208,12,174,238,2
01,173,239,201,2305                                                  LA
·250 DATA32,205,189,76,59,197,162,128,32,
251,199,162,1692                                                     FO
·255 DATA38,32,251,199,32,66,199,32,7,200
,32,228,1316                                                         EI
·260 DATA255,201,89,240,7,201,78,208,239,
76,120,195,1909                                                      FJ
·265 DATA169,1,162,8,141,240,201,142,241,
201,32,237,1775                                                      MF
·270 DATA199,141,138,2,141,245,201,32,127
,199,160,0,1585                                                      KH
·275 DATA177,253,208,3,76,116,164,32,15,1
99,177,253,1673                                                      ME
·280 DATA170,32,15,199,177,253,142,250,20
1,141,251,201,2032                                                   OC
·285 DATA205,237,201,144,7,208,23,236,236
,201,176,18,1892                                                     MK
·290 DATA32,205,189,32,15,199,162,0,161,2
53,240,82,1570                                                       LA
·295 DATA32,15,199,76,152,197,205,239,201
,144,9,208,1677                                                      NN
·300 DATA199,236,238,201,144,2,208,192,32
,205,189,169,2015                                                    KM
·305 DATA32,32,210,255,32,15,199,169,39,1
62,201,141,1487                                                      FB
·310 DATA249,201,141,253,201,134,252,169,
96,162,0,133,1991                                                    DI
·315 DATA251,142,242,201,142,246,201,161,
253,240,23,32,2134                                                   II
·320 DATA88,198,173,242,201,205,246,201,2
08,88,162,0,2012                                                     KB
·325 DATA161,251,240,91,32,15,199,76,211,
197,142,244,1859                                                     IN
·330 DATA201,32,240,199,173,245,201,208,1
6,32,15,199,1761                                                     HD
·335 DATA165,253,166,254,141,240,201,142,
241,201,76,94,2174                                                   OM
·340 DATA197,162,0,32,251,199,169,19,141,
119,2,169,1460                                                       IE
·345 DATA13,141,120,2,141,121,2,141,122,2
,169,4,978                                                           KO
·350 DATA133,198,76,116,164,32,127,199,32
,15,199,32,1323                                                      DJ
·355 DATA15,199,162,0,32,15,199,161,253,2
08,249,76,1569                                                       ND
·360 DATA249,197,32,7,200,32,15,199,76,19
9,197,172,1575                                                       CP
·365 DATA242,201,32,162,199,141,245,201,1
85,192,201,240,2241                                                  GE
·370 DATA233,32,195,198,200,76,76,198,166
,212,208,64,1858                                                     CL
·375 DATA174,253,201,142,249,201,162,18,2
01,131,240,5,1977                                                    IA
·380 DATA202,201,143,208,3,142,253,201,20
1,128,144,40,1866                                                    HL
·385 DATA56,233,127,170,160,255,202,240,8
,200,185,158,1994                                                    BK
·390 DATA160,16,250,48,245,200,185,158,16
0,48,14,238,1722                                                     EC
·395 DATA243,201,32,156,198,169,0,141,243
,201,76,133,1793                                                     FG
·400 DATA198,56,233,128,72,162,0,193,251,
208,26,173,1700                                                      KI
·405 DATA248,201,201,10,240,16,166,212,24
0,7,201,62,1804                                                      IL
·410 DATA240,8,76,189,198,205,249,201,208
,3,238,246,2061                                                      IH
·415 DATA201,230,251,238,242,201,104,174,
244,201,48,11,2145                                                   OO
·420 DATA208,12,166,211,224,79,144,3,142,
244,201,76,1710                                                      NG
·425 DATA210,255,173,243,201,240,2,104,10
```

```
4,104,104,162,1902                                          PJ
·430 DATA192,142,244,201,162,61,32,251,19
9,174,250,201,2109                                          MO
·435 DATA173,251,201,32,205,189,169,32,32
,210,255,32,1781                                            BI
·440 DATA127,199,169,3,133,252,164,252,17
7,253,240,8,1977                                            II
·445 DATA32,88,198,230,252,76,254,198,76,
116,164,230,1914                                            HC
·450 DATA253,208,2,230,254,96,201,8,240,2
51,201,11,1955                                              BH
·455 DATA240,247,201,16,240,243,201,19,24
0,239,201,24,2111                                           HA
·460 DATA240,235,201,27,240,231,201,32,24
0,227,201,35,2110                                           DJ
·465 DATA240,223,201,56,240,219,201,59,24
0,215,169,0,2063                                            BK
·470 DATA133,198,173,141,2,201,4,208,204,
165,203,201,1833                                            DF
·475 DATA64,240,243,160,0,201,39,240,24,1
60,9,201,1581                                               BA
·480 DATA10,240,18,160,18,201,17,240,12,1
60,27,201,1304                                              KP
·485 DATA18,240,6,160,36,201,62,208,169,1
41,248,201,1690                                             OD
·490 DATA162,16,185,249,200,240,121,32,17
1,199,232,200,2007                                          LK
·495 DATA76,114,199,173,240,201,174,241,2
01,133,253,134,2139                                         KM
·500 DATA254,76,15,199,32,251,199,162,0,1
89,96,201,1674                                              MM
·505 DATA240,7,32,210,255,232,76,145,199,
169,34,76,1675                                              BF
·510 DATA210,255,169,20,32,210,255,136,20
8,250,96,157,1998                                           MK
·515 DATA0,4,173,134,2,157,0,216,96,169,0
,162,1113                                                   HO
·520 DATA2,133,122,134,123,141,232,201,32
,121,0,32,1273                                              FL
·525 DATA107,169,165,20,174,235,201,157,2
36,201,232,165,2062                                         ON
·530 DATA21,157,236,201,232,142,235,201,9
6,162,11,160,1854                                           LB
·535 DATA0,24,32,240,255,169,42,160,39,32
,164,199,1356                                               IF
·540 DATA162,147,76,251,199,32,68,229,169
,0,133,198,1664                                             LN
·545 DATA133,199,133,212,133,216,96,189,2
5,200,240,250,2026                                          KL
·550 DATA32,210,255,232,76,251,199,32,225
,255,208,238,2213                                           EL
·555 DATA104,104,169,0,141,138,2,32,68,22
9,76,116,1179                                               OC
·560 DATA164,13,83,89,83,53,48,55,50,53,0
,17,708                                                     KM
·565 DATA13,79,76,68,32,69,78,84,82,89,63
,32,765                                                     DJ
·570 DATA0,13,13,78,69,87,32,69,78,84,82,
89,694                                                      FO
·575 DATA63,32,0,13,13,18,65,82,69,32,89,
79,555                                                      HM
·580 DATA85,32,83,85,82,69,63,32,40,89,47
,78,785                                                     LK
·585 DATA41,0,13,13,18,32,84,79,79,32,76,
79,546                                                      HK
·590 DATA78,71,32,13,13,0,13,13,69,82,65,
83,532                                                      EB
·595 DATA69,32,34,0,13,13,84,85,82,78,32,
34,556                                                      OP
·600 DATA0,13,13,73,78,84,79,32,34,0,13,1
3,432                                                       PJ
·605 DATA84,72,69,32,82,65,78,71,69,63,32
,0,717                                                      PN
·610 DATA13,13,76,73,78,69,83,32,0,69,78,
68,652                                                      BG
·615 DATA0,79,78,0,79,70,70,0,32,69,82,65
,624                                                        AH
·620 DATA83,69,32,0,13,13,13,67,84,82,76,
32,564                                                      JD
·625 DATA65,32,61,32,65,76,76,13,67,84,82
,76,729                                                     GB
·630 DATA32,68,32,61,32,68,65,84,65,13,67
,84,671                                                     EN
·635 DATA82,76,32,78,32,61,32,78,79,82,77
,65,774                                                     LG
·640 DATA76,13,67,84,82,76,32,81,32,61,32
,81,717                                                     JG
·645 DATA85,79,84,69,83,13,67,84,82,76,32
,82,836                                                     CB
·650 DATA32,61,32,82,69,77,83,19,0,160,14
2,143,900                                                   HP
·655 DATA146,141,129,140,160,0,160,160,12
9,140,140,160,1605                                          BM
·660 DATA160,160,0,160,160,146,133,141,14
7,160,160,0,1527                                            HM
·665 DATA160,160,132,129,148,129,160,160,
0,160,145,149,1632                                          MI
·670 DATA143,148,133,147,160,0,0,0,0,0,0,
0,731                                                       EP
```

MOUNTAINEER MACK

FROM PAGE 18

Starting address in hex: 2000
Ending address in hex: 2D5B
SYS to start: 8192

***Flankspeed* required for entry! See page 103.**

```
2000: A9 08 20 D2 FF A9 93 20 02
2008: D2 FF A9 00 8D 20 D0 A9 AC
2010: 0F 8D 21 D0 20 49 2A 8D BF
2018: 0F D4 EA EA A9 80 8D 12 9B
2020: D4 AD 0E DC 29 FE 8D 0E 51
2028: DC A5 01 29 FB 85 01 A2 F9
2030: 00 BD 00 D0 9D 00 30 E8 75
2038: E0 D8 D0 F5 A2 00 BD 00 19
2040: D1 9D 00 31 E8 E0 FF D0 7B
```

```
2048: F5 A2 00 BD 81 20 C9 21 2B
2050: F0 07 9D D8 31 E8 4C 4B 70
2058: 20 A2 00 BD 96 20 C9 21 7A
2060: F0 07 9D EC 31 E8 4C 5B A4
2068: 20 EA 20 2F 2A EA EA EA AD
2070: A5 01 09 04 85 01 AD 0E 66
2078: DC 09 01 8D 0E DC 4C AA CE
2080: 20 08 1C 2A 08 1C 2A 08 45
2088: FF 08 1C 3E 1E 3C 18 18 75
2090: FF 00 00 18 34 21 66 7E E2
2098: FF 00 00 00 0C 3C 7E 7E DD
20A0: FF 00 00 00 00 00 00 00 A0
20A8: 00 21 AD 18 D0 29 F0 18 92
20B0: 69 0C 8D 18 D0 20 DB 26 BE
20B8: A9 0C 8D 86 02 4C 80 22 73
20C0: 00 28 00 00 AA 00 00 AA 3E
20C8: 00 80 AA 02 80 AA 02 40 63
20D0: 28 01 F0 3C 0F 55 55 55 36
20D8: 3F FF FC 01 55 40 00 FF AB
20E0: 00 00 AA 00 00 FF 00 03 8E
20E8: FF C0 0F FF F0 0F 00 F0 A9
20F0: 0F 00 F0 0A 00 A0 2A 00 C5
20F8: A8 AA 00 AA AA 00 AA 00 4C
2100: 00 28 00 08 AA 20 08 AA AD
2108: 20 04 AA 10 0C AA 30 04 D1
2110: 28 10 0C 3C 30 05 55 50 6B
2118: 0F FF F0 01 55 40 00 FF AE
2120: 00 00 AA 00 00 FF 00 03 CD
2128: FF C0 0F FF F0 03 C3 C0 70
2130: 00 FF 00 00 AA 00 02 AA 87
2138: 80 0A AA A0 0A AA A0 00 63
2140: 0C 00 0C 3F 00 1F 3F 80 76
2148: 3F 7F 00 1E 7A 01 9E 30 6F
2150: 03 C8 00 67 C0 01 FF 80 C5
2158: 03 FF 80 03 FF C0 07 FF A6
2160: C0 07 FF E0 03 FF E0 01 ED
2168: FF E0 01 FF F0 03 FF F0 2F
2170: 07 FF F0 0F FF E0 0F FF 67
2178: C0 07 CB C0 03 81 80 00 D1
2180: 00 10 00 00 38 00 00 7C 45
2188: 00 00 FE 00 01 FF 00 00 88
2190: 7C 00 00 FE 00 01 FF 00 0D
2198: 03 FF 80 00 7C 00 00 FE 97
21A0: 00 01 FF 00 03 FF 80 07 2C
21A8: FF C0 00 FE 00 01 FF 00 69
21B0: 03 FF 80 07 FF C0 0F FF 0B
21B8: E0 00 38 00 00 38 00 00 0A
21C0: 00 00 00 00 00 00 00 00 C0
21C8: 00 00 00 00 00 00 00 00 C8
21D0: 00 00 00 00 00 00 03 00 D3
21D8: 00 33 00 00 33 00 00 73 B2
21E0: 00 00 73 30 00 77 30 03 2F
21E8: 77 70 03 77 70 03 77 70 A6
21F0: 0A 7B 70 0A AA 68 2A AA D8
21F8: AA 2A AA AA 08 88 88 00 3C
2200: 00 00 00 00 00 00 00 00 00
2208: 00 00 00 00 00 00 00 00 08
2210: 00 00 0E 00 00 1F 07 F0 35
2218: 1B FF F8 FF FF F8 FF FF 25
2220: FC 1F FF FC FF FF FC 0F 45
2228: FF FE 03 FF FE 01 FF FE 29
2230: 03 C0 7E 03 80 3F 07 80 BC
2238: FF 3F 03 FF 3E 03 FE 00 BA
2240: 00 00 80 00 01 C0 00 03 85
2248: E0 00 06 B0 00 07 F0 00 D7
2250: 03 E0 00 01 C0 00 03 80 79
2258: 00 0F 00 00 00 00 03 FF 6A
2260: 80 07 FF C0 03 FF 80 00 2C
2268: 00 00 03 FF 80 07 FF C0 B3
2270: 43 FF 80 60 00 00 7F FF 14
2278: 80 3F FF C0 1F FF 80 00 98
2280: A9 13 20 D2 FF A2 00 A0 73
2288: 00 20 A1 23 20 D2 FF C8 29
2290: C0 28 D0 F5 A0 00 E8 E0 AA
2298: 17 D0 EE 20 A5 22 4C 07 AA
22A0: 24 AD 1B D4 60 A2 17 A0 1D
22A8: 00 20 D0 2B 20 ED 26 A2 9B
22B0: 00 BD 7F 2C F0 07 20 D2 05
22B8: FF E8 4C B1 22 60 AD 00 CF
22C0: DC A0 00 A2 00 4A B0 01 DC
22C8: 88 4A B0 01 C8 4A B0 01 12
22D0: CA 4A B0 01 E8 4A 8E 9A F3
22D8: 42 8C 9B 42 60 20 78 2A A8
22E0: 20 78 2A AD 75 42 F0 0B 05
22E8: A9 01 8D 9B 42 20 9B 25 DF
22F0: 4C E3 22 20 1D 27 A9 83 D4
22F8: 8D F8 07 A2 00 BD 77 42 A0
2300: C9 30 D0 08 E8 E0 04 D0 71
2308: F4 4C F4 2A A9 00 85 FB 93
2310: A9 28 85 FD A9 04 85 FE 97
2318: 85 FC A0 00 B1 FD 91 FB 78
2320: E6 FD A5 FD C9 00 D0 02 45
2328: E6 FE A5 FE C9 07 D0 06 5A
2330: A5 FD C9 98 F0 0D E6 FB 17
2338: A5 FB C9 00 D0 DE E6 FC 37
2340: 4C 1C 23 20 9C 2A 4C FB FA
2348: 22 A9 97 85 FB A9 6F 85 CB
2350: FD A9 07 85 FE 85 FC A0 A6
2358: 00 B1 FD 91 FB C6 FD A5 FF
2360: FD C9 FF D0 02 C6 FE A5 66
2368: FE C9 03 D0 06 A5 FD C9 78
2370: FF F0 0D C6 FB A5 FB C9 9C
2378: FF D0 DE C6 FC 4C 59 23 B4
2380: 20 5C 26 20 73 2B A9 0C 97
2388: 8D 86 02 A2 00 A0 00 20 02
2390: D0 2B AD 82 42 F0 06 20 16
2398: B2 29 4C A0 23 20 35 2D 07
23A0: 60 20 A1 22 C9 3B 90 F9 74
23A8: C9 40 B0 F5 60 A0 00 A9 04
23B0: 30 99 6C 42 C8 C0 04 D0 87
23B8: F8 A9 3F 8D 15 D0 8D 1D B8
23C0: D0 A9 3D 8D 17 D0 A9 00 97
23C8: 8D 81 42 8D 8D 42 8D 27 2C
23D0: D0 8D 29 D0 8D 10 D0 8D 25
23D8: 80 42 A9 05 8D 1C D0 8D 52
23E0: 2A D0 8D 2B D0 A9 06 8D A2
```

```
23E8: 25 D0 A9 02 8D 26 D0 A9 B8
23F0: 0B 8D 28 D0 A9 85 8D F9 39
23F8: 07 A9 87 8D FA 07 A9 86 F0
2400: 8D FB 07 8D FC 07 60 20 A2
2408: AD 23 20 58 28 20 B0 27 71
2410: 20 61 29 AD 81 42 F0 08 25
2418: 20 86 28 A9 00 8D 81 42 E1
2420: AC 8A 42 F0 03 4C DD 22 D9
2428: 20 BE 22 AD 9A 42 F0 08 AC
2430: C9 01 F0 3D C9 FF F0 33 17
2438: AD 9B 42 F0 08 C9 01 F0 78
2440: 1A C9 FF F0 03 4C 0D 24 95
2448: AD 75 42 C9 09 B0 06 20 57
2450: 9B 25 4C 0D 24 20 49 23 1B
2458: 4C 0D 24 AD 75 42 C9 01 06
2460: 90 06 20 9B 25 4C 0D 24 55
2468: 4C 20 24 20 77 24 4C 0D 0E
2470: 24 20 A3 24 4C 0D 24 A2 9C
2478: 00 AD 00 D0 C9 18 D0 07 B0
2480: AD 10 D0 29 01 F0 0D AD E4
2488: 00 D0 C9 00 D0 03 20 95 AC
2490: 24 CE 00 D0 60 AD 10 D0 43
2498: 29 FE 8D 10 D0 A9 FF 8D 66
24A0: 00 D0 60 A2 00 AD 10 D0 03
24A8: 29 01 F0 07 AD 00 D0 C9 13
24B0: 29 F0 0D AD 00 D0 C9 FF 20
24B8: D0 03 20 C1 24 EE 00 D0 52
24C0: 60 AD 10 D0 09 01 8D 10 57
24C8: D0 A9 00 8D 00 D0 60 A9 AB
24D0: 08 8D 03 D0 AD 10 D0 29 F1
24D8: FD 8D 10 D0 20 A1 22 C9 F2
24E0: A5 90 15 AD 10 D0 09 02 C5
24E8: 8D 10 D0 20 A1 22 C9 20 25
24F0: B0 F9 8D 02 D0 4C 02 25 6F
24F8: 20 A1 22 C9 19 90 F9 8D D7
2500: 02 D0 60 78 AD 14 03 85 F5
2508: A3 AD 15 03 85 A4 A9 1A 5F
2510: 8D 14 03 A9 25 8D 15 03 29
2518: 58 60 AD 1E D0 29 01 F0 88
2520: 05 A9 01 8D 8A 42 AD 9B 73
2528: 42 D0 0D AD 9A 42 D0 08 AB
2530: A9 83 8D F8 07 4C 66 25 C2
2538: EE 74 42 AD 74 42 C9 05 11
2540: D0 24 AD 8A 42 D0 08 20 A8
2548: 5E 2A A9 10 8D 04 D4 A9 9A
2550: 00 8D 74 42 AD F8 07 C9 0C
2558: 83 D0 06 EE F8 07 4C 66 54
2560: 25 A9 83 8D F8 07 AD 82 70
2568: 42 F0 08 A9 08 8D 03 D0 B6
2570: 4C 98 25 EE 71 42 AD 71 3C
2578: 42 CD 72 42 D0 1A A9 00 D1
2580: 8D 71 42 A2 00 EE 03 D0 27
2588: AD 03 D0 C9 D6 D0 03 20 9E
2590: CF 24 E8 EC 7F 42 D0 ED DA
2598: 6C A3 00 AD 9B 42 30 0F 73
25A0: AD 01 D0 18 69 08 8D 01 38
25A8: D0 CE 75 42 4C BB 25 AD DA
25B0: 01 D0 38 E9 08 8D 01 D0 0C
25B8: EE 75 42 A9 01 20 48 2D 9F
25C0: 60 20 CE 25 20 EB 25 20 86
25C8: 43 26 20 17 26 60 20 A1 B1
25D0: 22 C9 88 90 F9 C9 8A B0 D4
25D8: F5 8D FD 07 C9 89 D0 05 8A
25E0: A9 08 4C E7 25 A9 09 8D 2C
25E8: 2C D0 60 AD 10 D0 29 FB F9
25F0: 8D 10 D0 20 A1 22 C9 80 8D
25F8: 90 12 AD 10 D0 09 04 8D C4
2600: 10 D0 20 A1 22 C9 1E B0 5D
2608: F9 4C 13 26 20 A1 22 C9 35
2610: 20 90 F9 8D 04 D0 60 AD 2B
2618: 10 D0 29 DF 8D 10 D0 20 90
2620: A1 22 C9 80 90 12 AD 10 8E
2628: D0 09 20 8D 10 D0 20 A1 52
2630: 22 C9 1E B0 F9 4C 3F 26 96
2638: 20 A1 22 C9 20 90 F9 8D 1E
2640: 0A D0 60 20 A1 22 C9 20 49
2648: 90 F9 C9 C3 B0 F5 8D 06 9A
2650: D0 20 A1 22 C9 1B B0 F9 94
2658: 8D 08 D0 60 AD 05 D0 C9 6C
2660: BE 90 0B A9 08 8D 05 D0 CF
2668: 20 EB 25 4C 83 26 AD 82 BF
2670: 42 F0 07 AD 05 D0 C9 08 FF
2678: F0 09 AD 05 D0 18 69 08 7F
2680: 8D 05 D0 AD 07 D0 C9 BE F1
2688: 90 0E A9 08 8D 07 D0 8D CB
2690: 09 D0 20 43 26 4C B0 26 17
2698: AD 82 42 F0 07 AD 07 D0 88
26A0: C9 08 F0 0C AD 07 D0 18 0D
26A8: 69 08 8D 07 D0 8D 09 D0 E6
26B0: AD 0B D0 C9 BE 90 0E A9 0B
26B8: 08 8D 0B D0 20 17 26 20 A7
26C0: CE 25 4C DA 26 AD 82 42 74
26C8: F0 07 AD 0B D0 C9 08 F0 0D
26D0: 09 AD 0B D0 18 69 08 8D 7A
26D8: 0B D0 60 AD 11 D0 29 EF BD
26E0: 8D 11 D0 60 AD 11 D0 09 49
26E8: 10 8D 11 D0 60 A9 00 8D FF
26F0: 86 02 60 A9 B8 8D 01 D0 9B
26F8: A9 08 8D 05 D0 A9 50 8D 95
2700: 07 D0 8D 09 D0 A9 88 8D FE
2708: 0B D0 A9 96 8D 00 D0 A9 2C
2710: 83 8D F8 07 AD 10 D0 29 D8
2718: FE 8D 10 D0 60 78 A5 A3 A7
2720: 8D 14 03 A5 A4 8D 15 03 B4
2728: 58 60 A9 33 8D 70 42 A0 9E
2730: 00 A9 30 99 68 42 C8 C0 D7
2738: 04 D0 F8 A0 00 99 77 42 F9
2740: C8 C0 08 D0 F8 A9 31 8D 04
2748: 7B 42 A9 00 8D 71 42 8D 7E
2750: 74 42 8D 75 42 A9 04 8D 87
2758: 7F 42 A9 02 8D 72 42 AD B5
2760: 10 D0 29 F7 09 10 8D 10 19
2768: D0 A9 31 8D 76 42 60 A2 5D
2770: 0C A0 0B 20 D0 2B A0 00 E4
2778: 20 6F 2C 60 A2 0D A0 0D F1
2780: 20 D0 2B A0 12 20 6F 2C 0B
```

```
2788: 60 A2 0D A0 0C 20 D0 2B 61
2790: A0 20 20 6F 2C 60 A2 0B 1B
2798: A0 0F 20 D0 2B A0 2F 20 54
27A0: 6F 2C 60 A2 0D A0 0C 20 19
27A8: D0 2B A0 39 20 6F 2C 60 9A
27B0: 20 ED 26 A2 17 A0 07 20 66
27B8: D0 2B AD 70 42 20 D2 FF 08
27C0: A2 17 A0 12 20 D0 2B A0 E9
27C8: 00 B9 6C 42 20 D2 FF C8 EC
27D0: C0 04 D0 F5 A2 17 A0 22 D8
27D8: 20 D0 2B A0 00 B9 68 42 F9
27E0: 20 D2 FF C8 C0 04 D0 F5 28
27E8: A2 18 A0 06 20 D0 2B AD 14
27F0: 76 42 20 D2 FF A2 18 A0 F7
27F8: 10 20 D0 2B A0 00 B9 7B FA
2800: 42 20 D2 FF C8 C0 04 D0 93
2808: F5 A2 18 A0 22 20 D0 2B 97
2810: A0 00 B9 77 42 20 D2 FF 17
2818: C8 C0 04 D0 F5 60 20 BE AB
2820: 22 B0 FB 60 A9 FC A2 01 99
2828: 9D 00 D0 E8 E8 E0 0D D0 27
2830: F7 60 A9 00 8D 85 42 A9 31
2838: 13 20 D2 FF 20 35 2D EE AF
2840: 85 42 AD 85 42 C9 0F D0 27
2848: F3 60 20 1D 27 A9 83 8D BB
2850: F8 07 A9 25 20 48 2D 60 15
2858: 20 27 2D AD 80 42 F0 19 47
2860: AD 8D 42 F0 0B 20 E4 26 05
2868: 20 1F 2B A9 00 8D 8D 42 D9
2870: 20 96 27 20 A3 27 4C 7C 02
2878: 28 20 7C 27 20 51 2C 20 22
2880: 2A 27 20 2E 2D 60 20 27 F4
2888: 2D AD 70 42 C9 33 D0 06 E9
2890: 20 E4 26 20 9B 2B 20 89 4C
2898: 27 20 51 2C A9 00 8D 75 0A
28A0: 42 AD 76 42 18 69 01 C9 95
28A8: 37 90 02 A9 31 8D 76 42 93
28B0: A0 00 A9 30 99 77 42 C8 47
28B8: C0 04 D0 F8 AD 7B 42 18 CA
28C0: 69 01 C9 37 90 08 EE 7F 33
28C8: 42 EE 7F 42 A9 31 8D 7B 9F
28D0: 42 20 2E 2D 60 AD 6B 42 4A
28D8: C9 35 D0 35 A9 30 8D 6B B0
28E0: 42 EE 6A 42 AD 6A 42 C9 E2
28E8: 3A 90 2B A9 30 8D 6A 42 F2
28F0: EE 69 42 AD 69 42 C9 3A E8
28F8: 90 1C A9 30 8D 69 42 EE A7
2900: 68 42 AD 68 42 C9 3A 90 97
2908: 0D A9 30 8D 68 42 4C 16 89
2910: 29 A9 35 8D 6B 42 AD 68 69
2918: 42 CD 6C 42 90 2F F0 06 8D
2920: 20 4E 29 4C 4D 29 AD 69 91
2928: 42 CD 6D 42 90 1F F0 03 8B
2930: 4C 20 29 AD 6A 42 CD 6E 5C
2938: 42 90 12 F0 03 4C 20 29 A6
2940: AD 6B 42 CD 6F 42 90 05 B0
2948: F0 03 4C 20 29 60 A2 00 D4
2950: BD 68 42 9D 6C 42 E8 E0 CE
2958: 04 D0 F5 A9 01 8D 8D 42 2B
2960: 60 A2 00 BD 77 42 DD 7B 34
2968: 42 D0 0D E8 E0 04 D0 F3 1B
2970: 20 4A 28 A9 01 8D 81 42 FE
2978: AD 7B 42 38 E9 01 CD 77 4C
2980: 42 D0 13 AD 78 42 C9 39 12
2988: D0 0C AD 79 42 C9 30 D0 99
2990: 05 A9 01 8D 82 42 60 20 13
2998: F3 26 20 C1 25 20 CF 24 CD
29A0: 20 B0 27 20 03 25 AD 1E AC
29A8: D0 A9 00 8D 8A 42 20 E4 82
29B0: 26 60 A9 0C 8D 86 02 A9 AC
29B8: 13 20 D2 FF 20 FA 29 A2 A5
29C0: 00 20 EB 29 E8 EC 84 42 92
29C8: D0 F7 20 FA 29 20 16 2A 36
29D0: A2 00 20 F2 29 E8 E0 03 7C
29D8: D0 F8 A2 00 20 F6 29 E8 6E
29E0: E0 06 D0 F8 60 A9 40 20 FB
29E8: D2 FF 60 20 A1 23 20 D2 F3
29F0: FF 60 EE 83 42 60 CE 84 B9
29F8: 42 60 A2 00 20 E5 29 E8 56
2A00: EC 83 42 D0 F7 60 A9 00 85
2A08: 8D 82 42 A9 03 8D 83 42 5A
2A10: A9 22 8D 84 42 60 A2 00 33
2A18: BD 00 04 C9 00 D0 05 A9 23
2A20: 0E 4C 26 2A A9 0C 9D 00 1E
2A28: D8 E8 E0 C8 D0 EA 60 A2 52
2A30: 00 BD 40 2A C9 21 F0 07 3B
2A38: 9D 00 30 E8 4C 31 2A 60 F6
2A40: FF FF FF FF FF FF FF FF 40
2A48: 21 A0 18 A9 00 99 00 D4 3A
2A50: 88 10 FA A9 0F 8D 18 D4 17
2A58: A9 FF 8D 0E D4 60 A9 28 A4
2A60: 8D 05 D4 A9 44 8D 06 D4 1E
2A68: A9 0D 8D 01 D4 A9 01 8D BA
2A70: 00 D4 A9 11 8D 04 D4 60 C6
2A78: A9 82 8D 05 D4 A9 88 8D CB
2A80: 06 D4 A9 01 8D 01 D4 A9 13
2A88: 00 8D 00 D4 A9 81 8D 04 A7
2A90: D4 A9 09 20 48 2D A9 80 D7
2A98: 8D 04 D4 60 A2 05 BD 00 C4
2AA0: D0 C9 0A B0 05 A9 C0 4C B1
2AA8: AD 2A 38 E9 08 9D 00 D0 19
2AB0: E8 E8 E0 0D D0 E8 20 16 60
2AB8: 2A CE 79 42 CE 79 42 AD A5
2AC0: 79 42 C9 2E D0 20 A9 38 47
2AC8: 8D 79 42 CE 78 42 AD 78 C1
2AD0: 42 C9 2F D0 11 A9 39 8D 5E
2AD8: 78 42 CE 77 42 C9 2F D0 E5
2AE0: 05 A9 30 8D 77 42 A2 16 BF
2AE8: A0 00 20 D0 2B 20 35 2D 28
2AF0: 20 B0 27 60 A9 25 20 48 80
2AF8: 2D CE 70 42 AD 70 42 C9 D1
2B00: 30 D0 0B 20 B0 27 A9 01 AE
2B08: 8D 80 42 4C 0A 24 20 DB CE
2B10: 26 20 2E 2D 4C 0D 24 A2 D1
2B18: 17 A0 12 20 D0 2B 60 A9 08
2B20: 04 8D 8E 42 20 17 2B A9 8E
```

```
2B28: 20 A2 04 20 D2 FF CA 10 BC
2B30: FA A9 04 20 48 2D 20 17 A5
2B38: 2B A0 00 B9 6C 42 20 D2 5F
2B40: FF C8 C0 04 D0 F5 20 4F 04
2B48: 2B CE 8E 42 10 D6 60 A9 04
2B50: 82 8D 01 D4 A9 09 8D 05 7B
2B58: D4 A9 1E 8D 0F D4 A9 15 25
2B60: 8D 04 D4 A9 04 20 48 2D 0A
2B68: A9 14 8D 04 D4 A9 FF 8D C3
2B70: 0F D4 60 20 D5 28 EE 79 3B
2B78: 42 EE 79 42 AD 79 42 C9 98
2B80: 3A 90 17 A9 30 8D 79 42 85
2B88: EE 78 42 AD 78 42 C9 3A 9E
2B90: 90 08 A9 30 8D 78 42 EE 3A
2B98: 77 42 60 A2 08 A0 0C 20 2A
2BA0: D0 2B A0 48 20 6F 2C A9 EA
2BA8: 30 8D 8F 42 A9 09 8D 90 09
2BB0: 42 20 4F 2B A9 10 8D 04 D8
2BB8: D4 20 D5 28 20 B0 27 CE 72
2BC0: 90 42 10 ED EE 8F 42 AD FF
2BC8: 8F 42 CD 76 42 D0 DD 60 30
2BD0: 18 20 F0 FF 60 78 AD 14 94
2BD8: 03 85 A3 AD 15 03 85 A4 F4
2BE0: A9 FA 8D 14 03 A9 2B 8D 8C
2BE8: 15 03 58 60 A9 88 8D 05 7E
2BF0: D4 8D 06 D4 A9 00 8D 8B F0
2BF8: 42 60 AD 8B 42 C9 00 F0 D1
2C00: 26 C9 05 F0 1A C9 14 F0 CE
2C08: 1E C9 19 F0 12 C9 28 F0 EE
2C10: 26 C9 2E F0 0A C9 50 F0 34
2C18: 30 EE 8B 42 6C A3 00 A9 BE
2C20: 80 8D 04 D4 4C 19 2C A9 42
2C28: 01 8D 01 D4 8D 00 D4 A9 98
2C30: 81 8D 04 D4 4C 19 2C A9 53
2C38: 34 8D 01 D4 A9 01 8D 00 08
2C40: D4 A9 81 8D 04 D4 4C 19 0C
2C48: 2C A9 00 8D 8B 42 6C A3 89
2C50: 00 20 6F 27 20 E4 26 20 52
2C58: EC 2B 20 D5 2B 20 1E 28 F7
2C60: 20 1D 27 A9 80 8D 04 D4 55
2C68: 20 DB 26 20 32 28 60 20 85
2C70: ED 26 B9 CE 2C F0 07 20 51
2C78: D2 FF C8 4C 72 2C 60 20 7F
2C80: 20 4D 45 4E 3A 20 33 20 2F
2C88: 20 20 20 48 49 47 48 3A 44
2C90: 20 30 30 30 30 20 20 20 D1
2C98: 20 20 53 43 4F 52 45 3A 90
2CA0: 20 30 30 30 30 20 20 20 E1
2CA8: 4D 54 4E 3A 20 31 20 20 64
2CB0: 45 4C 45 56 2E 3A 20 30 96
2CB8: 30 30 30 20 20 43 55 52 74
2CC0: 52 2E 20 45 4C 45 56 3A C8
2CC8: 20 30 30 30 30 00 50 52 4C
2CD0: 45 53 53 20 46 49 52 45 04
2CD8: 20 42 55 54 54 4F 4E 00 D6
2CE0: 54 4F 20 53 54 41 52 54 34
2CE8: 20 47 41 4D 45 00 46 4F B9
2CF0: 52 20 4E 45 58 54 20 4C 10
2CF8: 45 56 45 4C 00 47 41 4D FB
2D00: 45 20 4F 56 45 52 00 46 E8
2D08: 4F 52 20 41 20 4E 45 57 16
2D10: 20 47 41 4D 45 00 42 4F DC
2D18: 4E 55 53 20 41 57 41 52 5B
2D20: 44 45 44 20 21 21 00 20 70
2D28: DB 26 20 24 28 60 20 06 1D
2D30: 2A 20 97 29 60 A9 0C 8D DE
2D38: 86 02 A0 00 20 A1 23 20 66
2D40: D2 FF C8 C0 28 D0 F5 60 EB
2D48: 8D 87 42 A2 1C A0 00 C8 C7
2D50: D0 FD CA D0 F8 CE 87 42 4C
2D58: D0 F1 60 00 7B
```

128 TO 64 AUTOBOOT
FROM PAGE 50

128-64.BAS

```
·10 CK=0                                                    HH
·20 READ A:IF A=256 THEN 40                                 FN
·30 CK=CK+A:GOTO 20                                         OF
·40 IF CK<>18681 THEN PRINT"ERROR IN DATA
 ":END                                                      CM
·50 OPEN 2,8,2,"0:128-64.BIN,P,W"                           NL
·60 PRINT#2,CHR$(0);CHR$(128);                              GK
·70 RESTORE                                                 IO
·80 READ A:IF A=256 THEN CLOSE2:END                         HM
·90 PRINT#2,CHR$(A);:GOTO80                                 KG
·32768 DATA 16,128,16,128,195,194,205,56                    CC
·32776 DATA 48,0,0,0,0,0,162,5                              AI
·32784 DATA 162,5,142,4,128,142,5,128                       JE
·32792 DATA 142,6,128,142,22,208,32,163                     AM
·32800 DATA 253,32,80,253,32,21,253,32                      NI
·32808 DATA 91,255,88,32,83,228,32,191                      EG
·32816 DATA 227,169,0,141,32,208,141,33                     BK
·32824 DATA 208,169,147,32,210,255,169,11                   NP
·32832 DATA 141,137,2,76,80,128,85,48                       JF
·32840 DATA 62,77,48,0,48,58,42,0                           HK
·32848 DATA 169,15,162,8,160,15,32,186                      FC
·32856 DATA 255,160,0,185,70,128,240,7                      BE
·32864 DATA 153,0,128,200,76,91,128,169                     LI
·32872 DATA 5,162,0,160,128,32,189,255                      OP
·32880 DATA 32,192,255,169,15,32,195,255                    OO
·32888 DATA 169,2,162,8,160,1,32,186                        HA
·32896 DATA 255,160,0,185,76,128,240,7                      NK
·32904 DATA 153,0,128,200,76,131,128,169                    MO
·32912 DATA 3,162,0,160,128,32,189,255                      AF
·32920 DATA 169,0,32,213,255,169,2,32                       LN
·32928 DATA 195,255,162,251,154,76,139,22
 7,0,256                                                    KB
```

1-0.BAS

```
·10 OPEN15,8,15                                          AM
·20 OPEN5,8,5,"#"                                        DA
·30 PRINT#15,"B-P:"5;0                                   OH
·40 PRINT#5,CHR$(67);CHR$(66);CHR$(77); :
  REM  * CBM CODE *                                      EO
·50 PRINT#5,CHR$(0);CHR$(0);CHR$(0);CHR$(
0);                                                      GC
·60 PRINT"[CLEAR]ENTER BOOT MESSAGE":GOSU
B1000                                                    CO
·70 PRINT#5,CHR$(0);                                     HN
·80 PRINT#5,"128-64.BIN";                                MG
·90 PRINT#5,CHR$(0);CHR$(76);CHR$(75);CHR
$(226);CHR$(0);                                          DO
·100 PRINT#15,"U2:"5;0;1;0                               KF
·110 CLOSE5:CLOSE15                                      BF
·120 END                                                 IC
·1000 PRINT"  USE '[BACKARROW]' WHEN DONE
."                                                       MN
·1005 R$="":POKE212,1                                    DE
·1020 GETG$:IFG$=""THEN1020                              NB
·1030 IFG$=CHR$(95)THEN1050                              ML
·1040 R$=R$+G$:POKE216,1:PRINTG$;:GOTO102
0                                                        CF
·1050 POKE212,0:PRINT#5,R$;                              HE
·1060 RETURN                                             IM
```

VAULT OF TERROR
FROM PAGE 58

```
·1 REM VAULT V2.6 BY C.M. BLAKEMORE                      HE
·2 POKE53280,.:POKE53281,.:POKE808,234:GO
TO123                                                    NB
·3 VL=FC:ONFC+1GOSUB49,49,58,60,61                       JL
·4 RETURN                                                IM
·5 GOSUB3:ONFC+1GOSUB72,62,67,70,71:RETUR
N                                                        HJ
·6 ONFC+1GOSUB74,75,76,77:RETURN                         LB
·7 ONFC+1GOSUB78,79,80,81:RETURN                         ML
·8 LP=FC:ONFC+1GOSUB82,86,88,90:RETURN                   AA
·9 ONFC+1GOSUB91,93,95,96:RETURN                         IA
·10 PRINT"[CLEAR][DOWN][RVSOFF] [s M][s O
][34"[c Y]"][s P][s N]"                                  GN
·11 PRINT" [s M][c H]"TAB(37)"[c N][s N]"                GF
·12 PRINT" [s M][c H]"TAB(37)"[c N][s N]"                GF
·13 PRINT" [s M][c G][3" "][28"[c @]"][3"
 "][c M][s N]                                            EN
·14 PRINT" [s M][c G]  [c M][c *][RVSON][
26" "][RVSOFF][sEP][c G]  [c M][s N]                     PO
·15 PRINT" [s M][c G][SS] [c M] [c *][RVS
ON][24" "][RVSOFF][sEP] [c G] [SS][c M][
s N]                                                     NF
·16 PRINT" [s M][c G]  [c M] [c M][s M][2
2"[c T]"][s N][c G] [c G]  [c M][s N]                    LH
·17 PRINT" [s M][c G]  [c M] [c M][s M] [
s O][18"[c Y]"][s P][SS][s N][c G] [c G]
  [c M][s N]                                             EA
·18 PRINT"  [c G]  [c M] [c M][s M] [c G]
[c M][c *][RVSON][14" "][RVSOFF][sEP][c
G][c M] [s N][c G] [c G] [SS][c M]                       HI
·19 PRINT"  [c G]  [c M] [c M][s M] [c G]
[c M][c M][s M][12" "][s N][c G][c G][c
M] [s N][c G] [c G]  [c M]                               HK
·20 PRINT"  [s O][c Y][c Y][s P] [c M][s
M] [c G][c M][c M][s M][s O][10"[c T]"][
s P][s N][c G][c G][c M] [s N][c G] [s O
][c Y][c Y][s P]                                         FE
·21 PRINT"  [c G][RVSON][s U][s I][RVSOFF
][c M] [c M][SS] [RVSON][s U][s I][RVSOF
F][c M][s M][c G][RVSON][s T][8"[c I]"][
s Y][RVSOFF][c M][s N][c G][RVSON][s U][
s I][RVSOFF] [SS][c G] [c G][RVSON][s U]
[s I][RVSOFF][c M]                                       PA
·22 PRINT"  [c G][RVSON][s J][s K][RVSOFF
][c M] [c M]  [RVSON][s J][s K][RVSOFF][
c M] [RVSON][s W][s T][RVSOFF][s M][RVSO
N].[4"[s C]"].[RVSOFF][s N][RVSON][s Y][
s W][RVSOFF] [c G][RVSON][s J][s K][RVSO
FF]  [c G] [c G][RVSON][s J][s K][RVSOFF
][c M]                                                   EH
·23 PRINT"  [s L][c P][c P][s @] [c M][SS
] [c G][c M][c M][s N][c G][RVSON][s T][
RVSOFF][s N][RVSON][s G][RVSOFF][4" "][R
VSON][s H][RVSOFF][s M][RVSON][s Y][RVSO
FF][c M][s M][c G][c G][c M] [SS][c G] [
s L][c P][c P][s @]"                                     GL
·24 PRINT" [s N][c G][SS] [c M] [c M][s N
] [c G][c M][c M][s N][c G][RVSON][s T][
RVSOFF][s N][6"[c T]"][s M][RVSON][s Y][
RVSOFF][c M][s M][c G][c G][c M] [s M][c
 G][SS][c G] [SS][c M][s M]                              IE
·25 PRINT" [s N][c G]  [c M][SS][c M][s N
] [c G][c M][c M][s N][s L][sEP][8"[c T]
"][c *][s @][s M][c G][c G][c M] [s M][c
 G] [c G]  [c M][s M]                                    BK
·26 PRINT" [s N][c G][SS][SS][c M][SS][c
M][s N] [c G][c M][c M][s N][12"[c T]"][
s M][c G][c G][c M] [s M][c G] [c G]  [c
 M][s M]                                                 JE
·27 PRINT" [s N][c G] [SS][c M][SS][c M][
s N] [s L][s @][s N][14" "][s M][s L][s
@] [s M][c G] [c G] [SS][c M][s M]                       KC
·28 PRINT" [s N][c G][SS][SS][c M] [c M][
s N][c T][SS] [16"[c T]"]  [c T][s M][c
G] [c G]  [c M][s M]                                     EH
·29 PRINT" [s N][c G][SS][SS][c M][SS][c
M][24" "][c G] [c G]  [c M][s M]                         EK
·30 PRINT" [s N][c G][SS][SS][c M] [s N]"
```

```
TAB(32)"[s M][SS][c G][SS] [c M][s M]"  EK
·31 PRINT" [s N][s L][c P][c P][s @][s N]
 "TAB(33)"[s M][s L][c P][c P][s @][s M]
"                                       MG
·32 PRINT"[39"[c Y]"]"                  BI
·33 PRINT"[RVSON][RED][3" "]";D$(CD);"  [
SS][5" "]#[SS]$[SS]%[SS]&[SS]'[SS]([SS])
[SS]* +[7" "]";D$(CD);"  [RVSOFF]";      MO
·34 PRINTL$(DD);:RETURN                 OH
·35 PRINT"[HOME][DOWN][RVSON][c Y][s P][D
OWN][LEFT][LEFT]";:FORX=.TO19:PRINT" [c
M][DOWN][LEFT][LEFT]";:NEXT:PRINT"[c P][
s @][DOWN][LEFT][LEFT] [RVSOFF][sEP]":RE
TURN                                    AP
·36 PRINT"[HOME][8"[DOWN]"]";TAB(8)"[RVSO
N][c Y][s P][DOWN][LEFT][LEFT]";:FORX=.T
O8:PRINT" [c M][DOWN][LEFT][LEFT]";:NEXT
:PRINT"[c P][s @][DOWN][LEFT][LEFT][RVSO
FF][sEP]"                               JA
·37 RETURN                              IM
·38 PRINT"[HOME][11"[DOWN]"]";TAB(13)"[RV
SON][s P][DOWN][LEFT]";:FORX=.TO3:PRINT"
[c M][DOWN][LEFT]";:NEXT:PRINT"[s @][DOW
N][LEFT][RVSOFF][sEP]"                  NM
·39 RETURN                              IM
·40 PRINT"[HOME][13"[DOWN]"]";TAB(16)"[RV
SON][s P][DOWN][LEFT][s @][DOWN][LEFT][R
VSOFF][sEP]";                           AA
·41 RETURN                              IM
·42 PRINT"[HOME][DOWN][RVSON]";TAB(38)"[s
 O][DOWN][LEFT]";:FORX=.TO19:PRINT"[c G]
[DOWN][LEFT]";:NEXT:PRINT"[s L][DOWN][LE
FT][RVSOFF][c *]":RETURN                JB
·43 PRINT"[HOME][8"[DOWN]"]"TAB(30)"[RVSO
N][s O][c Y][DOWN][LEFT][LEFT]";:FORX=.T
O8:PRINT"[c G] [DOWN][LEFT][LEFT]";:NEXT
:PRINT"[s L][c P][DOWN][LEFT][RVSOFF][c
*]                                      EF
·44 RETURN                              IM
·45 PRINT"[HOME][11"[DOWN]"]";TAB(26)"[RV
SON][s O][DOWN][LEFT]";:FORX=.TO3:PRINT"
[c G][DOWN][LEFT]";:NEXT:PRINT"[s L][DOW
N][LEFT][RVSOFF][c *]"                  IN
·46 RETURN                              IM
·47 PRINT"[HOME][13"[DOWN]"]";TAB(23)"[RV
SON][s O][DOWN][LEFT][s L][DOWN][LEFT][R
VSOFF][c *]"                            AO
·48 RETURN                              IM
·49 PRINT"[HOME][5"[DOWN]"]";:FORX=.TO3:P
RINTTAB(6);                             FH
·50 PRINT"[s O][c Y][s P][s O][c Y][s P][
s O][c Y][s P][s O][c Y][s P][s O][c Y][
s P][s O][c Y][s P][s O][c Y][s P][s O][
c Y][s P][s O][c Y][s P][s O]"          PN
·51 PRINTTAB(6)"[s L][c P][s @][s L][c P]
[s @][s L][c P][s @][s L][c P][s @][s L]
[c P][s @][s L][c P][s @][s L][c P][s @]
[s L][c P][s @][s L][c P][s @][s L]"    PM
·52 PRINTTAB(6)"[s P][s O][c Y][s P][s O]
 [c Y][s P][s O][c Y][s P][s O][c Y][s P]
 [s O][c Y][s P][s O][c Y][s P][s O][c Y]
 [s P][s O][c Y][s P][s O][c Y][s P]"   NN
·53 PRINTTAB(6)"[s @][s L][c P][s @][s L]
 [c P][s @][s L][c P][s @][s L][c P][s @]
 [s L][c P][s @][s L][c P][s @][s L][c P]
 [s @][s L][c P][s @][s L][c P][s @]"   AA
·54 NEXT                                IA
·55 PRINTTAB(6)"[s O][c Y][s P][s O][c Y]
 [s P][s O][c Y][s P][s O][c Y][s P][s O]
 [c Y][s P][s O][c Y][s P][s O][c Y][s P]
 [s O][c Y][s P][s O][c Y][s P][s O]"   MG
·56 PRINTTAB(6)"[s L][c P][s @][s L][c P]
 [s @][s L][c P][s @][s L][c P][s @][s L]
 [c P][s @][s L][c P][s @][s L][c P][s @]
 [s L][c P][s @][s L][c P][s @][s L]"   PM
·57 RETURN                              IM
·58 PRINT"[HOME][9"[DOWN]"]";:FORX=.TO4:P
 RINTTAB(12);"[s P][c Y][s P][c Y][s P][c
  Y][s P][c Y][s P][c Y][s P][c Y][s P][c
  Y][s P][c Y]"                         BM
·59 PRINTTAB(12);"[c Y][s P][c Y][s P][c
 Y][s P][c Y][s P][c Y][s P][c Y][s P][c
 Y][s P][c Y][s P]":NEXT:RETURN         AI
·60 PRINT"[HOME][12"[DOWN]"]";:FORX=.TO4:
 PRINTTAB(15);"[10"[cEP]"]":NEXT:RETURN AC
·61 PRINT"[HOME][14"[DOWN]"]";:PRINTTAB(1
 8)"[4"[c +]"]":RETURN                  JC
·62 PRINT"[HOME][10"[DOWN]"]";TAB(15)"[RV
 SON][10":"]":PRINTTAB(15)"[RVSON]:[RVSOF
 F][8"[c Z]"][RVSON]:"                  MO
·63 PRINTTAB(15)"[RVSON]:[RVSOFF][c Z][s
 O][4"[c Y]"][s P][c Z][RVSON]:":PRINTTAB
 (15)"[RVSON]:[RVSOFF][c Z][c G][4"[s C]"
 ][c M][c Z][RVSON]:"                   NP
·64 PRINTTAB(15)"[RVSON]:[RVSOFF][c Z][s
 L][4"[c P]"][s @][c Z][RVSON]:":PRINTTAB
 (15)"[RVSON]:[RVSOFF][7"[c Z]"][RVSON][s
  B]:"                                  KJ
·65 PRINTTAB(15)"[RVSON]:[RVSOFF][7"[c Z]
 "][s S][RVSON]:"                       NM
·66 FORX=.TO5:PRINTTAB(15)"[RVSON]:[RVSOF
 F][8"[c Z]"][RVSON]:":NEXT:RETURN      OK
·67 PRINT"[HOME][11"[DOWN]"]";:PRINTTAB(1
 7)"[RVSON][6":"]":PRINTTAB(17)"[RVSON]:[
 RVSOFF][4"[c Z]"][RVSON]:"             FK
·68 PRINTTAB(17)"[RVSON]:[RVSOFF][c Z][s
 C][s C][c Z][RVSON]:":PRINTTAB(17)"[RVSO
 N]:[RVSOFF][3"[c Z]"][s Q][RVSON]:"    OG
·69 FORX=.TO3:PRINTTAB(17)"[RVSON]:[RVSOF
 F][4"[c Z]"][RVSON]:":NEXT:RETURN      OO
·70 PRINT"[HOME][13"[DOWN]"]";TAB(19)"[c
 D][c F]":FORX=.TO2:PRINTTAB(19)"[c Z][c
 Z]":NEXT:RETURN                        JM
·71 PRINT"[HOME][14"[DOWN]"]";TAB(19)"[c
 -][RVSON][c -]":RETURN                 PN
·72 PRINT"[HOME]":FORX=.TO21:PRINT"[s V][
```

```
s V]"TAB(38)"[s +]":NEXT                                        LJ
·73 PRINT"[s V][s N][RVSON][36"[c +]"][RV
SOFF][s M]";:GOSUB62:RETURN                                     GO
·74 PRINT"[HOME][23"[DOWN]"]";TAB(12)"[RV
SON][sEP][s O][12"[c Y]"][s P][c *]";:RE
TURN                                                            NA
·75 PRINT"[HOME][20"[DOWN]"]";TAB(14)"[RV
SON][sEP][s O][8"[c Y]"][s P][c *]";:RET
URN                                                             JF
·76 PRINT"[HOME][17"[DOWN]"]";TAB(16)"[RV
SON][sEP][s O][4"[c Y]"][s P][c *]";:RET
URN                                                             DH
·77 PRINT"[HOME][15"[DOWN]"]";TAB(18)"[RV
SON][sEP][s O][s P][c *]";:RETURN                               FK
·78 PRINT"[HOME]"TAB(12)"[c *][RVSON][s L
][12"[c P]"][s @][RVSOFF][sEP]":RETURN                          JL
·79 PRINT"[HOME][7"[DOWN]"]"TAB(14)"[c *]
[RVSON][s L][8"[c P]"][s @][RVSOFF][sEP]
":RETURN                                                        HH
·80 PRINT"[HOME][10"[DOWN]"]";TAB(16)"[c
*][RVSON][s L][4"[c P]"][s @][RVSOFF][sE
P]":RETURN                                                      LM
·81 PRINT"[HOME][12"[DOWN]"]";TAB(18)"[c
*][RVSON][s L][s @][RVSOFF][sEP]":RETURN                        HB
·82 PRINT"[HOME]";:PRINTTAB(14)"[RVSON][c
 M]"TAB(25)"[RVSON][c G]"                                       IK
·83 FORX=.TO6:PRINTTAB(14)"[RVSON][c M][1
0"[s E]"][c G]":PRINTTAB(14)"[RVSON][c M
]"TAB(25)"[RVSON][c G]"                                         EN
·84 PRINTTAB(14)"[RVSON][c M]"TAB(25)"[RV
SON][c G]":NEXT                                                 NA
·85 PRINTTAB(14)"[RVSON][c M][10"[s E]"][
c G]":PRINTTAB(14)"[RVSON][c M]"TAB(25)"
[RVSON][c G][HOME]":RETURN                                      FO
·86 PRINT"[HOME][7"[DOWN]"]";:FORX=.TO6                         GI
·87 PRINTTAB(16)"[RVSON][s Y][6"[s E]"][s
 T]":PRINTTAB(16)"[RVSON][s Y]"TAB(23)"[
RVSON][s T]":NEXT:RETURN                                        OM
·88 PRINT"[HOME][10"[DOWN]"]";:FORX=.TO7                        JA
·89 PRINTTAB(18)"[RVSON][c D][c I][c I][c
 F]":NEXT:RETURN                                                IL
·90 PRINT"[HOME][13"[DOWN]"]";:FORX=.TO2:
PRINTTAB(19)"[c Q][c W]":NEXT:RETURN                            LP
·91 PRINT"[HOME][22"[DOWN]"]";TAB(11)"[RV
SON][sEP][s N][14"-"][s M][c *]"                                OD
·92 PRINTTAB(10)"[RVSON][sEP][s N][16"-"]
[s M][c *][HOME]":RETURN                                        JN
·93 PRINT"[HOME][18"[DOWN]"]";TAB(14)"[RV
SON][sEP][s N][8"="][s M][c *]":PRINTTAB
(14)"[RVSON][s O][10"[c T]"][s P]"                              OA
·94 PRINTTAB(14)"[RVSON][s O][c T][c R][6
"[c T]"][c R][c T][s P]":RETURN                                 BM
·95 PRINT"[HOME][16"[DOWN]"]"TAB(16)"[RVS
ON][sEP][s N][4"="][s M][c *]":PRINTTAB(
16)"[RVSON][s O][c R][4"[c T]"][c R][s P
]":RETURN                                                       LO
·96 PRINT"[HOME][15"[DOWN]"]"TAB(17)"[RVS
ON][sEP][4"="][c *]":PRINTTAB(17)"[RVSON
 ][s O][c R][c T][c T][c R][s P]":RETURN                        FB
·97 CEM=FNTE(.):POKE648,CEM:PRINTL$(DD);:
GOSUB10                                                         OM
·98 FC=4:VL=FC:LP=FC:LV=CD-1+4*-(CD=1):RV
=CD+1+4*(CD=4)                                                  ED
·99 EBM=FNCP(.):IFEBM>-1ANDEBM<256THEN102                       HG
·100 IFFCTHENFC=FC-1:GOTO99                                     KL
·101 RETURN                                                     IM
·102 EL=FNCP(LV):ER=FNCP(RV)                                    PA
·103 IF(DL(EBM,.))THEN105                                       DH
·104 GOSUB3:FC=FC-1:ON1-(FC<.)GOTO99,113                        BD
·105 IFDL(EL,.)ANDFC<4THENONFC+1GOSUB35,3
6,38,40                                                         LA
·106 IFDL(ER,.)ANDFC<4THENONFC+1GOSUB42,4
3,45,47                                                         DD
·107 FORD=1TO5:IF(DL(EBM,.)ANDBO(D))THENO
NDGOSUB9,6,7,8,5                                                AN
·108 PRINT"[RVSOFF]";:NEXT:FC=FC-1:IFFC>-
1THEN99                                                         AK
·109 IFDL(CL,2)=.THEN113                                        JE
·110 FORX=.TO8:POKEXR,22:POKEYR,.:POKEPR,
.:SYSPLOT                                                       BB
·111 IF(DL(CL,2)ANDBO(X))THENPRINTTAB(X*4
+3);OA$(X)                                                      JI
·112 NEXT                                                       IA
·113 GOSUB116                                                   CM
·114 CSM=FNSW(.):POKE53272,(PEEK(53272)AN
D15)ORCSM:POKEFNPM(U),7:RETURN                                  OB
·115 :                                                          DI
·116 IFFCANDVLTHENVL=VL-1                                       OI
·117 D=61:POKEFNMP(1),D:POKEFNMP(2),D:POK
EFNMP(3),D:POKEFNMP(5),D                                        KF
·118 EL=.:ER=.:FORX=VLTO.STEP-1:FC=FNCM(X
):IFDL(FC,1)=.THEN122                                           CD
·119 FC=DL(FC,1)-1:EL=(ELORSP(X+4)):IFX>L
PTHENER=(ERORSP(X+4))                                           HI
·120 MB=SP(X)-1:FORD=1TOLEN(SB$(X,FC)):PO
KEFNMP(MB+D),ASC(MID$(SB$(X,FC),D,1))                           DG
·121 POKEBC+SP(X)+D,CS(FC):NEXT                                 AL
·122 NEXT:POKEV+21,EL:POKEV+27,ER:RETURN                        JA
·123 POKE56578,PEEK(56578)OR3:POKE56576,(
PEEK(56576)AND252)                                              NN
·124 POKE53272,PEEK(53272)AND15:POKE648,1
92:POKE53272,(PEEK(53272)AND240)OR12                            JC
·125 POKE53270,PEEK(53270)AND247                                FP
·126 PRINTCHR$(147):GOSUB234:GOSUB238                           EP
·127 DIMDL(255,2),BO(15),SP(7),SB$(3,7),D
$(4),L$(6),OA$(8),CS(7),M(15),MH(15)                            FJ
·128 DIMHC(2):X=.:D=.:FC=.:CL=17:CD=2:VL=
.:EL=.:ER=.:VS=.                                                OC
·129 D(1)=1:D(2)=16:D(3)=-1:D(4)=-16:CS(.
)=9:CS(1)=2:CS(2)=6:CS(3)=9:CS(4)=4                             II
·130 CS(5)=14:CS(6)=13:CS(7)=1:LP=.:P=.:L
=1:LF=10                                                        GG
·131 SP(.)=.:SP(1)=4:SP(2)=6:SP(3)=7:SP(4
)=15:SP(5)=48:SP(6)=64:SP(7)=128                                BP
·132 TGL=16:CSM=.:CEM=192:V=53248                               FL
·133 D$(1)="E":D$(2)="S":D$(3)="W":D$(4)=
```

```
"N"                                                OC
·134 NJ=255:FJ=254:LJ=251:RJ=247:BJ=253:U
S=239:MU=237:U=.                                   KG
·135 TL=388:DD=6:PLOT=65520:AR=780:XR=781
:YR=782:PR=783:S=54272:BC=V+38                     JB
·136 RU=156:CU=142:ALL=511:HX=157:WL=17:C
C=204:CR$=CHR$(13)                                 KE
·137 DEFFNCP(X)=FC*D(CD)+(CL+D(X)):DEFFNS
W(X)=(TGL-CSM)                                     JA
·138 DEFFNTE(X)=(TL-CEM):DEFFNCM(X)=X*D(C
D)+CL:DEFFNMP(X)=CEM*256+1016+X                    BM
·139 DEFFNPM(X)=55296+972+(X*2)                    EM
·140 FORX=.TO15:BO(X)=2[UPARROW]X:NEXT:L$
(.)="[WHITE]":L$(1)="[c 8]":L$(2)="[c 5]
":L$(3)="[c 4]"                                    PM
·141 L$(4)="[c 4]":L$(5)="[BLUE]":L$(6)="
[BLACK]":HC(.)=2:HC(1)=8:HC(2)=10                  LJ
·142 FORX=.TO239:READD:DL(X,.)=D:NEXT              FE
·143 FORX=.TO8:READA$:OA$(X)=A$:NEXTX              JK
·144 FORX=.TO15:READA:M(X)=A:NEXTX                 DF
·145 FORX=3TO.STEP-1:FORD=.TO7:READA$:SB$
(X,D)=A$:NEXTD,X                                   GC
·146 FORX=.TO8                                     KJ
·147 A=INT(RND(1)*93+34):IFDL(A,.)=.THEN1
47                                                 AF
·148 DL(A,1)=M(X):M(X)=A:MH(X)=X*4+1:NEXT          DJ
·149 FORX=9TO15                                    NN
·150 A=INT(RND(1)*96+142):IFDL(A,.)=.THEN
150                                                KH
·151 DL(A,1)=M(X):M(X)=A:MH(X)=X*4+1:NEXT          DJ
·152 D=52288:FORX=.TO767:READA:POKED+X,A:
NEXT:FORX=.TO63:POKE53055+X,.:NEXT                 LH
·153 PRINT"[c 4]":FORX=.TO8:POKEXR,15:POK
EYR,.:POKEPR,.:SYSPLOT                             PB
·154 PRINTTAB(X*4+2);OA$(X)                        BE
·155 NEXT                                          IA
·156 PRINT"[7"[DOWN]"][RED][RVSON] F1/ ST
ART NEW GAME  F2/ LOAD OLD GAME";:POKE19
8,.                                                DL
·157 GETA$:IFA$<>CHR$(133)ANDA$<>CHR$(137
)THEN157                                           HK
·158 IFA$=CHR$(137)THENGOSUB267                    CN
·159 DL(CC,1)=BO(8)                                IK
·160 POKEV+21,.:POKEV+23,127:POKEV+29,15:
POKEV,150:POKEV+1,193:POKEV+2,151                  CH
·161 POKEV+3,151:POKEV+4,119:POKEV+5,109:
POKEV+6,167:POKEV+7,109:POKEV+8,164                PE
·162 POKEV+9,174:POKEV+10,164:POKEV+11,13
3:POKEV+12,168:POKEV+13,152                        IK
·163 POKEV+14,172:POKEV+15,160                     PH
·164 GOSUB322:GOSUB97:IT=TI+1000                   JF
·165 JOY=PEEK(56321)                               LF
·166 IFRND(1)>.93THENPOKES+1,RND(1)*7+2            FM
·167 B=DL(CL,1):IFBANDRND(1)+B/100>.95THE
NGOSUB200                                          AN
·168 IFTI>ITTHENGOSUB247:GOTO181                   KN
·169 IFJOY=NJTHEN165                               CF
·170 IF(DL(CL+D(CD),.)ANDBO(5+CD))THEN172          HA
·171 IFJOY=FJANDDL(CL+D(CD),.)THENCL=CL+D
(CD):GOTO180                                       DC
·172 IF(DL(CL-D(CD),.)ANDBO(5))THEN174             HN
·173 IFJOY=BJANDDL(CL-D(CD),.)THENCL=CL-D
(CD):GOTO180                                       NO
·174 IFJOY=RJTHENCD=CD+1:GOTO179                   LI
·175 IFJOY=LJTHENCD=CD-1:GOTO179                   IP
·176 IFJOY=USTHENONU+1GOSUB210,211,4,219,
221,226,229,4,4:GOTO165                            PB
·177 IFJOY=MUTHENGOSUB185:GOSUB322:GOTO16
5                                                  JA
·178 GOSUB319:GOSUB322:GOTO165                     PL
·179 CD=CD+((CD=5)*4)+((CD=.)*-4)                  DH
·180 GOSUB97:GOTO165                               FP
·181 VS=VS+1:IFVS=DD*6THENVS=.:DD=DD+((DD
<6)*-1):IFDD=6THENL=(LAND254)                      EO
·182 LF=LF+((LF<60)*-.2):IF(LAND1)=.THEND
D=6                                                HN
·183 POKEV+32,(LF<5)*-2+(LF>=5ANDLF<10)*-
7                                                  CN
·184 GOTO165                                       CK
·185 POKE198,.                                     JP
·186 IFCU=14THENCU=142:PRINTCHR$(9);CHR$(
CU);CHR$(8);:GOSUB280:POKEV+33,.                   PE
·187 FORX=.TO8:IF(LANDBO(X))THENPOKEFNPM(
X),10                                              AJ
·188 NEXT:X=.:A=FRE(.):A=.:IF(LAND(BO(U))
)THENPOKEFNPM(U),7                                 GC
·189 IFPEEK(56321)<>NJTHEN189                      NM
·190 POKEFNPM(X),1:IF(LANDBO(A))THENPOKEF
NPM(A),10:GOTO192                                  IM
·191 POKEFNPM(A),2                                 NP
·192 JY=PEEK(56321):GETA$:IFA$=CHR$(140)T
HEN259                                             EF
·193 P=L:IFJY=RJTHENA=X:X=X+1+(X=8)*9              HA
·194 IFJY=USAND(LANDBO(X))THENPOKEFNPM(U)
,10:U=X:POKEFNPM(X),7:GOSUB319                     FC
·195 IFJY=USTHENRETURN                             MK
·196 IFJY=FJAND(DL(CL,2)ANDBO(X))THENL=(L
ORBO(X)):DL(CL,2)=DL(CL,2)-BO(X)                   HP
·197 IFJY=BJAND(LANDBO(X))THENL=(LAND(511
-BO(X))):DL(CL,2)=DL(CL,2)ORBO(X):U=.              OM
·198 IFL<>PTHENGOSUB319:GOSUB97:GOTO186            FC
·199 GOTO190                                       CK
·200 IFB=6THENU=.:DL(HX,2)=DL(HX,2)OR(LAN
D238):L=LAND273                                    HL
·201 IFCU=14THENRETURN                             HI
·202 D=INT(RND(1)*3+1):POKEV+33,HC(D-1):L
F=LF-(B/((LAND4)+1))+-D                            GI
·203 FORT=1TO100:NEXT:POKEV+33,.:IFLF<1TH
EN205                                              JA
·204 POKEV+32,(LF<5)*-2+(LF>=5ANDLF<10)*-
7:GOSUB324:GOSUB322:RETURN                         PJ
·205 POKEV+33,.:POKEV+32,.:GOSUB277:PRINT
"[CLEAR][RVSON][c 4]"TAB(9)"THOU ART SLA
IN !                                               AC
·206 PRINT"[RVSON][WHITE][DOWN][DOWN] DO
YOU WISH TO ENTER THE VAULT AGAIN?":POKE
```

```
198,.                                                    IA
·207 GETA$:IFA$=""THEN207                                IG
·208 IFA$="N"THENPOKE648,4:POKE679,.:SYS6
79                                                       AP
·209 POKEV+21,.:RUN                                      BO
·210 IFDD=6AND(LAND1)THENVS=.:DD=1:GOSUB9
7:RETURN                                                 JE
·211 IFB=.ORRND(.)+LF/100<.5THENGOSUB324:
RETURN                                                   KI
·212 POKEV+33,5:FORT=1TO100:NEXT:POKEV+33
,.:FORX=.TO15                                            KL
·213 ON1-(M(X)=.)GOTO214,217                             KJ
·214 IFCL=M(X)THENMH(X)=MH(X)-ABS(RND(.)+
((LAND2)*1)+LF/5)                                        PM
·215 IFMH(X)>.THEN217                                    NM
·216 M(X)=.:DL(CL,1)=.:POKEV+21,PEEK(V+21
)AND240:GOSUB277:GOSUB218                                IL
·217 NEXT:GOSUB324:GOSUB322:RETURN                       CA
·218 LF=LF+B/4:DL(CL,2)=(DL(CL,2)ORBO(B-1
)):GOSUB97:GOSUB319:RETURN                               PO
·219 CU=RU-CU:PRINTCHR$(9);CHR$(CU);CHR$(
8);:POKEV+33,(CU=14)*-4                                  LM
·220 GOSUB280:GOSUB322:RETURN                            EI
·221 IF(DL(CL,.)ANDBO(2))THENCL=CL+128:GO
SUB315:GOSUB97:GOSUB322:RETURN                           FJ
·222 IF(DL(CL,.)AND24)=24AND(LAND384)=384
THENDL(CL,2)=DL(CL,2)ORBO(8):L=LAND255                   FO
·223 IF(DL(CL,.)AND24)=24THENCL=CL-128:GO
SUB312:GOSUB97:GOSUB322:RETURN                           GL
·224 IFCL=WLANDL=ALLTHEN232                              GO
·225 GOSUB319:GOSUB322:RETURN                            GL
·226 IFB=6THENGOTO212                                    HK
·227 DL(CL+D(CD),.)=(DL(CL+D(CD),.)AND63)
:GOSUB280:GOSUB322                                       HF
·228 RETURN                                              IM
·229 IF(DL(CL+D(CD),.)ANDBO(5))=.THENRETU
RN                                                       DG
·230 IF(DL(CL+D(CD),.)ANDBO(5+CD))THENDL(
CL+D(CD),.)=DL(CL+D(CD),.)-BO(5+CD)                      ML
·231 GOSUB324:GOSUB322:RETURN                            FN
·232 PRINT"[CLEAR]":PRINT"[DOWN][DOWN][RV
SON][WHITE] THE [RED]CROWN [CYAN]OF [PUR
PLE]SARNOTH [RED]HAS [YELLOW]BEEN [WHITE
]RETURNED!"                                              FB
·233 PRINT"[RVSON][c 7] NOBLE WARRIOR,YOU
 HAVE SURVIVED VAULT ":GOSUB280:GOTO206                  AP
·234 FORX=1TO33:READA:POKE680+X,A:NEXT:PO
KE56334,PEEK(56334)AND254                                CE
·235 POKE1,PEEK(1)AND251:SYS681:POKE1,PEE
K(1)OR4:POKE56334,PEEK(56334)OR1                         JE
·236 READA:IFA=-1THENRETURN                              KN
·237 FORX=.TO7:READD:POKE61440+A*8+X,D:NE
XT:GOTO236                                               DH
·238 PRINTCHR$(142)CHR$(8)CHR$(31)CHR$(19
);                                                       FD
·239 FORH=1TO36:PRINT"[cEP]";:NEXT:FORV=1
TO23:PRINTTAB(1)"[cEP]"TAB(37)"[cEP]"                    PO
·240 NEXT:FORH=1TO38:PRINT"[cEP]";:NEXT                  FJ
·241 PRINT"[HOME][DOWN][DOWN]"TAB(2)"[PUR
PLE][3" "]PREPARE THYSELF,BRAVE WARRIOR[
3"."][DOWN]"                                             MN
·242 PRINTTAB(2)"[RVSON][c 4] [RVSOFF]$[R
VSON]  V A U L T  O F  T E R R O R  [RVS
OFF]$[RVSON] "                                           OJ
·243 PRINTTAB(4)"[DOWN][DOWN][RVSOFF][GRE
EN] A [WHITE]3-D[GREEN]IMENSIONAL [WHITE
]ADVENTURE[GREEN] GAME  "                                BL
·244 PRINT"[9"[DOWN]"] "TAB(2)"[RVSON] PL
EASE WAIT[5"."]OPENING DUNGEON[3"!"]"                    PP
·245 RETURN                                              IM
·246 D=.:FC=8:IFCL>127THEND=9:FC=15                      AB
·247 P=6:D=.:FC=8:IFCL>127THEND=9:FC=15                  LH
·248 FORX=DTOFC:ON1-(M(X)=.)GOTO249,255                  PD
·249 IFCL<M(X)-PTHENA=M(X)+D(4)                          NB
·250 IFCL>M(X)+PTHENA=M(X)+D(2)                          MM
·251 IFCL>M(X)ANDCL<M(X)+PTHENA=M(X)+D(1)                OI
·252 IFCL<M(X)ANDCL>M(X)-PTHENA=M(X)+D(3)                ND
·253 IFA<.ORA>255THEN255                                 FA
·254 IFDL(A,.)ANDDL(A,1)=.THENDL(A,1)=DL(
M(X),1):DL(M(X),1)=.:M(X)=A                              OK
·255 NEXT                                                IA
·256 D=DL(CL,1):IFDTHENONDGOSUB288,288,29
6,296,301,4,308,305:GOSUB322                             ND
·257 POKEV+21,.:FC=.:GOSUB116:IT=TI+1000:
RETURN                                                   JB
·258 RETURN                                              IM
·259 PRINT"[HOME][RVSOFF][BLUE] SAVING";                 KN
·260 OPEN1,8,15:PRINT#1,"S0:LASTDUNGEON":
CLOSE1                                                   IL
·261 OPEN1,8,2,"LASTDUNGEON,U,W":FORX=.TO
2:FORD=.TO255:PRINT#1,DL(D,X)CR$                         EP
·262 NEXTD:PRINT".";:NEXTX:PRINT".";:FORX
=.TO15:PRINT#1,M(X),CR$,MH(X),CR$                        OF
·263 NEXTX:PRINT".";                                     GH
·264 PRINT#1,CL,CR$,CD,CR$,LF,CR$,L,CR$,U
,CR$,DD,CR$,VS,CR$,HX,CR$,WL,CR$                         CM
·265 PRINT#1,CC,CR$                                      HC
·266 CLOSE1:GOTO185                                      DJ
·267 PRINT"[4"[UP]"]":PRINTTAB(7);:INPUT"
[RVSON] FILENAME  ";A$:A$=LEFT$(A$,11)                   MO
·268 F$="LASTDUNGEON":IFA$<>"[cEP]"THENF$
=A$                                                      OH
·269 OPEN1,8,15:PRINT#1,"I0:":CLOSE1:OPEN
1,8,2,F$+",U,R"                                          LH
·270 FORX=.TO2:FORD=.TO255:INPUT#1,DL(D,X
):NEXTD,X:FORX=.TO15                                     DC
·271 INPUT#1,M(X),MH(X)                                  FF
·272 NEXTX                                               NK
·273 INPUT#1,CL,CD,LF,L,U,DD,VS,HX,WL,CC                 NG
·274 CLOSE1:OPEN1,8,15:INPUT#1,A,B$,C,D:C
LOSE1                                                    KL
·275 IFATHENPRINT"[HOME][WHITE][RVSON]"TA
B(8)A;B$;C;D                                             KD
·276 FORT=1TO5000:NEXT:RETURN                            MP
·277 FORP=STOS+24:POKEP,.:NEXT:POKES+24,1
5:POKES+5,8:POKES+6,255                                  LB
```

```
·278 POKES+4,21:F1=2:FORZ=1TO24:F2=80:POK
  ES+1,F1:FORY=1TO5:POKES+15,F2                FE
·279 F2=F2/1.1:NEXTY:F1=F1+9:NEXTZ:POKES+
  6,15:RETURN                                  HH
·280 FORP=STOS+24:POKEP,.:NEXT:POKES+24,1
  5                                            MM
·281 POKES+5,8:POKES+6,255:POKES+4,23:F1=
  10                                           BG
·282 FORZ=1TO30:F2=30:POKES+1,F1:FORY=1TO
  10:POKES+15,F2                               OC
·285 F2=F2*1.01:NEXT:F1=F1+1:NEXT:POKES+6
  ,15:RETURN                                   FC
·288 FORP=STOS+24:POKEP,.:NEXT:POKES+24,1
  5                                            MM
·289 POKES+5,8:POKES+6,255:POKES+4,23:F1=
  11:F2=25                                     OJ
·290 FORZ=5TO50:POKES+1,F1:POKES+15,F2:F2
  =F2*.97:F1=F1+1:NEXT:POKES+6,15:RETURN       OE
·296 FORP=STOS+24:POKEP,.:NEXT:POKES+24,1
  5:POKES+5,24:POKES+6,248:POKES+4,33          NJ
·299 FORP=17TO13STEP-1:POKES,P:FORT=125TO
  135:POKES+1,T:NEXTT,P:POKES+4,32             JF
·300 RETURN                                    IM
·301 FORP=STOS+24:POKEP,.:NEXT:POKES+24,1
  5                                            MM
·302 POKES+5,8:POKES+6,255:POKES+4,23:F1=
  15:F2=F1                                     PN
·303 FORZ=1TO10:POKES+1,F1:POKES+15,F2:F2
  =F2*1.01:F1=F1*1.3:NEXTZ                     OI
·304 POKES+6,15:RETURN                         CA
·305 FORP=STOS+24:POKEP,.:NEXT:POKES+24,1
  5:POKES+5,8:POKES+6,255:POKES+4,23           GE
·306 F1=8:F2=4:FORZ=1TO30:POKES+1,F1:POKE
  S+15,F2:F2=F2*1.01:F1=F1+1:NEXT              KO
·307 POKES+6,15:RETURN                         CA
·308 FORP=STOS+24:POKEP,.:NEXT                 CN
·309 POKES+24,15:POKES+5,155:POKES+6,245       OH
·310 POKES+4,129:FORP=50TO55:POKES,D           CE
·311 FORT=35TO55:POKES+1,T:NEXTT,P:POKES+
  4,128:FORT=1TO100:NEXT:RETURN                CE
·312 FORP=STOS+24:POKEP,.:NEXT:POKES+24,1
  5                                            MM
·313 POKES+5,8:POKES+6,255:POKES+4,23:FOR
  F1=4TO18STEP2:POKES+15,F1                    CI
·314 FORF2=10TO180STEP2:POKES+1,F2:NEXTF2
  ,F1:POKES+6,15:RETURN                        NM
·315 FORP=STOS+24:POKEP,.:NEXT:POKES+24,1
  5                                            MM
·316 POKES+5,8:POKES+6,255:POKES+4,23          GF
·317 FORF=15TO8STEP-1:FORF1=18TO10STEP-2:
  POKES+15,F1                                  JA
·318 FORF2=12TO4STEP-1:POKES+1,F2:POKES+1
  5,F:NEXTF2,F1,F:POKES+6,15:RETURN            GP
·319 FORP=STOS+24:POKEP,.:NEXT:POKES+24,1
  5                                            MM
·320 POKES+5,96:POKES+6,96:POKES+4,23:POK
  ES+2,85:POKES+1,36                           IG
·321 FORF=1TO10:NEXT:POKES+6,15:RETURN         OI
·322 FORP=STOS+24:POKEP,.:NEXT:POKES+24,1
  5                                            MM
·323 POKES+1,10:POKES,5:POKES+5,129:POKES
  +6,129:POKES+4,21:RETURN                     BI
·324 FORP=STOS+24:POKEP,.:NEXT:POKES+24,1
  5                                            MM
·325 POKES+1,25:POKES,5:POKES+5,20:POKES+
  6,20:POKES+4,129:FORT=1TO75:NEXT             MP
·326 POKES+4,128:RETURN                        EH
·327 DATA 169,,133,251,133,253,169,208,13
  3,252,169,240,133,254,162,8                  IB
·328 DATA 160,,177,251,145,253,136,208,24
  9,230,252,230,254,202,208,240,96             IC
·329 DATA035,24,60,110,110,110,110,110,11
  0                                            BJ
·330 DATA036,255,199,231,129,,60,126,255       KF
·331 DATA037,255,60,24,255,219,153,153,24      KF
·332 DATA038,24,24,60,102,102,60,,             MG
·333 DATA039,,126,127,127,112,126,,255         PD
·334 DATA040,110,110,110,110,110,110,110,
  110                                          EK
·335 DATA042,60,60,255,153,60,60,60,60         HG
·336 DATA081,255,231,231,231,231,219,195,
  231                                          DO
·337 DATA083,231,195,153,189,165,153,195,
  255                                          PD
·338 DATA086,27,31,31,27,59,123,219,155        CH
·339 DATA088,126,102,102,126,24,24,30,30       DK
·340 DATA091,216,248,248,216,220,222,219,
  217                                          JI
·341 DATA092,175,88,171,88,168,88,168,88       CL
·342 DATA102,219,54,219,102,219,54,219,10
  2                                            OK
·343 DATA104,24,255,192,192,192,255,24,24      BA
·344 DATA107,31,24,31,24,31,24,31,24           JM
·345 DATA108,255,182,255,191,240,176,247,
  176                                          JE
·346 DATA109,255,129,255,129,255,255,129,
  255                                          BK
·347 DATA115,248,24,248,24,248,24,248,24       GD
·348 DATA123,255,109,255,253,15,13,239,13      JH
·349 DATA163,255,159,159,239,247,251,253,
  255                                          IN
·350 DATA164,239,239,239,239,239,199,239,
  239                                          EG
·351 DATA165,189,129,153,189,165,129,195,
  231                                          MG
·352 DATA166,255,255,231,219,219,231,255,
  255                                          MD
·353 DATA167,207,183,123,195,129,,255,255      ED
·354 DATA168,199,239,131,239,239,239,239,
  239                                          IJ
·355 DATA169,199,215,199,239,239,239,231,
  231                                          CI
·356 DATA170,129,195,231,195,153,16,153,1
  95                                           KE
·357 DATA171,255,255,102,,90,,255,255          AC
·358 DATA173,255,129,255,255,231,231,255,
```

```
255                                                    IP
·359 DATA186,255,231,255,231,255,231,255,
231                                                    CN
·360 DATA189,255,129,255,239,255,129,255,
239                                                    EK
·361 DATA197,255,,255,255,255,255,,255                 DE
·362 DATA220,250,21,218,21,26,21,26,21                 IP
·363 DATA230,54,177,129,108,108,153,153,1
02                                                     MI
·364 DATA242,255,255,189,189,129,255,219,
255,-1                                                 EA
·365 DATA .,,,,,,,,,,,,,,,.                            MH
·366 DATA .,8,2,1,2,1,,1,1,1,,1,352,4,2,.              JC
·367 DATA .,1,,16,,1,,1,,1,32,1,,2,2,.                 FI
·368 DATA .,1,,1,1,2,16,1,,1,,1,,,1,.                  FO
·369 DATA .,4,1,2,,1,,1,2,1,1,2,1,,4,.                 HL
·370 DATA .,1,,1,,1,1,1,,,,,1,,,,.                     KK
·371 DATA .,2,1,1,1,1,32,1,1,4,1,2,1,1,34
,,                                                     IO
·372 DATA .,,,,,,,,,,,,,,,,.                           MH
·373 DATA .,2,1,1,2,16,2,1,1,1,,2,2,2,2,.              OI
·374 DATA .,1,,1,,1,,1,,1,1,1,1,984,2,.                LO
·375 DATA .,2,1,1,,1,2,1,2,1,,32,2,2,2,.               FG
·376 DATA .,,,2,,1,,1,,1,,129,,,,,.                    CG
·377 DATA .,24,1,1,1,1,,2,1,1,,1,65,,24,.              NI
·378 DATA .,1,,,1,,1,1,,1,,1,,,1,.                     FG
·379 DATA .,1,2,2,1,2,1,,,168,,129,2,1,1,
.                                                      MB
·381 DATA "[UP][RIGHT][s +][DOWN][RVSOFF]
[LEFT][s M][s M][DOWN][LEFT][s M][s M]"                GA
·382 DATA "[UP][UP]#[DOWN][LEFT]([DOWN][L
EFT]([DOWN][LEFT]*"                                    NG
·383 DATA "[UP][RVSON][c *][RIGHT][RIGHT]
[sEP][DOWN][4"[LEFT]"][s B][RVSOFF]$$[RV
SON][s B][DOWN][4"[LEFT]"][s B][c R][c R
][s B]"                                                IK
·384 DATA "[DOWN]&"                                    PN
·385 DATA "[s J][s I][DOWN][LEFT][LEFT][3
"[c +]"]"                                              KF
·386 DATA "[UP]*[DOWN][LEFT][RVSOFF][s B]
[DOWN][LEFT][s B]"                                     BB
·387 DATA "[DOWN][LEFT][s X]"                          FA
·388 DATA "[UP][UP][3"[LEFT]"][c *][RVSON
] [RVSOFF][sEP][DOWN][3"[LEFT]"][RVSON][
sEP][c T][c *][DOWN][4"[LEFT]"][sEP] [RV
SOFF]'[RVSON][SS][c *][DOWN][5"[LEFT]"][
RVSOFF][c *][RVSON][3"[c T]"][RVSOFF][sE
P]"                                                    IK
·389 DATA "[LEFT][LEFT][RVSON][c *][sEP][
c *][sEP][DOWN][4"[LEFT]"][4"="]"                      JL
·390 DATA 1,1,2,2,3,3,4,5,5,6,6,6,7,7,7,8              LP
·391 DATA 1,1,2,2,3,4,<,<                              HF
·392 DATA 1,1,2,2,3,4,<,<                              HF
·393 DATA 1,1,2,2,3,65,":;",":;"                       NK
·394 DATA 1,1,2,2,3,65,":978",":978"                   NM
·395 DATA0,,,,,,,                                      PD
·396 DATA0,,,,4,,32,6                                  OC
·397 DATA0,96,5,60,160,8,255,16                        JE
·398 DATA016,255,8,19,255,200,36,255                   EC
·399 DATA036,9,255,144,18,255,72,37                    FF
·400 DATA126,164,42,60,84,36,24,36                     GF
·401 DATA042,60,84,4,90,32,2,126                       AN
·402 DATA064,1,90,128,,36,,                            AC
·403 DATA0,,,56,255,,67,255                            CO
·404 DATA192,55,255,224,15,255,240,15                  ED
·405 DATA255,240,15,255,240,14,255,176                 CE
·406 DATA014,127,48,30,190,184,62,221                  PJ
·407 DATA188,127,107,126,252,255,159,120               NB
·408 DATA221,143,120,73,7,56,127,3                     LK
·409 DATA024,93,3,12,8,30,30,                          FB
·410 DATA047,53,,10,85,,20,                            DJ
·411 DATA052,66,15,24,102,31,44,90                     KG
·412 DATA033,6,102,1,3,90,1,7                          BN
·413 DATA165,227,11,219,211,21,231,175                 PC
·414 DATA046,231,119,89,126,189,113,219                II
·415 DATA153,96,165,1,113,153,129,107                  JI
·416 DATA126,193,102,195,97,37,129,161                 CP
·417 DATA015,,241,15,,241,10,                          DO
·418 DATA081,14,,113,58,,92,220                        OD
·419 DATA0,120,,,132,,1,74                             IC
·420 DATA016,1,2,40,3,135,40,5                         OO
·421 DATA134,144,14,205,144,31,123,208                 PH
·422 DATA061,183,248,61,207,228,41,183                 FH
·423 DATA100,41,207,44,41,255,56,26                    KB
·424 DATA001,24,3,207,,3,183,                          HP
·425 DATA007,207,,5,122,128,11,239                     NG
·426 DATA128,13,123,64,13,86,192,220                   HF
·427 DATA0,248,,1,4,,2,2                               FH
·428 DATA0,2,138,,2,2,,2                               EO
·429 DATA002,,2,3,6,3,7,9                              NF
·430 DATA003,143,9,3,223,9,13,254                      ID
·431 DATA198,30,253,230,63,123,246,127                 DO
·432 DATA183,255,191,207,247,223,183,250               JD
·433 DATA159,207,242,159,255,242,151,255               EA
·434 DATA210,103,255,204,2,,64,231                     FO
·435 DATA003,231,192,7,219,224,7,189                   LJ
·436 DATA224,15,219,224,15,231,240,15                  JJ
·437 DATA255,240,15,255,240,15,255,240                 KD
·438 DATA015,255,240,15,255,240,15,255                 IJ
·439 DATA240,15,255,240,31,255,248,31                  AP
·440 DATA255,248,31,127,216,61,239,124                 LG
·441 DATA047,191,236,58,250,172,45,149                 DN
·442 DATA246,95,86,174,164,221,202,102                 OM
·443 DATA0,,254,,1,127,,2                              HB
·444 DATA223,,4,207,,11,103,                           GE
·445 DATA021,191,,43,215,,87,223                       PF
·446 DATA0,87,224,,87,243,,91                          KO
·447 DATA251,,45,255,,22,255,                          PE
·448 DATA011,,,5,191,,2,192                            PO
·449 DATA0,1,63,,,213,,                                AA
·450 DATA090,,,45,,,42,220                             MJ
·451 DATA127,,,254,128,,251,64                         ED
·452 DATA0,243,32,,230,208,,253                        FD
·453 DATA168,,235,212,,251,234,                        FB
·454 DATA007,234,,207,234,,223,218                     JP
·455 DATA0,255,180,,255,104,,                          IP
```

```
•456 DATA208,,253,160,,3,64,                PI
•457 DATA252,128,,171,,,90,                 CO
•458 DATA0,180,,,84,,,220                   JK
•459 DATA061,188,,42,84,,61,188             JI
•460 DATA0,42,84,,61,188,,62                NK
•461 DATA124,,45,180,,61,188,               KI
•462 DATA042,84,,45,180,,62,124             EP
•463 DATA0,61,188,,42,84,,61                EL
•464 DATA188,,42,84,,45,180,                PN
•465 DATA058,92,,45,180,,21,90              IB
•466 DATA0,22,173,,11,86,128,239            GI
•467 DATA002,189,64,3,195,192,2,189         OC
•468 DATA064,2,195,64,3,189,194,2           FB
•469 DATA195,65,3,189,193,2,67,67           CH
•470 DATA005,166,199,10,221,74,21,106       KH
•471 DATA156,38,182,148,91,87,92,108        MC
•472 DATA171,182,87,85,202,88,169,170       LF
•473 DATA103,214,106,89,69,214,39,255       AA
•474 DATA236,26,105,88,7,255,240,28         AE
•475 DATA007,231,224,11,126,208,21,60       DM
•476 DATA168,46,255,116,95,126,250,95       IA
•477 DATA129,250,95,189,250,47,255,244      GK
•478 DATA020,255,40,11,,208,4,255           PB
•479 DATA032,3,,192,2,230,64,3              GL
•480 DATA153,192,3,219,192,2,36,64          GK
•481 DATA002,219,64,3,36,192,2,219          JH
•482 DATA064,3,36,192,2,219,64,             JI
•483 DATA0,252,,1,122,,3,183                FE
•484 DATA0,3,207,,1,254,,                   ID
•485 DATA252,,,120,,,120,                   JI
•486 DATA0,120,,,120,,,120                  JO
•487 DATA0,,120,,,120,64,                   FB
•488 DATA120,32,,120,32,,252,96             LG
•489 DATA001,255,192,15,,224,31,255         AH
•490 DATA240,24,3,240,15,255,224,221        OK
```

DVORAK KEYBOARD
FROM PAGE 57

C-64 VERSION

```
•10 REM DVORAK KEYBOARD FOR THE C-64        ND
•20 FORT=49152TO49198:READDT:POKET,DT:NEX
 TT                                         HO
•30 SYS49152:POKE49153,0:POKE49154,224:PO
 KE49156,0:POKE49157,224                    ML
•40 POKE49162,255:POKE49192,255:SYS49152    LJ
•50 POKE1,53:REM ACTIVATE RAM KERNAL        EE
•60 FORT=0TO32:REM REDEFINE NORMAL KEY DE
 FS                                         JN
•70 READAD:READVA                           HP
•80 POKE60289+AD,VA                         KF
•90 NEXTT                                   NG
•100 FORT=0TO37:REM REDEFINE SHIFTED KEY
 DEFS                                       DH
•110 READAD:READVA                          HP
•120 POKE60354+AD,VA                        KK
•130 NEXTT                                  NG
•140 NEW:END                                GH
•150 REM ML ROUTINE FOR COPYING BASIC AND
  KERNAL ROM TO RAM                         IN
•160 DATA173,0,160,141,0,160,172,5,192,19
 2,191,240,23,238,1,192,238,4,192           FI
•170 DATA173,4,192,240,3,76,0,192,238,2,1
 92,238,5,192,76,0,192,172,4,192,192        AF
•180 DATA255,240,3,76,13,192,96             FD
•190 REM NON SHIFTED DATA                   LC
•200 DATA62,39,9,44,14,46,17,80,22,89,25,
 70,30,71,33,67,38,82,41,76,46,63           GE
•210 DATA13,79,18,69,21,85,26,73,29,68,34
 ,72,37,84,42,78,45,83,50,45                GI
•220 DATA12,59,23,81,20,74,31,75,28,88,39
 ,66,36,77,47,87,44,86,55,90,40,91,43,61    IM
•230 REM SHIFTED DATA                       MH
•240 DATA59,64,24,38,27,42,32,40,35,41,40
 ,93,43,43                                  GP
•250 DATA62,34,9,60,14,62,17,208,22,217,2
 5,198,30,199,33,195,38,210,41,204,46,47    HH
•260 DATA13,207,18,197,21,213,26,201,29,1
 96,34,200,37,212,42,206,45,211,50,45       IO
•270 DATA12,58,23,209,20,202,31,203,28,21
 6,39,194,36,205,47,215,44,214,55,218       BO
```

C-128 VERSION

```
•10 REM DVORAK KEYBOARD FOR THE C-128       LI
•20 FAST                                    CF
•30 FORI=0TO177:REM COPY DOWN NORMAL KEY
 DEFS                                       EA
•40 POKEDEC("1300")+I,PEEK(DEC("FA80")+I)   CM
•50 NEXTI                                   MN
•60 FORI=0TO177:REM COPY DOWN SHIFTED KEY
  DEFS                                      BE
•70 POKEDEC("1359")+I,PEEK(DEC("FAD9")+I)   FH
•80 NEXTI                                   MN
•90 REM CHANGE KEY TABLE POINTERS           FD
•100 POKE830,DEC("00")                      CN
•110 POKE831,DEC("13")                      EO
•120 POKE832,DEC("59")                      DD
•130 POKE833,DEC("13")                      DI
•140 REM REDEFINE NORMAL KEY DEFS           LN
•150 FORT=0TO32                             NN
•160 READAD:READVA                          HP
•170 POKEDEC("1300")+AD,VA                  NA
•180 NEXTT                                  NG
•190 REM REDEFINE SHIFTED KEY DEFS          LM
•200 FORT=0TO37                             NI
•210 READAD:READVA                          HP
•220 POKEDEC("1359")+AD,VA                  LE
•230 NEXTT                                  NG
•240 SLOW                                   CE
•250 NEW:END                                GH
```

```
·260 REM NON SHIFTED DATA                                    LC
·270 DATA62,39,9,44,14,46,17,80,22,89,25,
 70,30,71,33,67,38,82,41,76,46,63                            GE
·280 DATA13,79,18,69,21,85,26,73,29,68,34
 ,72,37,84,42,78,45,83,50,45                                 GI
·290 DATA12,59,23,81,20,74,31,75,28,88,39
 ,66,36,77,47,87,44,86,55,90,40,91,43,61                     IM
·300 REM SHIFTED DATA                                        MH
·310 DATA59,64,24,38,27,42,32,40,35,41,40
 ,93,43,43                                                   GP
·320 DATA62,34,9,60,14,62,17,208,22,217,2
 5,198,30,199,33,195,38,210,41,204,46,47                     HH
·330 DATA13,207,18,197,21,213,26,201,29,1
 96,34,200,37,212,42,206,45,211,50,45                        IO
·340 DATA12,58,23,209,20,202,31,203,28,21
 6,39,194,36,205,47,215,44,214,55,218                        BO
```

DISK CHECKUP
FROM PAGE 97

```
·100 GOTO 280                                                CG
·110 REM INNER TRACE LOOP                                    NP
·120 PRINT#1,"U1 2 0";T;S                                    IN
·130 INPUT#1, EN,EM$,ET,ES                                   OA
·140 IF EN<>0 THEN GOTO 1960:REM DISK TRA
 CE ERROR                                                    AK
·150 LT%(I)=LT%(I)+1                                         HO
·160 IF OB%(T,S)<>0 THEN GOTO 1410:REM FI
 LE INTERSECT                                                FJ
·170 OB%(T,S)=I                                              AD
·180 GET#2,A$,B$                                             DH
·190 IF A$="" THEN GOTO 1450:REM TRACE LO
 OP CONTINUE                                                 FI
·200 IF B$="" THEN B$=CHR$(0)                                OE
·210 T=ASC(A$):S=ASC(B$)                                     EN
·220 GOTO 120:REM INNER TRACE LOOP                           EP
·230 REM******************************                      CE
·240 REM DISK CHECK-UP                                       AI
·250 REM                                                     JD
·260 REM   DON FULTON                                        FI
·270 REM******************************                      CE
·280 DIM I,S,T,X,Y,Z                                         GA
·290 DIM BC,EN,ES,ET,IM,LC,OF,PF,S1,S2,SF
 ,T1,T2                                                      EB
·300 DIM TS%(144),SS%(144),LD%(144),LT%(1
 44),PF%(144),N$(144)                                        NG
·310 DIM OB%(35,20)                                          HC
·320 CLOSE 1:OPEN 1,8,15:CLOSE 2:CLOSE 1                     AN
·330 REM----------------------------                        CD
·340 PRINT CHR$(147)                                         FG
·350 PRINT                                                   JJ
·360 PRINT "[12" "]DISK CHECK-UP                             OO
·370 PRINT "[11" "][c T]                                     LL
·380 PRINT " CHECKS:                                         OD
·390 PRINT                                                   JJ
·400 PRINT "  -- START LINKS DIFFERENT                       PB
·410 PRINT                                                   JJ
·420 PRINT "  -- TRACE FILE LEN = DIR FIL
 E LEN                                                       NA
·430 PRINT                                                   JJ
·440 PRINT "  -- FILES DO NOT OVERLAP                        HC
·450 PRINT                                                   JJ
·460 PRINT "  -- (3) BLOCKS FRE                              FO
·470 PRINT                                                   JJ
·480 REM----+++++++-------+++++++++---                       DD
·490 REM SHO        -36 CY-          SHP                     NP
·500 REM CMG                         CMM                     GB
·510 REM CMG       BOX BELOW         CMM                     BD
·520 REM CMG                         CMM                     GB
·530 REM SHL        -36 CP-          SH@                     FJ
·540 REM------------------------------                       DD
·550 PRINT " [s 0][36"[c Y]"][s P]                           HL
·560 PRINT " [c G]THIS UTILITY DOES NOT W
 RITE TO DISK.[c M]                                          EN
·570 PRINT " [c G][36" "][c M]                               CK
·580 PRINT " [c G][3" "][RVSON](PROBLEM F
 ILES ARE DISPLAYED)[RVSOFF][4" "][c M]                      IA
·590 PRINT " [c G][10" "][RVSON]IN INVERS
 E VIDEO)[RVSOFF][9" "][c M]                                 OK
·600 PRINT " [s L][36"[c P]"][s @]                           PM
·610 FOR X=1 TO 1000:NEXT                                    DO
·620 PRINT                                                   JJ
·630 REM------------------------------                       CD
·640 REM FIND START OF ACTIVE FILES                          HM
·650 PRINT                                                   JJ
·660 PRINT "[6" "]START[20" "]LENGTH  "                      GB
·670 PRINT "FILE  T , S[4" "]FILE NAME[5"
  "]DIR--TRACE"                                              CE
·680 PRINT "[4"[c T]"]  [5"[c T]"][3" "][
 11"[c T]"][4" "][3"[c T]"]  [5"[c T]"]"                     DD
·690 OPEN 1,8,15                                             PG
·700 OPEN 2,8,2,"#"                                          EI
·710 T1=18:S1=1:I=1                                          KH
·720 REM NEXT DIRECTORY SECTOR                               LJ
·730 :  IF T1=0 THEN GOTO1010:REM DIRECTO
 RY CLOSE                                                    BI
·740 :  PRINT#1,"U1 2 0";T1;S1                               EI
·750 :  INPUT#1, EN,EM$,ET,ES                                JL
·760 :  IF EN<>0 THEN GOTO 1980:REM DISK
 ERROR                                                       PN
·770 :  GET#2,A$,B$                                          IE
·780 :  T2=T1:S2=S1                                          EG
·790 :  T1=ASC(A$+CHR$(0))                                   DG
·800 :  S1=ASC(B$+CHR$(0))                                   CK
·810 :  FOR X=2 TO 255 STEP 32                               CJ
·820 :    PRINT#1,"U1 2 0";T2;S2                             FF
·830 :    PRINT#1,"B-P 2";X                                  FI
·840 :    INPUT#1, EN,EM$,ET,ES                              LI
·850 :    IF EN<>0 THEN GOTO 1980:REM DIS
 K ERROR                                                     CK
```

```
·860 :    GET#2,C$,D$,E$,F$,G$,H$,I$,J$,K
 $,L$,M$,N$,O$,P$,Q$,R$,S$,T$,U$          KM
·870 :    IF C$="" THEN GOTO 980:REM DIRE
 CTORY CONTINUE                           MG
·880 :    TS%(I)=ASC(D$+CHR$(0))           FB
·890 :    SS%(I)=ASC(E$+CHR$(0))           GF
·900 :    N$(I)=F$+G$+H$+I$+J$+K$+L$+M$+N
 $+O$+P$+Q$+R$+S$+T$+U$                   LI
·910 :    PRINT#1, "B-P 2";X+28            GB
·920 :    GET#2,V$,W$                      NN
·930 :    LD%(I)=ASC(V$+CHR$(0))+256*ASC(
 W$+CHR$(0))                              NH
·940 :    PRINT I;TAB(4)TS%(I);TAB(8)",";
 TAB(9)SS%(I);                            IC
·950 :    PRINT TAB(13)N$(I);TAB(29)LD%(I
 );TAB(33);"--";LT%(I)                    MA
·960 :    I=I+1                            MJ
·970 :    REM  DIRECTORY CONTINUE          BM
·980 :    NEXT                             DM
·990 GOTO 730:REM NEXT DIRECTORY SECTOR    IM
·1000 REM DIRECTORY CLOSE                  PI
·1010 IM=I-1                               GN
·1020 CLOSE 2:CLOSE 1                      CI
·1030 REM------------------------------    NE
·1040 PRINT                                JJ
·1050 PRINT " CHECKING START LINKS FOR OV
 ERLAP"                                   HM
·1060 PRINT                                JJ
·1070 IF IM=0 OR IM=1 THEN GOTO 1250:REM
  START LINK END                           ON
·1080 FOR I=2 TO IM                        FJ
·1090 :  FOR Y=1 TO I-1                    GN
·1100 :    IF TS%(Y)=TS%(I)THEN IF SS%(Y)
 =SS%(I)THEN GOTO 1150:REM FILE OVERLAP    NK
·1110 :  NEXT                              PP
·1120 NEXT                                 IA
·1130 GOTO 1250:REM START LINK END         CO
·1140 REM FILE OVERLAP                     BO
·1150 :    PF%(I)=1:PF%(Y)=1               GM
·1160 :    OF=1:REM SET OVERLAP FLAG       AK
·1170 :    PRINT Y;CHR$(18);TAB(4)TS%(Y);
 TAB(8)",";                               EH
·1180 :    PRINT TAB(9)SS%(Y);TAB(13)N$(Y
 );CHR$(146);                             LO
·1190 :    PRINT TAB(29)LD%(Y);TAB(33);"-
 -";LT%(Y)                                LI
·1200 :    PRINT I;CHR$(18);TAB(4)TS%(I);
 TAB(8)",";                               IH
·1210 :    PRINT TAB(9)SS%(I);TAB(13)N$(I
 );CHR$(146);                             DO
·1220 :    PRINT TAB(29)LD%(I);TAB(33);"-
 -";LT%(I)                                FI
·1230 GOTO 1110:REM START LINK CONTINUE    HN
·1240 REM  START LINK END                  LM
·1250 IF OF=0 THEN PRINT "[5" "]--START L
 INKS OK--"                               IH
·1260 REM------------------------------    KC
·1270 PRINT                                JJ
·1280 PRINT " TRACING FILES"               KC
·1290 PRINT                                JJ
·1300 OPEN 1,8,15                          PG
·1310 OPEN 2,8,2,"#"                       EI
·1320 IF IM=0 THEN GOTO 1530:REM TRACE EN
 D                                         KO
·1330 FOR I=1 TO IM                        FI
·1340 :    T=TS%(I)                        BC
·1350 :    S=SS%(I)                        BC
·1360 :    LT%(I)=0                        PD
·1370 :    PF$=""                          GN
·1380 :    SF=0                            CH
·1390 :    GOTO 120:REM INNER TRACE LOOP   KI
·1400 :    REM FILE INTERSECT              FA
·1410 :    PF%(OB%(T,S))=1                 DO
·1420 :    PF%(I)=1                        OA
·1430 :    SF=1:REM SET INTERSECT FLAG     FN
·1440 :    REM TRACE LOOP CONTINUE         GG
·1450 :    BC=BC+LT%(I)                    GJ
·1460 :    PRINT I;TAB(4)TS%(I);TAB(8)","
 ;TAB(9)SS%(I);                           IC
·1470 :    IF LT%(I)<>LD%(I) THEN PF%(I)=
 PF%(I)+2:PRINT CHR$(18);                 CM
·1480 :    PRINT TAB(13)N$(I);TAB(29)LD%(
 I);TAB(33);"--";LT%(I)                   MA
·1490 :    IF PF$<>""THEN PRINT "[3" "](F
 ILE";I;"CONTAINS ";+PF$;")":PRINT        FG
·1500 :    IF SF=1 THEN PRINT "[3" "](FIL
 E";I;"INTERSECTS FILE ";OB%(T,S);")":PRI
 NT                                       PC
·1510 NEXT                                 IA
·1520 REM TRACE END                        CM
·1530 CLOSE 2:CLOSE 1                      CI
·1540 REM------------------------------    NE
·1550 PRINT                                JJ
·1560 PRINT " PROBLEM FILES--"             MI
·1570 PRINT " [13"[c T]"]"                 IF
·1580 IF IM=0 THEN GOTO 1710:REM PROBLEM
 FILE END                                  CO
·1590 FOR I=1 TO IM                        FI
·1600 :  IF PF%(I)=0 THEN GOTO 1690:REM P
 ROBLEM LOOP CONTINUE                      IE
·1610 :  PF=1:REM SET PROBLEM FILE FLAG    KG
·1620 :  PRINT I;                          LL
·1630 :  IF PF%(I)=1 OR PF%(I)=3 THEN PRI
 NT CHR$(18);                              PG
·1640 :  PRINT TAB(4)TS%(I);TAB(8)",";TAB
 (9)SS%(I);                                CE
·1650 :  PRINT CHR$(18);TAB(13)N$(I);CHR$
 (146);                                    FM
·1660 :  IF PF%(I)=2 OR PF%(I)=3 THEN PRI
 NT CHR$(18);                              OD
·1670 :  PRINT TAB(29)LD%(I);TAB(33);"--"
 ;LT%(I)                                   BH
·1680 :  REM  PROBLEM LOOP CONTINUE        LM
·1690 NEXT                                 IA
·1700 REM PROBLEM FILE END                 OM
·1710 IF PF=0 THEN PRINT "[6" "]NONE"      GB
```

```
•1720 PRINT                                          JJ
•1730 PRINT " BLOCK COUNT = ";BC                     NP
•1740 PRINT                                          JJ
•1750 PRINT " BLOCKS FREE = ";664-BC;TAB(
 20)"(TRACE)"                                        CO
•1760 FOR I=0 TO IM                                  FH
•1770 :  LC=LC+LD%(I)                                EA
•1780 NEXT                                           IA
•1790 PRINT " BLOCKS FREE = ";664-LC;TAB(
 20)"(DIR)"                                          OH
•1800 REM------------------------------              NE
•1810 REM DIR BAM BLOCKS FREE                        KG
•1820 OPEN 1,8,15                                    PG
•1830 OPEN 2,8,2,"#"                                 EI
•1840 PRINT#1,"U1 2 0";18;0                          EA
•1850 INPUT#1, EN,EM$,ET,ES                          BP
•1860 IF EN<>0 THEN GOTO 1980:REM DISK ER
 ROR                                                 PO
•1870 FOR X=0 TO 35                                  CC
•1880 :  GET#2,W$,X$,Y$,Z$                           DK
•1890 :  IF X<>0 AND X<>18 THEN BU=BU+ASC
 (W$+CHR$(0))                                        JE
•1900 NEXT                                           IA
•1910 PRINT " BLOCKS FREE = ";BU;TAB(20)"
 (BAM)"                                              KN
•1920 CLOSE 2:CLOSE 1                                CI
•1930 END                                            IC
•1940 REM------------------------------              NE
•1950 REM DISK TRACE ERROR                           EI
•1960 IF EN=66 THEN PF$="ILLEGAL LINK":GO
 TO 1450:REM TRACE LOOP CONTINUE                     LB
•1970 REM DISK ERROR                                 IP
•1980 PRINT                                          JJ
•1990 PRINT " UNRECOVERABLE DISK ERROR"              GI
•2000 PRINT                                          JJ
•2010 PRINT EN;EM$;ET;ES                             NP
•2020 CLOSE 2:CLOSE 1                                CI
•2030 END                                            IC
```

SCUTTLEBUTT

Continued from page 14

within a range of 10,000 years, and zoom in for views of constallations, the moons of Jupiter, a solar eclipse, the position of Halley's Comet, the Virgo Cluster of galaxies, Venus crossing the sun, and more. Price is $64.95; lab pack, $194.85.

CBS Interactive Learning, 203-622-2500 (see address list, page 14).

AMIGA MANUAL

The final release in Addison-Wesley's four-volume Amiga Technical Reference Series, the *Amiga ROM Kernal Reference Manual: Libraries and Devices* ($34.95), lists and describes the Amiga's built-in ROM routines and systems software which support graphics, sound, and animation. (Previously published were the *Amiga Hardware Reference Manual, Amiga Intuition Reference Manual,* and *Amiga ROM Kernal Reference Manual: Exec,* each $24.95.)

Addison-Wesley Publishing Company, 617-944-3700 (see address list, page 14).

DIGITAL SAMPLING

Sampler-64 allows the musician to digitally convert audible sound into a series of numbers that can be stored in memory, processed in various ways, and replayed on a keyboard. Complex sounds can be created with facilities like looping, dubbing (mixing different sampled sounds), block editing (dividing a sample into eight blocks which can be arranged at will), echo, and reverb. The package includes a small hardware box that attaches to the 64's user port, menu-driven software, a microphone, and a cable to route output to a TV. A MIDI interface that will be available by the end of the year will allow keyboard control of *Sampler-64*, and incorporate a sequencer. Price is $89.95 plus $3.50 shipping; PA residents add 6% sales tax.

The *Com-Drum* upgrade for *Sampler-64* turns the program into a drum machine, with real time and step time sequencer, three different drumkits with eight precussion sounds to each, and variable tempos and time signatures. Price is $29.95, or $14.95 when purchased with *Sampler-64.* (PA residents add 6%.)

Micro Arts Products, 215-336-1199 (see address list, page 14).

HOW-TO PROGRAMS

ShareData will market a *Home Companion* series of interactive how-to software, providing information to help the user diagnose and solve specific problems in areas like auto maintenance, weight control and nutrition, and money management. Twenty programs are planned, with five scheduled for fall release. Price will be under $10 each.

ShareData, Inc., 612-829-0409 (see address list, page 14).

HI TECH ♥ EXPRESSIONS™

Celebrate Every Occasion With Your Computer!

You've no idea just how much fun your computer can be! With Hi Tech Expressions creative software, your computer will come alive with musical animation, paper printouts, and computer greetings.

With CardWare and HeartWare you can personalize the animation to make your own greeting disks. Or express special wishes on full-page printouts and folded cards.

Throw a high-tech party with PartyWare. Create festive invitations, banners, hats, games . . . all kinds of party stuff. You can even make floppy disk invitations!

And the data base will make sure that no occasion or friend is forgotten.

Add an artistic flair to all of your Hi Tech Expressions with WareWithAll. It's the state-of-the-art supply kit with everything you need to jazz up your computer creations.

Your computer never had such fun!

CardWare™

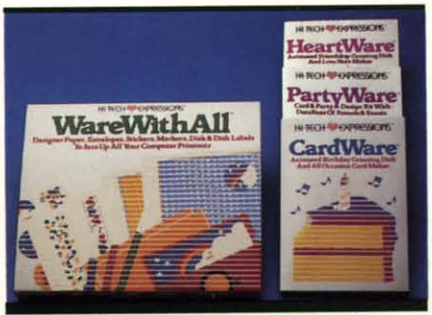
Animated Birthday Greeting Disk And All-Occasion Card Maker
$9.95 Suggested Retail Price

PartyWare™

Card And Party Design Kit With Database Of Friends And Events
$14.95 Suggested Retail Price

Available for Commodore 64/128. Atari 400, 800, 800 XL, 1200 XL, 130 XE, Apple IIe, c,+ with 64K, and IBM PC, XT, AT and PC jr. with 256K and a color graphics adapter.

*Software not included in WareWithAll.



Order Now From Your Distributor.

Reader Service No. 109

HeartWare™

Animated Friendship Greeting Disk And Love Note Maker
Suggested Retail Price **$9.95**

WareWithAll™

Colorfully Designed Paper, Envelopes, Stickers, Markers, Disk Labels, Disk*
Suggested Retail Price **$14.95**

Hi Tech Expressions, Inc.
2699 South Bayshore Drive
Suite 1000A, Coconut Grove, Florida 33133
305/854-2318 1-800-848-9273
Telex 6712408 IMICO

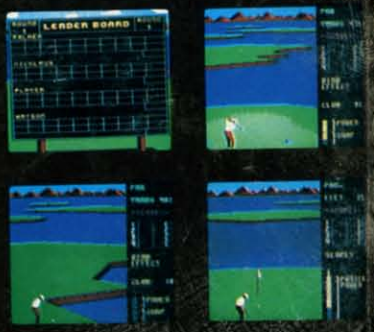
TWO EXCITING SPORTS SIMULATORS FROM ACCESS!

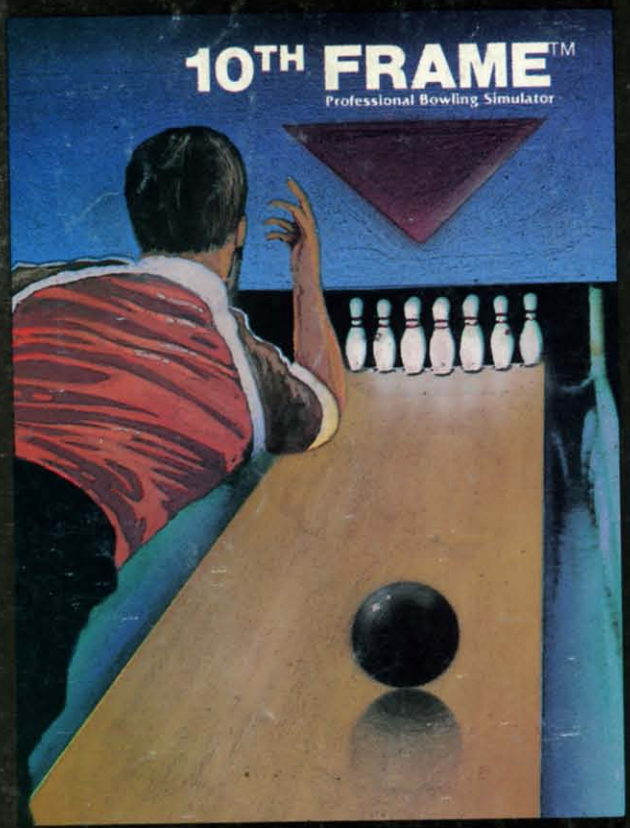
LEADER BOARD™

"Pro Golf Simulator"
Bruce & Roger Carver

Access welcomes you to the exciting world of professional golf with **LEADER BOARD.** Featuring amazing realistic 3-D animation, golfers will face multiple 18-hole courses that demand strategy, total concentration and control to come in under par. **LEADER BOARD** is the most realistic golf game available because it is actually a simulator that gives you a true perspective view of your golf game. Features include:

- 1 to 4 Players
- Choice of Club, Distance, Type of Shot (hook, slice, putting) and More
- Realistic Sound
- Joystick Controlled
- Multiple 18-Hole Golf Courses
- Automatic Scoring
- Disk: Commodore 128/64, Amiga, Atari 520ST, Atari 8 BIT
- Practice Driving Range
- Suggested Retail Price $39.95





NEW!

10TH FRAME™

The Professional Bowling Simulator
By Bruce & Roger Carver

Put on your shoes and get ready to roll with 10th Frame. Featuring outstanding graphics and true perspective view, 10th frame captures the look and "feel" of real bowling and sets a new standard of excellence for computer based sport simulation. Features include:

- Up to eight different players including league play.
- Multiple levels of play.
- Outstanding sound and pin action
- Joystick controlled
- Automatic scoring
- Suggested retail price $39.95
- Available for Commodore 128/64

Attention Dealers!
If you wish to be included on our dealer list for new product information and special promotional material, please call 1-(800) 824-2549

ACCESS™
Software Incorporated

Reader Service No. 110

If you can't find our products at your local dealer, you can order direct.
For mail orders, enclose check or money order, plus $1.00 shipping and handling, and specify machine version desired. Order by phone on VISA, Master Charge or C.O.D. by calling (801) 298-9077.
ACCESS SOFTWARE, INC. 2561 South 1560 West
Woods Cross, UT 84087

October 1986 *Ahoy!* Void after October 6, 1986

READER SERVICE CARD

To request additional information on any product advertised or otherwise pictured in this issue of *Ahoy!* that is accompanied by a reader service number, circle the corresponding number below and mail this card. We will promptly forward your request to the designated companies.

101	102	103	104	105	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127	128	129	130
131	132	133	134	135	136	137	138	139	140	141	142	143	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159	160
161	162	163	164	165	166	167	168	169	170	171	172	173	174	175	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190
191	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207	208	209	210	211	212	213	214	215	216	217	218	219	210
211	212	213	214	215	216	217	218	219	220	221	222	223	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239	240
241	242	243	244	245	246	247	248	249	250	251	252	253	254	255	256	257	258	259	260	261	262	263	264	265	266	267	268	269	270
271	272	273	274	275	276	277	278	279	280	281	282	283	284	285	286	287	288	289	290	291	292	293	294	295	296	297	298	299	300

A. Please check whether you are...
1. ☐ male 2. ☐ female

B. What is your age?
1. ☐ under 18 2. ☐ 18-24 3. ☐ 25-34
4. ☐ 35-44 5. ☐ 45-54 6. ☐ 55-64 7. ☐ 65+

C. Education level completed
1. ☐ elementary 2. ☐ high school
3. ☐ junior college 4. ☐ college graduate
5. ☐ master's degree 6. ☐ PhD

D. From which of the following sources did you obtain your copy of *Ahoy!*?
1. ☐ newsstand 2. ☐ subscription (mail)
3. ☐ from a friend or family member
4. ☐ other_____

If not currently a subscriber, do you plan to become one?
1. ☐ yes 2. ☐ no

Name_____

Street_____

City_____ State_____ Zip_____

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 65 MT. MORRIS, ILL.

POSTAGE WILL BE PAID BY ADDRESSEE

Ahoy!

P.O. BOX 341
MT. MORRIS, ILL. 61054

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 65 MT. MORRIS, ILL.

POSTAGE WILL BE PAID BY ADDRESSEE

Ahoy!

P.O. BOX 341
MT. MORRIS, ILL. 61054

PLACE
STAMP
HERE

AHOY! MAGAZINE
P.O. Box 2501
Clinton, Iowa 52735