COMPUTE!'s

Machine Language Games for the COMMODORE 64

Seven outstanding machine language games for the Commodore 64 and 128 (in 64 mode). Dodge missiles, slide on ice, run for president, and much more. Complete source code for all games included.

A **COMPUTE! Books** Publication

$16.95

COMPUTE!'s

Machine Language Games for the COMMODORE 64

COMPUTE!™ Publications, Inc. abc
Part of ABC Consumer Magazines, Inc.
One of the ABC Publishing Companies
Greensboro, North Carolina

The following article was originally published in *COMPUTE!'s Second Book of Commodore 64 Games*, copyright 1984, COMPUTE! Publications, Inc.: "Nessie."

The following articles were originally published in *COMPUTE!'s Gazette*, copyright 1984, COMPUTE! Publications, Inc.: "CUT-OFF!" (March); "Campaign Manager" (August).

The following article was originally published in *COMPUTE!* magazine, copyright 1985, COMPUTE! Publications, Inc.: "Laser Beam" (October).

The following articles were originally published in *COMPUTE!'s Gazette*, copyright 1985, COMPUTE! Publications, Inc.: "Heat Seeker" (March); "Whirlybird" (December).

The following article was originally published in *COMPUTE!* magazine, copyright 1986, COMPUTE! Publications, Inc.: "Miami Ice" (June).

The following article was originally published in *COMPUTE!'s Gazette*, copyright 1986, COMPUTE! Publications, Inc.: "An Introduction to Assemblers" (August).

Printed in the United States of America

ISBN 0-87455-061-0

10 9 8 7 6 5 4 3 2 1

The authors and publisher have made every effort in the preparation of this book to insure the accuracy of the programs and information. However, the information and programs in this book are sold without warranty, either express or implied. Neither the authors nor COMPUTE! Publications, Inc., will be liable for any damages caused or alleged to be caused directly, indirectly, incidentally, or consequentially by the programs or information in this book.

The opinions expressed in this book are solely those of the authors and are not necessarily those of COMPUTE! Publications, Inc.

COMPUTE! Publications, Inc., Post Office Box 5406, Greensboro, NC 27403, (919) 275-9809, is part of ABC Consumer Magazines, Inc., one of the ABC Publishing Companies, and is not associated with any manufacturer of personal computers. Commodore 64 and Commodore 128 are trademarks of Commodore Electronics Limited.

Contents

Chapter 1
The Games

Fast and Furious

Gregg Keizer/Editor

BASIC is just too slow. Anyone who's programmed on the Commodore 64 knows that for arcade game speed, or for simulation sophistication (and its accompanying complexity), nothing beats machine language.

Machine language (ML) is the native "tongue" of any computer, and the 64 is no exception. The ones and zeros (ons and offs) of machine language, though, are a bit more difficult to work with than the English-style syntax of a computer language like BASIC. To write an ML game, you have to know what you're doing.

The authors of these seven machine language games for the Commodore 64 (and 128 running in 64 mode) obviously know their stuff. The games are first and foremost fun to play. But almost as important is their length. Since they all appeared originally in *COMPUTE!* magazine or *COMPUTE!'s Gazette,* they had to be short enough to publish—not so long that people refused to type them in. The longest game, "Campaign Manager," is just over 7K long in its typed-in form. That's considerably shorter than most commercial games. Consider the length of these seven games and they're even more impressive.

Play One, Play Them All

You'll find a variety of games, both in subject matter and style of play, in *COMPUTE!'s Machine Language Games for the Commodore 64.* From strategy games that take careful thought to frenetic games where things *blip* and *zip,* there's something here for any gamester.

CUT-OFF! is a takeoff on the classic game *Blockade.* Two players hurtle their lines of light across the screen, each trying to make the other smash into a wall or previous path. Ten levels make the game enjoyable for everyone in the family.

Nessie is out of the ordinary, if only because it's a completely nonviolent action game. Hunt the famed Loch Ness monster, but with your instant camera. Capture Nessie on film—a clean, clear shot—and you're ahead of the game. But snap a pic of an eel or turtle, blur or crop the photo, and you end up with nothing. Fun for everybody.

Campaign Manager is a premier simulation of our country's most important process—electing a president. Two players direct the nine-week campaign of a candidate from the nomination to the final balloting. Just some of the options include pressing the flesh, television advertising, polling, fundraising, and taking stands on the issues. This may not be an election year, but the way politicians are acting, 1988 can't be far away.

Heat Seeker is a one-player game of jet plane rolls and loops. Heat-seeking missiles home in on your jet as you try to destroy their bases. Arcade-fast, with lightning-quick controls, this game almost makes you dizzy as your plane jinks out of harm's way.

Laser Beam sends your character scurrying across a chaotic field of bouncing balls. Let one touch you and you're a goner. Zap the right ball at the right time with your laser, render it harmless, then run out, grab it, and stuff it in the basket at the bottom of the screen. Dodge and twist (sounds like a dance, doesn't it?) from top to bottom. Frantic fun.

Whirlybird twirls a spread-eagled bird at the top of the screen. Use it to return the eggs that bounce back from the bricks below. Let one get by you and the trouble begins. The unusual accelerating movement of the bird makes it tougher than it sounds.

Miami Ice isn't a Canadian version of a popular TV show. Instead, it's a nightmare of slick ice, a sliding car, and guardrails that fill the windshield. Maneuver through seven different screens as you try to make it to the parking garage across town before you smash your car to smithereens. You'll get sweaty palms from this one.

Easy Entry

If you had to use an assembler (see Chapter 3) to enter these games, you'd probably give up before you started. Fortunately, you can use our "MLX" machine language entry program instead. MLX is a program in its own right, one that checks your typing and lets you know—almost immediately—that you've made a mistake. Typing in machine language programs with MLX guarantees that you'll have a working copy of the game the first time.

All seven games in *COMPUTE!'s Machine Language Games for the Commodore 64* are listed in MLX format. Refer to Chapter 2 for the MLX program, its instructions, and each game's MLX listing and individual directions.

From the Source

Source code is the original code of a program. In the case of machine language games like these seven, the source code was written with an assembler (again, see Chapter 3). If you know how to read it (even if you don't know how to write it), source code can be illuminating. It shows you exactly how the program was put together, what routines were used, and which tricky techniques were called into play.

Source code is hard to get. Most programmers guard it with their lives. That makes this book even more of a departure, because we've published **the complete assembled source code for each game.**

If you do dive into machine language programming, then you'll be sure to turn to Chapter 3. You'll undoubtedly find programming methods that will make your own creations run faster, look better, and play easier.

Grab the Joystick

No matter what you use this book for—to play the games or pore over the source code—*COMPUTE!'s Machine Language Games for the Commodore 64* will satisfy even the most jaded player or programmer. Get ready for some serious fun.

All the programs in this book are available in disk form. Each game's object code (what you run, and what you'd normally create by typing in the MLX listing in Chapter 2) and complete source code (in single statement line PAL format) is included on the disk. You can order this $12.95 disk only through COMPUTE! Publications. Call toll-free 1-800-367-6767 (in NY 1-212-887-8525).

CUT-OFF!

Tom R. Halfhill

"CUT-OFF!" is a fast-paced, two-player game for the Commodore 64 and Commodore 128 in 64 mode. Ten levels of difficulty—ranging in speed from moderately slow to impossibly fast—make CUT-OFF! enjoyable for everyone.

Some computer games over the years have become classics. Usually they are simple in concept, yet universal in appeal, and general enough to be translated for almost any computer. Some good examples are *Pong*, the granddaddy of all videogames, *Breakout*, *Lunar Lander*, and the venerable *Space Invaders*. For legal reasons they may be disguised by different names, but there probably isn't a home computer anywhere for which some version of these all-time favorites isn't available.

Another classic game is *Blockade*. Again, it goes by different names (sometimes *Surround*), but the basic concept remains the same: Two players square off against each other by steering a moving line around the screen, each trying to head off the other, or to force the other player to crash into a wall or into his or her own trail. This concept dates back to the early days of videogames. In fact, the very first videogame I ever played was a *Blockade*-style game. It was during the mid-1970s, and a friend and I encountered the machine in a dimly lit cafe. By today's standards the game was downright primitive. No color, crude sound effects, and slow action. Yet we had never played anything like it before. (We thought it would never catch on, because it cost 25 cents per play at a time when a quarter bought you three plays on most pinball machines.)

Years later, the basic concept of *Blockade* was revived and updated in the 1982 film *TRON*. In this Walt Disney production, humans trapped inside a bizarre computer world were forced to become gladiators on "light cycles"—space-age motorcycles which left walls in their wakes, counterparts of the lengthening trails in *Blockade*.

Anyway, that's the story behind the latest incarnation of this popular game, now dubbed "CUT-OFF!" It preserves all the traditional concepts and includes color, sound, and the broad range of speed levels possible only in a program written entirely in machine language.

Starting the Game

Plug a joystick into each of the two joystick ports, then load CUT-OFF! by typing (make sure you've already entered **GO 64** if you're using the 128):

LOAD"*filename*",8,1 (for disk)
LOAD"*filename*",1,1 (for tape)

where *filename* is the name you gave to the program when you saved it to disk or tape from "MLX."

Note: If you forget to append the ,1 to the command, the program loads into the wrong area of memory and won't work.

Assuming the program load was successful, next enter

SYS 49152

The game screen appears instantly. (One of the best things about machine language is that you don't have to wait around for programs to initialize.)

The opening screen allows you to select a skill level ranging from 0 (the slowest speed, suitable for youngsters) to 9 (recommended for superhumans only). The skill levels are spaced equally apart, so you might want to start at 3 or 4. The level you select remains the same for the entire game. To change levels in the middle of a game, just press the RUN/STOP—RESTORE keys and restart the program with another SYS 49152. Of course, this cancels the game in progress.

To choose a skill level, move the joystick plugged into port 1 up or down. The number on the screen changes, and even wraps around if you go below 0 or above 9. To lock in your choice and begin the game, press the fire button of this joystick.

The game starts with the players aimed at each other head-on. Don't get the joysticks mixed up between players—joystick 1 belongs to the left-hand player, while joystick 2 should be used by the right-hand player. To steer, move the joystick up, down, right, or left. You can't move diagonally.

The joystick buttons also toggle a pause feature. To freeze the action, quickly press and release the button of either joystick. This leaves you free to answer the phone or take a break. To restart action, press and release either joystick button again.

Scoring and Winning

There are four ways you can crash: hitting a wall, running into the other player's trail, crossing your own trail, or backing into yourself by trying to reverse your direction.

After a crash, the surviving player is awarded points equal to the number of segments in the crashed player's trail. This means that the longer the players last before crashing, the more points are at stake. Thus, it's possible to catch up even if you're way behind.

Each time you crash, you lose one "life." Each player starts with ten lives, and the game ends when one player runs out. After each crash, the screen updates the score and reminds you how many lives each player has left. To restart each round, press the joystick fire button.

When the game is over, you get a chance to change the skill level for the next game. Just to get a peek at how fast machine language can be, try a game at level 9. You'll be lucky if you can make one turn before crashing into a wall. Yet even this level had to be slowed down with delay loops.

Typing

Refer to Chapter 2, page 51, for typing instructions for CUT-OFF! and its MLX listing.

Source Code

The source code for CUT-OFF! is in Chapter 3, beginning on page 134.

Nessie

Tom R. Halfhill
64 version by Charles Brannon

In "Nessie," a nonviolent action game, you're posing as a photographer trying to snap a clear photograph of the Loch Ness monster. You need a joystick to aim your camera.

For decades, fans and believers of Scotland's Loch Ness monster have affectionately referred to the mysterious creature as Nessie—hence the title of this game.

Inspired by a TV documentary on Loch Ness that recounted the hundreds of attempts to photograph the monster, this game tries to re-create the difficulties of capturing the creature on film. Whether in the game or in reality, it's not easy. Almost all of the photographic attempts have ended in failure—there exist only a few controversial photos showing parts of fins, shadowy shapes, and blurred figures. Maybe your steady hand and sharp eye will succeed where others have failed.

Starting Nessie

Load Nessie by entering (only after typing **GO 64** if you're using a Commodore 128):

LOAD"*filename*",8,1 (for disk)
LOAD"*filename*",1,1 (for tape)

where *filename* is the name you gave the program when you saved it to disk or tape from "MLX."

Make sure you append the ,1 to the command, or Nessie won't work.

To start Nessie, first type

NEW

once Nessie has loaded. This won't erase the program, but does reset some pointers that otherwise might give you an *?OUT OF MEMORY ERROR* or cause other problems with the game. Next, enter

SYS 49152

Instantly (isn't machine language nice?), the game screen appears, with a boxed-in crosshair in the center of the screen. There are also prompts to select game options. The keys to press for each are

f1 Telephoto
f3 Wide Angle
f5 Speed
f7 Easy Game
f8 Hard Game

There are two types of camera lenses: *Telephoto* and *Wide Angle.* By far the easiest is the wide angle, obtained by pressing the f3 key. A wide angle lens lets you cover more area from your camera position. You'll see the viewfinder expand when you press the f3 key. Since its larger size makes it easier to enclose Nessie, you may want to begin play with this.

The telephoto lens, which is the default selection (in other words, if you don't make a choice, this lens is used automatically), is about half as large, making it harder to catch Nessie within the border of the lens. In fact, the telephoto viewfinder barely frames Nessie. Press the f1 key to switch back to the telephoto lens if you previously selected the wide angle lens.

Function key f5 is used to increase or decrease the speed. The default speed is 5. Speed 1 is very slow, 9 almost too fast. Hitting the f5 key repeatedly cycles through the levels from 1 to 9.

Make sure your joystick is plugged into port 2—you're ready to begin. Press f7 for an easy (default) game or f8 for a more difficult level of play. (Pressing f7 during play restarts the game, something you might find handy.) If you choose the hard game, the playfield will be littered with black squares that interfere with your photography. It's better to begin with the easy version.

The photo session has started, and the clock is moving.

The Big Picture

At the top left of the screen is your camera's film counter, which shows how many pictures remain on your roll of film. You start with a 20-exposure roll. Each time you snap a picture, the film counter decrements. The current game speed also displays.

Your camera viewfinder starts in the center of the screen. You can move it in any direction with the joystick. Pressing the fire button releases the shutter. The viewfinder frame itself is blue, with a red aiming crosshair in the center. To take a properly centered photo, you must position the crosshair over Nessie. If any part of Nessie is touching the viewfinder frame when you snap the shutter, it registers as a cropped photo when the film is developed at the end of the game. A picture of a piece of Nessie is better than nothing, but it's not nearly as valuable as a photo of the whole monster. (Let's face it, wouldn't you feel better walking into the New York *Times* with an indisputable picture of Nessie instead of a doubtful snapshot of a dorsal fin?)

For the same reason, you must be careful not to include any other objects in the viewfinder while photographing Nessie. This isn't as easy as it sounds. When you start the game, you'll find that Loch Ness is alive with turtles and eels. If you photograph one of these instead, you've been "fooled"—and your photo is worth only a handful of points. (The reason you get any points at all is that you might be able to sell the photo to *Field & Stream* or an airline magazine.) The eels are particularly troublesome. They bear an uncanny resemblance to Nessie, which is why so many hopeful photographers over the years have been fooled.

Make sure your camera's viewfinder doesn't touch any text (such as the score line) or any of the black squares in the hard game, or you'll just end up with a cropped picture.

Another hazard to beware of is jittery hands. Nessie is not an easy target—the creature swims around the loch in random directions, staying still for only a moment before slipping away. Meanwhile, you're trying to center the monster in the viewfinder. If you snap the shutter while moving the finder, the picture will be blurred. And that's worth zero points.

Pressing any key (except for the CTRL, RESTORE, SHIFT, and Commodore keys) freezes the screen. Hitting another key restores motion. You can do this if you want to take a short break from the game. Or if you get particularly frustrated with Nessie's darting behavior, you can use this feature to "freeze" the loch and capture her on film. It's actually cheating, but....

When you get down to your last five pictures on the roll of film, the viewfinder frame automatically turns from blue to bright yellow as a warning. This is in case you're too busy to pay attention to the film counter.

Developing the Film

After you snap your last shot, the film instantly develops (machine language—faster than Polaroid) and is displayed. Each of the 20 finished prints shows what you photographed when you snapped the shutter. They are arranged in the order in which you shot them. (Programmers may want to note that each picture is a sprite. Using raster interrupts permits you to have up to 64 sprites simultaneously. In this program, we have 20 sprites displayed on one screen.) The frames show a whole Nessie, a cropped Nessie, a colored blur, another loch creature, or nothing, depending on your photographic skill. At the bottom of the screen is your final score, adjusted for the amount of time that elapsed.

To restart Nessie, just snap the shutter button. This returns you to the setup screen, where you can change lenses, if you wish, before playing again.

Pictures for Points

Since the telephoto lens is harder to use, it scores more points. A clear, properly framed photo of Nessie taken with the wide angle lens scores 2000 points, but a whopping 20,000 points if you used the telephoto lens. If Nessie isn't completely enclosed (cropped), the photo scores only 100 points (1000 with the telephoto). Snapping a picture of a turtle or eel means you've been fooled, and you receive only 50 points, or 500 if you were using the telephoto. Moving the camera when you press the shutter button gives you a blurred picture (shown by the horizontal lines across the developed print). You get no points for this, just as you receive no reward for taking an empty picture of the loch.

Another important factor in the scoring is time. One point (or ten points with the telephoto) is subtracted from your final score for every four seconds you took to shoot all your pictures. You have to balance your pace between reckless photography and careful shooting in order to get the best pictures in the least amount of time.

Anyone Can Use a Camera

Nessie is a game that can be enjoyed by almost anyone. Children especially will delight in capturing the creature on instant film. Setting the speed level allows you to slow down or speed up the creatures' movements. Slower speeds are appropriate for smaller children. Nessie is easy to snap, and there's none of the usual frustration with action games.

But set the speed to 8 or 9, the game selection to hard, and Nessie becomes a challenge to even the most agile photographer. You'll need a quick shutter and lots of luck to see the monster on your developed prints. In fact, it's almost as hard as getting the real Loch Ness monster on film.

Typing

Look to Chapter 2, page 56, for typing instructions for Nessie and its MLX listing.

Source Code

The source code for Nessie is in Chapter 3, starting on page 139.

Campaign Manager

Todd Heimarck

Campaign, advertise, poll regions, take stands on issues, and more, in this two-player national-election simulationfor the 64 and 128 (in 64 mode). The right strategy and a good candidate can lead you to the White House.

The Democratic delegates crowd the floor, wearing straw hats, carrying balloons and signs. The floor fights are done. The time has come to nominate.

"Maryland?"

"Mister Chairman—the great state of Maryland, the Free State, Home of the World Champion Baltimore Orioles, casts all of its votes for the senator from Arizona."

The chairman pounds his gavel. The din of cheers and jeers subsides.

Power Broker

The convention is deadlocked. And you control a large block of uncommitted delegates. It's all up to you.

The vice president from Rhode Island has good charisma and intelligence, but you know his health is poor. The reverend from Arkansas is attractive, but a bit conservative. Although the senator from Arizona is experienced, he's not very smart. Perhaps the New Jersey doctor? No, the Ohio senator has the best combination of personality and issues, plus you'll get a home region advantage in the populous Heartland.

Now it's the Republican's turn. Of the five choices, the woman from South Carolina is the best all-around candidate. She has high charisma and fund-raising appeal, which translates well into television ads.

It's time to hit the campaign trail.

Nine Weeks on the Trail

The Democratic senator starts with nine million dollars and 59 health points. He rests two days (to build up his health), then spends two days fund-raising. Campaign stops in Illinois and Texas sway the voters slightly to the Democratic side.

The Republican campaigns in her home state of South Carolina. She then moves on to North Carolina, Virginia, and Florida, followed by a couple of days resting.

As the campaign progresses, the Democrat concentrates on personal appearances in the industrial Northeast, plus forays into the larger states such as Texas, California, and Florida. The Republican candidate does less actual campaigning, preferring to spend more time on fund-raising to pay for her (expensive) television ads.

In the crucial eighth week, both candidates rest and fundraise in preparation for the last-minute campaigning. The Democrat does a media blitz in the Pacific, Southern, and Atlantic states. The Republican hits the Heartland, Arklatex, and the Urban Northeast.

Election Night

Initial returns from New England show the Republicans sweeping the region, but the large states of New York and Pennsylvania go Democratic. The Republicans win most states from Ohio to the Great Plains, but the Democrats pick up the Southern Atlantic states (except Florida). Texas votes for the GOP, while the rest of the region slides Democratic. The Rocky Mountain states are solid Republican. The Democrats take the Pacific States.

The final results show the Republicans winning six of nine regions and capturing the presidency, with 315 electoral votes to the Democrats' 223. Three of the four biggest states voted Democratic, but Ohio and Illinois (with 47 electoral votes between them) made the difference. The TV ads in the last week moved these two key states into the Republican camp.

Managing the Campaign

"Campaign Manager" pits your candidate against your opponent's. Each of you manages a campaign, from the nomination to the actual election. Make the right decisions and your candidate finds a home in the Oval Office. Make mistakes and you'll have to wait another four years.

To get the campaign on the road, load the game by entering

LOAD ***"filename",*8** (for disk)
LOAD ***"filename"*** (for tape)

where *filename* is the name you gave to the program when you saved it to disk or tape from "MLX" (it's assumed you've already typed **GO 64** on the Commodore 128).

Unlike some other machine language programs on the Commodore 64 and 128, Campaign Manager doesn't require the *,1* suffix. The game's first few bytes look like a BASIC program with the command SYS 2061. Fortunately, you don't have to remember the SYS—it's built into the program.

You have nine weeks to campaign. Each week, you plan your moves and enter them via the menu on the itinerary. You have two defensive moves, resting and fund-raising, and two ways to gain votes, campaigning (personal appearances) and advertising on television.

At the beginning of each turn you see a medium-resolution map of the U.S. which indicates which way each state is leaning. The MAP option allows you to move a cursor around the country to identify which states are which. If the Republicans are ahead, the state is red. Democratic states are cyan (light blue). If you're using a black-and-white television, the Republican states are the darker ones. You may notice that states occasionally switch back and forth, even though neither candidate campaigned or advertised there. This indicates that the voters in that state are split down the middle, and because of slight errors in polling, seem to be leaning one way or the other.

Since you have only 63 days (nine weeks of 7 days), you have enough time to campaign in each state once or twice. But in terms of electoral votes, California (with 47) is far more important than some of the smaller (three-vote) states like North Dakota or Vermont. Generally, it makes more sense to campaign more heavily in the ten biggest states, sometimes called "megastates."

State	**Electoral Votes**	**State**	**Electoral Votes**
CA	47	OH	23
NY	36	FL	21
TX	29	MI	20
PA	25	NJ	16
IL	24	NC	13

Winning the election requires 270 electoral votes (of a possible 538). The ten biggest states account for 254, just 16 short of a majority.

At the beginning of the campaign, each state has a large pool of undecided voters. As the game progresses, they make up their minds and the pool diminishes. It's possible, but unlikely, for all of a state's voters to decide before the end of the campaign. You would have to go to the state at least eight times before the undecided points were used up.

Each state has a built-in bias toward one party, based on past elections for president, senator, governor, and so on. The District of Columbia, for example, is staunchly Democratic, so the Democratic candidate will automatically get seven campaign points there, compared to a Republican's two.

Since the Republicans have won three of the last four elections (including a landslide victory in 1972), you might expect them to begin the game with a huge advantage. But if you look at nonpresidential elections, you'll find that a lot of states elect Democratic governors, senators, and representatives, then vote for a Republican president. And a lot of those basically Democratic states were split by third-party campaigns (Wallace in 1968, Anderson in 1980).

To even things up and make the game more playable, the Democrats begin with an electoral vote advantage of 282 to 256, although four of the megastates (Pennsylvania, Ohio, Florida, and North Carolina) are barely leaning to the Democratic side. The Republicans have the advantage of beginning with 29 of the 51 states (since the District of Columbia has three electoral votes, it counts as a state). Most of the states west of the Mississippi are Republican, while the Democrats have most of the industrial Northeast and the South.

In addition to the natural political leanings, each state believes certain things about five general issues:

- Unemployment/inflation
- Poverty/crime
- Agriculture
- Education
- Defense

The issues are based on things like census reports and almanacs. A very urban state might be conservative on crime, but not care much about agriculture, for example. Each candidate has certain stands on these issues. When you campaign or advertise in a state, you can get up to three extra campaign points for each issue, *if* your stand agrees with the voters'.

Finally, the candidate you choose has a campaign effectiveness rating based on charisma and intelligence. This factor translates to votes each time you campaign.

Choosing a Candidate

At the beginning of Campaign Manager, you choose which party will go first and decide if one candidate will be the incumbent. You might want to flip a coin, the winner choosing either a party or being the first or second player. In testing, we found that the second player has the slight advantage of making the last move. Being an incumbent gives you some extra campaigning strength and isn't recommended if you want an even game.

Note that all choices on the menu can be made with a joystick. (It's easier to play if each player has a joystick, although one can be shared. It doesn't matter which port it's plugged into.) Move the pointer with your joystick and press the fire button *twice* to make your choice. If you don't have a joystick, you can use the keyboard instead; I, J, K, and L for up, left, down, and right, respectively. Press M in place of the fire button.

Players then pick which candidate will represent their party. Five randomly chosen candidates are available. To the right of the candidate's stats is the YES/NO counter. Before making your choice, pick NO for each possibility until you have seen all five. They'll cycle around again, so you can make your choice. After selecting a candidate, enter his or her name from the keyboard.

The heart of the game is the actual campaign, but in some ways the convention is more important. Nominate a terrible candidate, and you'll spend most of your campaign trying to catch up.

Healthy, Wealthy, and Wise

A candidate's personality greatly affects the outcome of the election. In the lower-left corner you'll see a list of five attributes, each associated with a number from one (worst) to eight (best). With a couple of exceptions, the ideal candidate is the one with straight eights.

First is charisma (CHAR), which is personal magnetism, panache, the ability to influence and excite people. This is the most important personality trait because it's part of both campaign effectiveness and advertising effectiveness.

Stamina (STAM) rates your candidate's health. A candidate with low stamina will have to rest frequently to regain health and strength.

Intelligence (INTL) adds points to campaign effectiveness and last-minute campaigning.

Experience (EXPR) helps you with fund-raising. If your candidate has lots of experience, he or she has more contacts and connections for raising money. Since experience comes with age, it counts against your health, although stamina counts for more health points.

Appeal (APPL) also contributes to fund-raising appeals. But if you have maximum appeal (eight), you may be tainted by your affiliations with special-interest groups, and there's a backlash when you advertise. It's best to have an appeal of six or seven.

The candidates' attributes are generated by adding three random numbers, so candidates are more likely to have a middle number (four or five) than one of the extremes.

The personality traits translate into these five campaign factors:

Campaign effectiveness (CHAR*2 + INTL): the key factor in campaign stops.

Strength/Health (STAM*4 + 9 − EXPR): determines the effectiveness of a rest day.

Fund-raising appeal (EXPR*3 + APPL): determines how much money can be raised in a day.

TV ads (APPL OR 8 + CHAR): translates into votes when advertising.

Last-minute campaigning (INTL + STAM): wins last-minute votes to your side after the ninth week.

The significance of each factor is explained below in the section called "Getting Votes."

Here I Stand

Next to the personality factors are the candidate's stands on various issues. You see five issues, each with a sliding scale of one (at the far left, representing liberal) to six (conservative). A

Republican who wants to get tough on crime, for example, will have a rank of six. A Democrat who wants to solve the unemployment problem will have a rating of one.

Candidates will range from two to five on the issues of agriculture and education. On the other three issues, the Democrats will have stands from one to four, while the Republicans will go from three to six.

You'll generally get more votes with middle of the road beliefs. Look for a candidate with twos or threes if you're the Democrat. Fours and fives are best for the Republican. The exception is agriculture and education, where you do best with a three or a four.

Common sense tells you which issues are important in most states. Agriculture is a major issue in the farming states. Your stand on defense makes a difference in states with a lot of military-related industry, like Washington state, Utah, and Maine (the formula was obtained by dividing military contract spending by population, for a per capita expense).

The candidate's personality is generally more crucial than the stands on issues. If you have a lot of charisma, intelligence, and appeal, it doesn't matter if you have radical views on one or two issues.

If you have five very bad candidates, press RUN/STOP–RESTORE and try again. It's not much fun running a doomed campaign (ask Gary Hart, campaign manager for George McGovern in 1972).

Beginning the Campaign

After the nominees have been chosen, the first week begins. You may notice that some states have changed colors. That's because each nominee gets the equivalent of campaigning once in each state. (Some voters make up their minds before the campaign even starts.) If one candidate is much more charismatic or happens to hit the right issues, a state may jump over to his or her side. In addition, each gets a home state and home region advantage.

You should develop a strategy. If your appeal and charisma are strong, you might want to concentrate on television ads. If your candidate has a strong anticrime stance, visit the more urban states. At the very least, you should plan to visit each of the megastates.

You begin in your home state; it's traditional to campaign there once. And the first week usually means some fund-raising and resting, as purely defensive moves.

The Doctor, Treasurer, Pollster, and Pilot

Under the week's itinerary should be two numbers. At the beginning of each week, your treasurer tells you how much money you have, up to about $25 million. Your personal physician figures out how healthy you are. At most you'll have 255 health points.

If you fall below four million dollars at any time during the week, television advertising will be useless. If you have less than one million, you won't be able to pay the pollster (you won't even see the bar graph to the left of the map). When your bank account falls to zero, the campaign is paralyzed until you sponsor a fund-raiser. You can't even afford to pay your doctor or staff.

It takes time away from campaigning, but you have to raise money once in a while. Each fund-raising point (EXPR*3 + APPL) is worth $200,000.

Campaigning is exhausting, and even though it takes time, you'll occasionally have to rest. When you decide to catch some Z's, the itinerary will be filled with (you guessed it) Z's. Each day of rest adds double your strength factor, plus campaign effectiveness, plus the number of states you're winning, to the health you have. A high campaign effectiveness gives you optimism; you rest better. If you're behind, you worry and toss and turn. Resting two days in a row gets you 16 extra health points.

There are two reasons to keep your health up. First, when you campaign in a state, you get an extra campaign point for every 32 health points you possess. Second, if your health falls below 8, you look haggard and falter over the simplest speeches; campaigning does you no good.

The treasurer counts dollars, the doctor counts your health, and your pollster counts votes.

The pollster does three things. First, you get a bar chart that shows how many electoral votes would go to the Democrats and Republicans if the election were held today. You can see it to the left of the map. The gray bar marked *U* represents undecided states too close to call. Second, you have a map of the U.S. to show at a glance which way each state is leaning. Republican states are red, Democratic states blue. These first two services are part of the pollster's contract and cost you nothing. Of course, if your money drops lower than one million dollars, you have to stop paying the pollster; all you get is the map.

The third service is the most important—regional polls. To get a poll of all states in a region, move to POLL on the main menu and press the fire button twice. You'll see a bar chart showing which way each state in the region is leaning, from one (half a character wide) to four (two characters). The poll reflects the political situation at the beginning of the week; whatever campaigning you have planned for the week isn't included. A state with a thin bar can usually be taken with a single campaign stop.

Don't use polls in the first couple of weeks—most states start out fairly even, and you won't learn much. But polling can be a powerful tool toward the end of the game. If New York is firmly committed to you, forget about further efforts in that state. And if you find a whole region weakly supporting your opponent, you can hit them with TV ads and score a few dozen electoral votes.

Regional polls cost $100,000 and are not available if you begin the week with less than a million dollars.

The final character (although transparent) in your entourage is the jet pilot. Your jet can carry you on short hops within a region for almost nothing. But if you travel to a new region, you'll shell out $100,000 for fuel and maintenance. As long as you're in a region, you might as well stay there a few days to avoid a lot of travel expenses. Again, you don't actually move to a new region until you've campaigned in one of the states. You can use the travel option to conduct regional polls; you'll pay $100,000 for the poll and another $100,000 if you decide to campaign in a region. Don't campaign and you won't be charged for travel.

Guests and Fish

Benjamin Franklin once said that after three days, guests and fish begin to smell. The same principle applies to campaigning.

Campaign once and you gain some votes. Stay for a second day and the voters of a state are flattered; you gain a couple of bonus votes. But stick around for a third or fourth day, and you've overstayed your welcome.

Getting Votes

Each state begins with 255 undecided voter points. Your main goal is to use campaigning and television advertising to sway the undecided. And you have to maintain your health and money.

The effects of a personal appearance can vary. You get up to 3 points for each issue (if the state agrees with you), 1 point for every 32 health points, and up to 24 for your campaign effectiveness (intelligence plus double charisma), plus a 2-point bonus if it's your second day in the state.

If your money is down to zero, you get no campaign points. If your health is below eight, you get a single vote.

Each campaign stop decreases your health and money. It's possible to run out in the middle of the week, making each succeeding visit ineffective until you rest or raise money. Let's say you go to Connecticut and impress 23 of the 255 undecideds. The pool of available voters is reduced by that number. Half of 23 (11 points) is charged against your health. Half again (5 points) times $100,000 is subtracted from your money. In addition, each state has some people who don't agree with you, so a quarter of your total (5 points) goes to your opponent as a reaction against your speech. If you had previously been in a different region, travel expenses of $100,000 are subtracted.

Television advertising is a little different. It affects every state in the region and quickly swings voters to your side. To advertise, first travel to the region and, before doing any ads, make at least one campaign stop to establish your presence. After campaigning once, you can advertise as much as you like.

Unlike resting and campaigning, the effects of advertising do not accumulate from day to day. If you advertise two days in a row, you don't get bonus points. Advertising does grow in strength from week to week, however, and will be more effective toward the end of the campaign.

If you flood the region with ads, it's possible to bring a whole section of the country to your side. But it *is* costly. In each state, advertising credits you with half your campaign effectiveness, half your TV ads' effectiveness rating, points for issues, plus two times the week number (in week 7, for example, you get 14 extra campaign points).

The cost is the usual one-fourth of campaign points gained, plus double the TV ads' effectiveness. The large regions can cost a lot. Going on TV in the Atlantic States (all nine) or in the Rocky Mountains (eight states) can easily deplete your treasury.

On the day you plan to advertise, you must have at least four million dollars. If you don't, you waste the day and gather no new votes. So, if you begin the week with five million, and campaign in six states, it's likely you'll have less than four million by Saturday. Your ad campaign will do you no good.

There's one more item you can choose: RECONSIDER. If you make a mistake, this option wipes your itinerary clean so that you can start the week anew. Your choices are not permanent until you fill out the seventh day and press the fire button. (If you pull on the joystick instead, it's the same as reconsidering.)

Here's a summary of the commands on the main menu:

- CAMPAIGN—allows you to make a personal appearance in one of the states of the region you're visiting. Results depend on campaign effectiveness, built-in party bias of the state, health, and issues. Doesn't work if you have zero health or money, or if all undecided voters have been claimed. Gains votes, costs health and money.
- TV ADS—blankets the region with advertising. Reduces health and costs a lot of money, but can quickly deliver a big chunk of votes. Net votes based on TV advertising effectiveness, campaign effectiveness, and issues. Doesn't work if you have less than four million dollars.
- FUNDRAIS—raises money for your campaign based on fund-raising ability. Takes a day, gains no votes, costs nothing.
- REST—builds up your health points, according to strength factor. Extra points if you rest two days in a row. Gains no new votes, costs nothing.
- MAP—moves the cursor around the map, prints the state name, electoral votes, and region number. For information only, costs nothing.
- POLL—provides a bar graph showing which way the states in the region are leaning. Costs $100,000 (immediately). Not available if money falls below one million dollars.
- RECONSIDER—erases the week's itinerary if you make a mistake.

- TRAVEL—takes you to a new region of the country. Costs $100,000 (not charged to you until you actually campaign there).

The Home Stretch

The ninth week is usually the most hectic. If you sponsored some fund-raisers in week 8, you'll want to spend a lot on TV advertising in the regions where you have a chance. Polls can tell you which states are most vulnerable.

After both candidates have finished their last week of campaigning, a couple of things happen. The last region to be visited by a candidate gives that candidate a few extra votes. And the last-week routine goes into action, as all the undecided voters make up their minds. Both candidates get their last-minute campaigning points (intelligence plus stamina) added to each state in the country. The undecided voters are split between the candidates and ties are resolved (based on the built-in bias to one party or the other).

The map is drawn for the final time. The final bar chart appears to the left (which should indicate at a glance which candidate won). Beginning with region 1 (New England), the electoral votes are displayed, with region totals below.

The winner is the candidate with the most electoral votes. There is a slight chance that there will be a tie, in which case you'd have to flip a coin.

Campaign Etiquette

Here are a few traditional rules of Campaign Manager etiquette:

First, since the joystick routine reads both joysticks, try to avoid interfering with your opponent's choices. Put down your joystick when it's not your turn.

Second, when you have filled out your itinerary and the prompt PRESS FIRE BUTTON TO CONTINUE appears, don't press the fire button. Let your opponent study what moves you made. Then he or she can press the fire button.

Third, since polls cost money, they should be kept private. When the other player is taking a poll, look away from the screen.

On to the White House

Campaign Manager is a true simulation. It accurately represents different candidate's strengths and weaknesses, campaign strategies and tactics, and allows for that bane of all politicians, a bit of luck. You'll find the game entertaining and highly educational as well. We all can't run for the presidency. But with Campaign Manager, you can at least see how you *would* have done, given the chance.

Typing

Refer to Chapter 2, page 64, for Campaign Manager's typing instructions and MLX listing.

Source Code

Campaign Manager's source code can be found in Chapter 3, beginning on page 147.

Heat Seeker

Jeff Wolverton
64 version by Tim Victor

Your jet climbs upward to avoid the missile, then dives for the ground. But it's still on your tail. You can't shake a heat seeker. A fast-action game for the Commodore 64 or Commodore 128 (in 64 mode). Joystick required.

Heat-seeking missiles are dangerous. They sense the heat from your jet engine and home in on you. They'll catch you, too—they're faster than any jet.

Your assignment—eliminate the heat-seeker base. It's easy enough to strafe the missiles on the ground, but if any are launched, you'll have to take evasive action.

Piloting the Jet

You can load and run "Heat Seeker" as you would any BASIC program since an enabling SYS command is built in. Simply enter

LOAD "HEATSEEKER",8 (for disk)
LOAD "HEATSEEKER" (for tape)

(Use the filename you saved the program with in place of HEATSEEKER.)
Press RETURN, then type

RUN

and press RETURN again.

Use the joystick to control the movement of the plane. The controls may seem a little confusing at first. You pull back to loop upward (counterclockwise) and push forward to loop down (clockwise), like in a real airplane. The jet moves at a constant velocity—you can't speed up or slow down. Press the fire button to launch a missile at the heat seekers on the ground.

If you do manage to eliminate all the heat seekers, you get to start all over again, this time with a new group of heat seekers. You have eight jets to work with—the number remaining is displayed on the screen, next to the score. To pause the game, press the SHIFT LOCK key.

Fast

The jets and missiles in Heat Seeker are sprites, making for smooth movement. Written entirely in machine language, the game is *fast.*

Though you can fire at heat seekers on the ground, it does no good to fire at a moving heat seeker. They're equipped with an improved electronic evasion (IEE) circuit which makes them impossible to hit. The only way to get rid of a seeker is to make it crash into the ground. When you're being pursued, dive for the ground and pull up at the last second. Seekers are faster, but they can't turn as quickly.

Don't worry about errant rounds you've fired—your plane can't shoot itself. If your jet is destroyed, all the heat seekers reappear.

A two-player mode is also available. It's not competitive: The players take turns flying the plane, trying for the highest possible score. The game reads both joystick ports, so if you're using two joysticks, the inactive player should put down the joystick to avoid interfering.

There are three levels of difficulty: Novice, Intermediate, and Expert. The higher levels have faster action and tighter curves. A flight-time bonus of ten points is awarded every few seconds, just for staying in the air.

Typing

Chapter 2 contains the typing instructions and "MLX" listing for Heat Seeker. Refer to page 84.

Source Code

Heat Seeker's source code begins on page 173.

Laser Beam

Mike Greenfield
64 version by Kevin Mykytyn

This fast-paced arcade game challenges the quickest reflexes. You'll need a joystick to shoot the laser, dodge the bouncing balls, and make it back to the safe zone.

"Laser Beam" is a fast-action, arcade-style game that, like almost every good game, sounds simple in the explaining but is far from easy in its play. Render the dangerous balls harmless with your laser, snatch the now-touchable ball while avoiding all the others, deposit that ball in the basket, and (whew!) make it back to the safe zone.

The Highlight Zone

Plug a joystick into port 2 and type

```
LOAD"LASER BEAM",8,1    (for disk)
LOAD"LASER BEAM",1,1    (for tape)
```

assuming that you saved the program with the filename LASER BEAM, of course. Next, type

```
SYS 49152
```

and press RETURN.

Move the joystick up or down to select the difficulty level, which ranges from 0 (slowest) to 9 (impossibly fast). The game starts when you press the joystick's fire button.

You'll immediately see seven colored balls bouncing around the screen. At the left is a highlighted *safe zone* occupied by a small stick figure—that's you. By moving the joystick, you can maneuver your figure around the screen. But if you venture out of the safe zone and bump into a bouncing ball, you're zapped. (You get three lives per game, as indicated by the figures at the bottom of the screen.)

Your goal is to render the balls harmless, grab them one by one, and drop them into the basket in the lower-right corner. To make a ball safe to touch, you have to shoot it with the laser gun. The laser is visible along the edge of the screen.

To control it, first move your figure into the uppermost corner of the safe zone. *The laser gun is under your control only when your figure is at this spot.* Pushing the joystick to the right moves the laser clockwise around the edge of the screen, and pushing the joystick to the left moves the laser counterclockwise. Press the fire button to activate the beam. If you push the joystick in any other direction, you'll move your figure away from the top of the safe zone, and the laser gun will no longer be under your control.

You can't shoot just *any* bouncing ball with the laser to make it safe to grab. You have to shoot the ball which matches the border color of the screen. As soon as you hit the ball, it turns white. Then you can maneuver your figure out of the safe zone, grab the white ball by touching it, carry it to the basket, and drop it in by pressing the fire button. Afterward, you must scurry back to the safe zone before a collision with another ball.

Let's say the border color is red. First, you move your figure to the top of the safe zone to take control of the laser gun. Next, you push the joystick right or left to aim the laser at the red ball. When you have a clear shot, press the fire button to shoot the beam. If you score a hit, the red ball turns white. Then you can push the joystick in another direction to move your figure out of the safe zone. Grab the ball, stuff it into the basket by pressing the fire button, and make your escape—all while avoiding the other bouncing balls. If you succeed, the border color changes to correspond to one of the remaining balls.

You continue with the process until all the balls are safely dropped into the basket. Then another round begins.

Bouncing Chaos

Sounds simple, right? Well, it's not. There are a few complications. Suppose you fire the laser and hit a bouncing ball that *doesn't* match the border color. It turns white, too. But it *isn't* safe to grab. If you touch it, you're zapped. This becomes a real problem when you accidentally shoot several of the balls and turn them white. Only one of them is safe, and you have to remember which one. It's not easy when three or more white balls are bopping all over the place.

There's an incentive for creating this chaos, however. The number of points you get for dropping a ball in the basket doubles for each white ball on the screen. If the only white ball is the one you're grabbing, you get only 5 points. If a second ball is white, you get 10 points; if a third ball is white, you get 20 points; and so on. If all seven balls are white when you drop the first one into the basket, you score 320 points.

Another complication is that your safe zone doesn't always stay safe for very long. After you shoot the ball that matches the border color, it begins shrinking from the bottom up. If you don't hurry out, you'll get zapped.

And there's yet another reason to move quickly: If you finish a round by dropping every ball into the basket before the horizontal bar at the bottom of the screen disappears, you get a 50-point bonus.

If all this action causes your brain to momentarily suffer a system crash, you can freeze everything by pressing and holding the SHIFT key. Press SHIFT LOCK to freeze the game for extended periods. Release SHIFT to resume play.

Typing

Refer to Chapter 2, page 95, for Laser Beam's typing instructions and "MLX" listing.

Source Code

The source code for Laser Beam can be found in Chapter 3, page 188.

Whirlybird

Philip I. Nelson

This hyperactive bird lays some rather unusual eggs. A quick-reflex action game for the Commodore 64 and 128 (in 64 mode). You'll need a joystick.

"Whirlybird" is a fast, updated version of the classic game *Breakout.* You control the Whirlybird, who flies back and forth at the top of the screen, spinning continuously. The goal is to clear out the layers of colored bricks at the bottom by bouncing eggs against them. When an egg moves back in your direction, move the Whirlybird into its path to bounce it back down. The game ends when you clear the field of bricks or run out of eggs.

Starting Up

Plug a joystick into port 2, then load and run Whirlybird as if it were a BASIC program. *Don't try to start the program with the SYS command.*

To load the game, type

LOAD ***"filename",*8** (for disk)
LOAD ***"filename"*** (for tape)

where *filename* is the name you used when you saved the game from "MLX." Type RUN.

The playfield appears immediately, with the Whirlybird twirling across the top of the screen. On either edge is a sparkling row of sidebeams and below are multicolored rows of bricks. At the upper right is the number of eggs you'll have in this game. Though the usual number is 10, you can get a smaller or larger number (up to 255) by pushing the joystick backward or forward. (Don't select zero—you'll simply return to the startup screen.)

Now pick the skill level. Press the f1 key to play a normal game or the f3 key for an expert game. At the expert level, the Whirlybird is only half its normal size, making it harder to hit the eggs. After choosing the level, the prompts disappear and the game begins.

Lay an Egg

Press the fire button to release the first egg. You earn 1 point every time you hit a brick, and 10 points every time you bounce an egg back down with the Whirlybird. Letting the egg fly past you into space subtracts 10 points from your score (unless it's already under 10). When that happens, the sidebeams sparkle continuously until you press the fire button again to release the next egg. Play continues until you lose all your eggs or clear the field of bricks. A 50-point bonus is awarded for clearing every brick from the screen.

In addition to normal bricks, the playfield contains a few round ones. When the egg hits one of these, the Whirlybird swoops down to deposit a sidebeam somewhere above the playfield. This doesn't affect your score—you get a free egg as well (press the fire button to release it). However, the extra sidebeams add an element of uncertainty since the egg rebounds sideways from them (not up and down), and they may or may not disappear when struck.

Whirlybird displays your current score as well as the highest score attained during the current session (which may include more than one game). When a game ends, both scores are displayed until you release the first egg in the next game, after which the score is cleared to zero. Since the high score is stored within the program code, you can record it for future comparison by resaving the game after each session ends. (It's a good idea to resave it with a different filename—like BIRDHIGH—to distinguish it from the original copy, which then serves as a backup.) The next time you load and run Whirlybird, it displays the previous highest score.

The Joystick Accelerator

It may take you some time and practice to master the joystick motion in Whirlybird. Rather than zooming immediately to the spot you want, the bird moves in an elastic manner, simulating the inertia and momentum of a natural object. It takes a bit of pushing to overcome the bird's inertia and get it moving. The longer you push in one direction, the more speed and momentum you gain (up to a point). After you let up on the stick, the bird slows down and eventually stops. Hitting the sidebeam reverses your direction immediately. Reversing direction in midscreen may take a long or short time depending on your speed.

Think of the joystick as a gas pedal in a car. Keeping your foot on the accelerator makes the car move faster and faster. Let up, and the car gradually slows down. Learning to control the acceleration takes time. (You may find it more effective to move the Whirlybird with short taps on the joystick, rather than with long periods of acceleration.)

Egg Juggling

The direction in which an egg bounces when it hits the Whirlybird depends on which way you're pressing the joystick. If you're not pressing it left or right, the egg rebounds as if the bird were a solid wall. If you're pressing the stick left, the egg bounces to the left; if you're pressing right, it bounces to the right. This lets you control the direction of every bounce. But don't move the stick so much that you miss the egg completely.

The result of a bird/egg collision also depends on what part of the bird you hit. The safest strategy is to hit the egg with the bird's feet. Bouncing it off a wing is more dangerous, since the bird never stops turning. If the egg arrives when the wing is outspread, you'll usually get a normal collision. But if the Whirlybird happens to be facing sideways—making it very slim in profile—the egg may miss completely.

Occasionally, a wing collision causes the bird to juggle the egg briefly. Juggling gives you extra points and may also change the angle of rebound slightly (a boon when you're short on eggs and need to clear the last few bricks). But it's also dangerous, since the bird may juggle the egg upward and out of play. Juggling occurs less frequently at the expert level; with smaller wings it's more difficult to get multiple collisions.

Rings and Filters

Whirlybird's unusual sound effects are created with two voices of the computer's sound chip. Voice 3 is used as a random number generator (to pick a random screen position and direction for each new egg) and doesn't produce any sound. Voice 1, set for a pulse waveform, makes a *boing* sound when an egg bounces, and voice 2 creates low, thrumming tones in the background with a ring-modulated triangle wave. The vibrato-like effect results from ring-modulated overtones that slowly pass in and out of phase with the frequencies of voices 1 and 2.

These sounds are made more interesting by passing them through a combined low-pass/band-pass filter and sweeping the filter's cutoff frequency up or down in conjunction with other game events. When you start the game or when the egg is flying freely, the cutoff frequency sweeps up and down, creating a gradual *meow-meow* effect. When the egg hits something, the cutoff frequency is set high and swept rapidly downward to accentuate the bouncing noise.

Typing

Refer to Chapter 2, page 105, for Whirlybird's typing instructions and MLX listing.

Source Code

Whirlybird's source code can be found in Chapter 3, starting on page 199.

Miami Ice

Jeff Kulczycki
64 version by Kevin Mykytyn

Forget the flamingos, the sun, the powerboats, and the designer clothes. "Miami Ice" doesn't have a smuggler within miles. What it does have is ice—lots and lots of slick ice under your wheels. This action game challenges both your driving skills and powers of concentration. Joystick required.

Ah, Miami—sun city of the South. A sparkling metropolis blessed with a tropical climate, palm trees, beaches, revived art deco architecture, stylish pastels, and classy elegance. Almost paradise.

You wake up on another bright, sunny Miami morning, sip a glass of freshly squeezed orange juice, don your white linen suit and sunglasses, and stroll outside—then get the shock of your life.

What's going on here? Overnight, a freak shift in the jet stream has piped a blistering cold front down from Ohio. The weather forecaster had predicted a brief shower last evening, but that's not what happened. Instead, the Florida peninsula was blasted by the worst ice storm in 400 years. The Everglades are frozen solid. The pink flamingos are blue. And the streets of Miami are coated with a shimmering layer of slippery ice.

As you start your car—the pampered engine coughs and sputters in the bitter cold—you wonder what it's going to be like driving to work. A Miami native, you've never driven on ice before. In fact, you've never even *seen* this much ice since your boss's retirement party last year, when the caterers made that life-size ice sculpture of Ponce de Leon. You've heard the horror stories told by tourists about winter driving conditions up North, but never thought it could happen to you—not here, in Miami.

The minute you pull out onto the street, your worst fears come true. When you step on the gas pedal, the wheels spin and the car accelerates sluggishly. When you turn the steering wheel, the car slides all over the road. And when you step on the brakes—well, forget it.

You realize, desperately, that you've got to make it to the parking garage across town without smashing your car to smithereeens. It won't be easy. But at least there's one thing in your favor—you've got the whole road to yourself. Everyone else, it seems, had the good sense to stay home.

Out of Control

Use the joystick plugged into port 1 to maneuver your car across ice-covered Miami streets and reach the safety of a garage. The joystick button is the gas pedal, and pushing the stick right or left steers the car in the corresponding direction.

Here's the twist—the car doesn't respond instantly to your commands. It tends to slide in the same direction even after you've steered it toward another direction. Then, when you try to recover, you often overcorrect and start sliding in yet another new direction. It's an inertial nightmare—much like real winter driving.

When you hit a guardrail or some other obstruction, your car cracks up. You get three cars per game. To steer your car safely into the parking garage and advance to the next screen, enter the front of the garage without bumping into the black lines which mark its three walls.

But the game's not over even if you reach the safety of the garage. Instead, you advance to another screen whose streets are even harder to navigate.

The number of points you score depends on how soon you reach the garage. As an incentive to recklessness, a timer starts counting down when you begin each new screen. If you reach the garage, you score the number of points left on the timer. If the timer runs out, you can still reach the garage, but you won't get any points. However, you will advance to the next screen. Indicators on the screen show the timer value and your current score.

Hands at Ten and Two

Though Miami Ice is written in machine language, you start the program as if it were written in BASIC. To get a grip on the steering wheel of your car, load the program by typing

LOAD"*filename*",8 (for disk)
LOAD"*filename*" (for tape)

Press RETURN, then type

RUN

and press RETURN again.

Make sure you've plugged a joystick into port 1.

Miami Ice includes seven screens. The game normally starts at screen 1, but you can begin a new game at any screen by moving the joystick up or down to change the screen number. This lets you skip the easier screens as you become a better player or peek at the hardest screens while you're still a beginner.

Typing

Look to Chapter 2, page 114, for Miami Ice's typing instructions and "MLX" listing.

Source Code

You'll find that the source code for Miami Ice begins on page 210 in Chapter 3.

Chapter 2
Type It In

MLX
Machine Language Entry Program

Ottis R. Cowper

"MLX" is a labor-saving utility that allows almost fail-safe entry of machine language programs.

Type in and save at least two copies of "MLX" (you'll want to use it to enter future Commodore 64 machine language programs from *COMPUTE!* magazine, *COMPUTE!'s Gazette,* and COMPUTE! books). If you're using a Commodore 128, you can still use MLX—you must enter it and any ML programs in 64 mode, however.

When you're ready to enter an ML program, load and run MLX. It asks you for a starting address and an ending address. These addresses appear in the short article accompanying the MLX-format listing of each game. You'll find the listings in Chapter 2.

If you're unfamiliar with machine language, the addresses (and all other values you enter in MLX) may appear strange. Instead of the usual decimal numbers you're accustomed to, these numbers are in *hexadecimal*—a base 16 numbering system commonly used by ML programmers. Hexadecimal—*hex* for short—includes the numerals 0–9 and the letters A–F. But don't worry—even if you know nothing about ML or hex, you should have no trouble using MLX.

After you enter the starting and ending addresses, you'll be offered the option of clearing the workspace. Choose this option if you're starting to enter a new listing. If you're continuing a listing that's partially typed from a previous session, don't choose this option.

A functions menu appears. The first option in the menu is ENTER DATA. If you're just starting to type in a program, pick this. Press the E key, and type the first number in the first line of the program listing. If you've already typed in part of a program, type the line number where you left off typing at the end of the previous session (be sure to load the partially completed program before you resume entry). In any case, make sure the address you enter corresponds to the address of a line in the listing you are entering. Otherwise, you'll be unable to enter the data correctly. If you press E by mistake, you can return to the command menu by pressing RETURN alone when asked for the address. (You can get back to the menu from most options by pressing RETURN with no other input.)

Entering a Listing

Once you're in Enter mode, MLX prints the address for each program line for you. You then type in all nine numbers on that line, beginning with the first two-digit number after the colon (:). Each line represents eight data bytes and a checksum. Although an MLX-format listing appears similar to the "hex dump" listings from a machine language monitor program, the extra checksum number on the end allows MLX to check your typing.

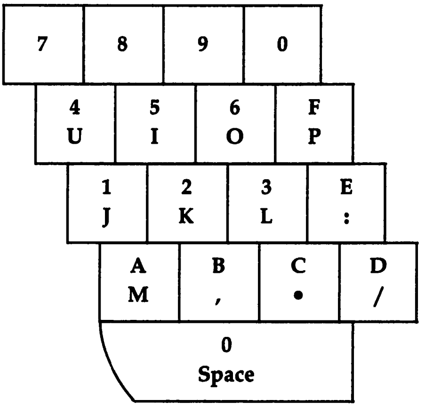
When you enter a line, MLX recalculates the checksum from the eight bytes and the address, and compares this value with the number from the ninth column. If the values match, you'll hear a bell tone, the data will be added to the workspace area, and the prompt for the next line of data will appear. But if MLX detects a typing error, you'll hear a low buzz and see an error message. The line will then be redisplayed for editing.

Invalid Characters Banned

Only a few keys are active while you're entering data, so you may have to unlearn some habits. You *do not* type spaces between the columns; MLX automatically inserts these for you. You *do not* press RETURN after typing the last number in a line; MLX automatically enters and checks the line after you type the last digit.

Only the numerals 0–9 and the letters A–F can be typed in. If you press any other key (with some exceptions noted below), you'll hear a warning buzz. To simplify typing, a keypad is available. The keypad—part of the keyboard—is active only while you're entering data. Addresses must be entered with the normal letter and number keys. The figure below shows the keypad configuration.

64 MLX Keypad



MLX checks for transposed characters. If you're supposed to type in A0 and enter 0A instead, MLX will catch your mistake. There is one error that can slip past MLX: Because of the checksum formula used, MLX won't notice if you accidentally type FF in place of 00, and vice versa. And there's a very slim chance that you could garble a line and still end up with a combination of characters that adds up to the proper checksum. However, these mistakes should not occur if you take reasonable care while entering data.

Editing Features

To correct typing mistakes before finishing a line, use the INST/DEL key to delete the character to the left of the cursor. (The cursor-left key also deletes.) If you really mess up a line, press CLR/HOME to restart the line. The RETURN key is also active, but only before any data is typed on a line. Pressing RETURN at this point returns you to the command menu. After you type a character of data, MLX disables RETURN until the cursor returns to the start of a line. Remember, you can press CLR/HOME to get to a line number prompt quickly.

More editing features are available when you're correcting lines in which MLX has detected an error. To make corrections in a line that MLX has redisplayed for editing, compare the line on the screen with the one printed in the listing, then move the cursor to the mistake and type the correct key. The cursor-left and -right keys provide the normal cursor controls. (The INST/DEL key now works as an alternative cursor-left key.) You cannot move left beyond the first character in the line. If you try to move beyond the rightmost character, you'll reenter the line. During editing, RETURN is active; pressing it tells MLX to recheck the line. You can press the CLR/HOME key to clear the entire line if you want to start from scratch or if you want to get to a line number prompt to use RETURN to get back to the menu.

Display Data

The second menu choice, DISPLAY DATA, examines memory and shows the contents in the same format as the program listing (including the checksum). When you press D, MLX asks you for a starting address. Be sure that the starting address you give corresponds to a line number in the listing. Otherwise, the checksum display will be meaningless. MLX displays program lines until it reaches the end of the program, at which point the menu is redisplayed. You can pause the display by pressing the space bar. (MLX finishes printing the current line before halting.) Press the space bar again to restart the display. To break out of the display and get back to the menu before the ending address is reached, press RETURN.

Other Menu Options

Two more menu selections let you save programs and load them back into the computer. These are SAVE FILE and LOAD FILE; their operation is quite straightforward. When you press S or L, MLX asks you for the filename. You'll then be asked to press either D or T to select disk or tape.

You'll notice the disk drive starting and stopping several times during a load or save. Don't panic; this is normal behavior. MLX opens and reads from or writes to the file instead of using the usual LOAD and SAVE commands. Disk users should also note that the drive prefix 0: is automatically added to the filename (line 750), so it should *not* be included when you enter the name. This also precludes the use of @ for Save-with-Replace, so remember to give each version you save a different name.

Remember that MLX saves the entire workspace area from the starting address to the ending address, so the save or load may take longer than you might expect if you've entered only a small amount of data from a long listing. When saving a partially completed listing, make sure to note the address where you stopped typing so that you'll know where to resume entry when you reload.

MLX reports the standard disk or tape error messages if any problems are detected during the save or load. (Tape users should bear in mind that Commodore computers are never able to detect errors during a save to tape.) MLX also has three special load error messages: INCORRECT STARTING ADDRESS, which means the file you're trying to load does not have the starting address you specified when you ran MLX; LOAD ENDED AT *address*, which means the file you're trying to load ends before the ending address you specified when you started MLX; and TRUNCATED AT ENDING ADDRESS, which means the file you're trying to load extends beyond the ending address you specified when you started MLX. If you see one of these messages and feel certain that you've loaded the right file, exit and rerun MLX, being careful to enter the correct starting and ending addresses.

The QUIT menu option has the obvious effect—it stops MLX and enters BASIC. The RUN/STOP key is disabled, so the Q option lets you exit the program without turning off the computer. (Of course, RUN/STOP–RESTORE also gets you out.) You'll be asked for verification; press Y to exit to BASIC or any other key to return to the menu. After quitting, you can type RUN again and reenter MLX without losing your data, as long as you don't use the clear workspace option.

The Finished Product

When you've finished typing all the data for an ML program and saved your work, you're ready to see the results. The instructions for loading and using the finished product vary from program to program. Some ML programs are designed to be loaded and run like BASIC programs, so all you need to type is LOAD *"filename"*,8 for disk or LOAD *"filename"* for tape, and then RUN. Such programs will usually have a starting address of 0801. Other programs must be reloaded to specific addresses with a command such as LOAD *"filename"*,8,1 for disk or LOAD *"filename"*,1,1 for tape, then started with a SYS to a particular memory address. On the Commodore 64, the most common starting address for such programs is 49152, which corresponds to MLX address C000. In either case, refer to the general description for each game—you'll find these in Chapter 1.

An Ounce of Prevention

By the time you finish typing in the data for a long ML program, you may have several hours invested in the project. Don't take chances—read Appendix A, "How to Type In Programs," use our "Automatic Proofreader" to type in MLX, and then test your copy *thoroughly* before first using it to enter any significant amount of data. Make sure all the menu options work as they should. Enter fragments of the program starting at several different addresses, then use the Display option to verify that the data has been entered correctly. And be sure to test the Save and Load options several times to insure that you can recall your work from disk or tape. Don't let a simple typing error in the new MLX cost you several nights of hard work.

MLX

Read Appendix A before entering program. For error-free entry, use "The Automatic Proofreader" (Appendix B) to type in this program.

```
SS 1Ø REM VERSION 1.1: LINES 83Ø,95Ø MODIFIED, LIN
      ES 485-487 ADDED
EK 1ØØ POKE 56,5Ø:CLR:DIM IN$,I,J,A,B,A$,B$,A(7),N
       $
DM 11Ø C4=48:C6=16:C7=7:Z2=2:Z4=254:Z5=255:Z6=256:
       Z7=127
CJ 12Ø FA=PEEK(45)+Z6*PEEK(46):BS=PEEK(55)+Z6*PEEK
       (56):H$="Ø123456789ABCDEF"
SB 13Ø R$=CHR$(13):L$="{LEFT}":S$=" ":D$=CHR$(2Ø):
       Z$=CHR$(Ø):T$="{13 RIGHT}"
CQ 14Ø SD=54272:FOR I=SD TO SD+23:POKE I,Ø:NEXT:PO
       KE SD+24,15:POKE 788,52
FC 15Ø PRINT"{CLR}"CHR$(142)CHR$(8):POKE 5328Ø,15:
       POKE 53281,15
EJ 16Ø PRINT T$" {RED}{RVS}{2 SPACES}⁅8 @⁆
       {2 SPACES}"SPC(28)"{2 SPACES}{OFF}{BLU} MLX
        II {RED}{RVS}{2 SPACES}"SPC(28)"
       {12 SPACES}{BLU}"
FR 17Ø PRINT"{3 DOWN}{3 SPACES}COMPUTE!'S MACHINE
       {SPACE}LANGUAGE EDITOR{3 DOWN}"
JB 18Ø PRINT"{BLK}STARTING ADDRESS⁅4⁆";:GOSUB3ØØ:S
       A=AD:GOSUB1Ø4Ø:IF F THEN18Ø
GF 19Ø PRINT"{BLK}{2 SPACES}ENDING ADDRESS⁅4⁆";:GO
       SUB3ØØ:EA=AD:GOSUB1Ø3Ø:IF F THEN19Ø
KR 2ØØ INPUT"{3 DOWN}{BLK}CLEAR WORKSPACE [Y/N]⁅4⁆
       ";A$:IF LEFT$(A$,1)<>"Y"THEN22Ø
```

```
PG 210 PRINT"{2 DOWN}{BLU}WORKING...";:FORI=BS TO
       {SPACE}BS+EA-SA+7:POKE I,Ø:NEXT:PRINT"DONE"
DR 220 PRINTTAB(1Ø)"{2 DOWN}{BLK}{RVS} MLX COMMAND
        MENU {DOWN}⁅4⁆":PRINT T$"{RVS}E{OFF}NTER D
       ATA"
BD 230 PRINT T$"{RVS}D{OFF}ISPLAY DATA":PRINT T$"
       {RVS}L{OFF}OAD FILE"
JS 240 PRINT T$"{RVS}S{OFF}AVE FILE":PRINT T$"
       {RVS}Q{OFF}UIT{2 DOWN}{BLK}"
JH 250 GET A$:IF A$=N$ THEN25Ø
HK 260 A=Ø:FOR I=1 TO 5:IF A$=MID$("EDLSQ",I,1)THE
       N A=I:I=5
FD 270 NEXT:ON A GOTO42Ø,61Ø,69Ø,7ØØ,28Ø:GOSUB1Ø6Ø
       :GOTO25Ø
EJ 280 PRINT"{RVS} QUIT ":INPUT"{DOWN}⁅4⁆ARE YOU S
       URE [Y/N]";A$:IF LEFT$(A$,1)<>"Y"THEN22Ø
EM 290 POKE SD+24,Ø:END
JX 300 IN$=N$:AD=Ø:INPUTIN$:IFLEN(IN$)<>4THENRETUR
       N
KF 310 B$=IN$:GOSUB32Ø:AD=A:B$=MID$(IN$,3):GOSUB32
       Ø:AD=AD*256+A:RETURN
PP 320 A=Ø:FOR J=1 TO 2:A$=MID$(B$,J,1):B=ASC(A$)-
       C4+(A$>"@")*C7:A=A*C6+B
JA 330 IF B<Ø OR B>15 THEN AD=Ø:A=-1:J=2
GX 340 NEXT:RETURN
CH 350 B=INT(A/C6):PRINT MID$(H$,B+1,1);:B=A-B*C6:
       PRINT MID$(H$,B+1,1);:RETURN
RR 360 A=INT(AD/Z6):GOSUB35Ø:A=AD-A*Z6:GOSUB35Ø:PR
       INT":";
BE 370 CK=INT(AD/Z6):CK=AD-Z4*CK+Z5*(CK>Z7):GOTO39
       Ø
PX 380 CK=CK*Z2+Z5*(CK>Z7)+A
JC 390 CK=CK+Z5*(CK>Z5):RETURN
QS 400 PRINT"{DOWN}STARTING AT⁅4⁆";:GOSUB3ØØ:IF IN
       $<>N$ THEN GOSUB1Ø3Ø:IF F THEN4ØØ
EX 410 RETURN
HD 420 PRINT"{RVS} ENTER DATA ":GOSUB4ØØ:IF IN$=N$
        THEN22Ø
JK 430 OPEN3,3:PRINT
SK 440 POKE198,Ø:GOSUB36Ø:IF F THEN PRINT IN$:PRIN
       T"{UP}{5 RIGHT}";
GC 450 FOR I=Ø TO 24 STEP 3:B$=S$:FOR J=1 TO 2:IF
       {SPACE}F THEN B$=MID$(IN$,I+J,1)
HA 460 PRINT"{RVS}"B$L$;:IF I<24THEN PRINT"{OFF}";
HD 470 GET A$:IF A$=N$ THEN47Ø
FK 480 IF(A$>"/"ANDA$<":")OR(A$>"@"ANDA$<"G")THEN5
       4Ø
GS 485 A=-(A$="M")-2*(A$=",")-3*(A$=".")-4*(A$="/"
       )-5*(A$="J")-6*(A$="K")
```

```
FX 486 A=A-7*(A$="L")-8*(A$=":")-9*(A$="U")-1Ø*(A$
       ="I")-11*(A$="O")-12*(A$="P")
CM 487 A=A-13*(A$=S$):IF A THEN A$=MID$("ABCD123E4
       56FØ",A,1):GOTO 54Ø
MP 49Ø IF A$=R$ AND((I=Ø)AND(J=1)OR F)THEN PRINT B
       $;:J=2:NEXT:I=24:GOTO55Ø
KC 5ØØ IF A$="{HOME}" THEN PRINT B$:J=2:NEXT:I=24:
       NEXT:F=Ø:GOTO44Ø
MX 51Ø IF(A$="{RIGHT}")ANDF THENPRINT B$L$;:GOTO54
       Ø
GK 52Ø IF A$<>L$ AND A$<>D$ OR((I=Ø)AND(J=1))THEN
       {SPACE}GOSUB1Ø6Ø:GOTO47Ø
HG 53Ø A$=L$+S$+L$:PRINT B$L$;:J=2-J:IF J THEN PRI
       NT L$;:I=I-3
QS 54Ø PRINT A$;:NEXT J:PRINT S$;
PM 55Ø NEXT I:PRINT:PRINT"{UP}{5 RIGHT}";:INPUT#3,
       IN$:IF IN$=N$ THEN CLOSE3:GOTO22Ø
QC 56Ø FOR I=1 TO 25 STEP3:B$=MID$(IN$,I):GOSUB32Ø
       :IF I<25 THEN GOSUB38Ø:A(I/3)=A
PK 57Ø NEXT:IF A<>CK THEN GOSUB1Ø6Ø:PRINT"{BLK}
       {RVS} ERROR: REENTER LINE ⁅4⁆":F=1:GOTO44Ø
HJ 58Ø GOSUB1Ø8Ø:B=BS+AD-SA:FOR I=Ø TO 7:POKE B+I,
       A(I):NEXT
QQ 59Ø AD=AD+8:IF AD>EA THEN CLOSE3:PRINT"{DOWN}
       {BLU}** END OF ENTRY **{BLK}{2 DOWN}":GOTO7
       ØØ
GQ 6ØØ F=Ø:GOTO44Ø
QA 61Ø PRINT"{CLR}{DOWN}{RVS} DISPLAY DATA ":GOSUB
       4ØØ:IF IN$=N$ THEN22Ø
RJ 62Ø PRINT"{DOWN}{BLU}PRESS: {RVS}SPACE{OFF} TO
       {SPACE}PAUSE, {RVS}RETURN{OFF} TO BREAK⁅4⁆
       {DOWN}"
KS 63Ø GOSUB36Ø:B=BS+AD-SA:FORI=BTO B+7:A=PEEK(I):
       GOSUB35Ø:GOSUB38Ø:PRINT S$;
CC 64Ø NEXT:PRINT"{RVS}";:A=CK:GOSUB35Ø:PRINT
KH 65Ø F=1:AD=AD+8:IF AD>EA THENPRINT"{DOWN}{BLU}*
       * END OF DATA **":GOTO22Ø
KC 66Ø GET A$:IF A$=R$ THEN GOSUB1Ø8Ø:GOTO22Ø
EQ 67Ø IF A$=S$ THEN F=F+1:GOSUB1Ø8Ø
AD 68Ø ONFGOTO63Ø,66Ø,63Ø
CM 69Ø PRINT"{DOWN}{RVS} LOAD DATA ":OP=1:GOTO71Ø
PC 7ØØ PRINT"{DOWN}{RVS} SAVE FILE ":OP=Ø
RX 71Ø IN$=N$:INPUT"{DOWN}FILENAME⁅4⁆";IN$:IF IN$=
       N$ THEN22Ø
PR 72Ø F=Ø:PRINT"{DOWN}{BLK}{RVS}T{OFF}APE OR
       {RVS}D{OFF}ISK: ⁅4⁆";
FP 73Ø GET A$:IF A$="T"THEN PRINT"T{DOWN}":GOTO88Ø
HQ 74Ø IF A$<>"D"THEN73Ø
HH 75Ø PRINT"D{DOWN}":OPEN15,8,15,"IØ:":B=EA-SA:IN
       $="Ø:"+IN$:IF OP THEN81Ø
```

```
SQ 76Ø OPEN 1,8,8,IN$+",P,W":GOSUB86Ø:IF A THEN22Ø
FJ 77Ø AH=INT(SA/256):AL=SA-(AH*256):PRINT#1,CHR$(
       AL);CHR$(AH);
PE 78Ø FOR I=Ø TO B:PRINT#1,CHR$(PEEK(BS+I));:IF S
       T THEN8ØØ
FC 79Ø NEXT:CLOSE1:CLOSE15:GOTO94Ø
GS 8ØØ GOSUB1Ø6Ø:PRINT"{DOWN}{BLK}ERROR DURING SAV
       E:⁅4⁆":GOSUB86Ø:GOTO22Ø
MA 81Ø OPEN 1,8,8,IN$+",P,R":GOSUB86Ø:IF A THEN22Ø
GE 82Ø GET#1,A$,B$:AD=ASC(A$+Z$)+256*ASC(B$+Z$):IF
        AD<>SA THEN F=1:GOTO85Ø
RX 83Ø FOR I=Ø TO B:GET#1,A$:POKE BS+I,ASC(A$+Z$):
       IF(I<>B)AND ST THEN F=2:AD=I:I=B
FA 84Ø NEXT:IF ST<>64 THEN F=3
FQ 85Ø CLOSE1:CLOSE15:ON ABS(F>Ø)+1 GOTO96Ø,97Ø
SA 86Ø INPUT#15,A,A$:IF A THEN CLOSE1:CLOSE15:GOSU
       B1Ø6Ø:PRINT"{RVS}ERROR: "A$
GQ 87Ø RETURN
EJ 88Ø POKE183,PEEK(FA+2):POKE187,PEEK(FA+3):POKE1
       88,PEEK(FA+4):IFOP=ØTHEN92Ø
HJ 89Ø SYS 63466:IF(PEEK(783)AND1)THEN GOSUB1Ø6Ø:P
       RINT"{DOWN}{RVS} FILE NOT FOUND ":GOTO69Ø
CS 9ØØ AD=PEEK(829)+256*PEEK(83Ø):IF AD<>SA THEN F
       =1:GOTO97Ø
SC 91Ø A=PEEK(831)+256*PEEK(832)-1:F=F-2*(A<EA)-3*
       (A>EA):AD=A-AD:GOTO93Ø
KM 92Ø A=SA:B=EA+1:GOSUB1Ø1Ø:POKE78Ø,3:SYS 63338
JF 93Ø A=BS:B=BS+(EA-SA)+1:GOSUB1Ø1Ø:ON OP GOTO95Ø
       :SYS 63591
AE 94Ø GOSUB1Ø8Ø:PRINT"{BLU}** SAVE COMPLETED **":
       GOTO22Ø
XP 95Ø POKE147,Ø:SYS 63562:IF ST>Ø THEN97Ø
FR 96Ø GOSUB1Ø8Ø:PRINT"{BLU}** LOAD COMPLETED **":
       GOTO22Ø
DP 97Ø GOSUB1Ø6Ø:PRINT"{BLK}{RVS}ERROR DURING LOAD
       :{DOWN}⁅4⁆":ON F GOSUB98Ø,99Ø,1ØØØ:GOTO22Ø
PP 98Ø PRINT"INCORRECT STARTING ADDRESS (";:GOSUB3
       6Ø:PRINT")":RETURN
GR 99Ø PRINT"LOAD ENDED AT ";:AD=SA+AD:GOSUB36Ø:PR
       INT D$:RETURN
FD 1ØØØ PRINT"TRUNCATED AT ENDING ADDRESS":RETURN
RX 1Ø1Ø AH=INT(A/256):AL=A-(AH*256):POKE193,AL:POK
        E194,AH
FF 1Ø2Ø AH=INT(B/256):AL=B-(AH*256):POKE174,AL:POK
        E175,AH:RETURN
FX 1Ø3Ø IF AD<SA OR AD>EA THEN1Ø5Ø
HA 1Ø4Ø IF(AD>511 AND AD<4Ø96Ø)OR(AD>49151 AND AD<
        53248)THEN GOSUB1Ø8Ø:F=Ø:RETURN
HC 1Ø5Ø GOSUB1Ø6Ø:PRINT"{RVS} INVALID ADDRESS
        {DOWN}{BLK}":F=1:RETURN
```

```
AR 1060 POKE SD+5,31:POKE SD+6,208:POKE SD,240:POK
        E SD+1,4:POKE SD+4,33
DX 1070 FOR S=1 TO 100:NEXT:GOTO1090
PF 1080 POKE SD+5,8:POKE SD+6,240:POKE SD,0:POKE S
        D+1,90:POKE SD+4,17
AC 1090 FOR S=1 TO 100:NEXT:POKE SD+4,0:POKE SD,0:
        POKE SD+1,0:RETURN
```

Program 1
CUT-OFF!

Assuming that you have a copy of MLX typed in and saved to disk or tape, load and run it. You'll be asked to provide two addresses before you begin entering CUT-OFF!. Respond to the questions with

Starting address: C000
Ending address: C5D7

You're ready to start.

CUT-OFF!

For mistake-proof program entry, use "MLX: Machine Language Entry Program" to type in this program.

```
CØØØ:2Ø C1 C1 4C Ø6 CØ AD 42 CF
CØØ8:Ø3 18 69 Ø1 8D 42 Ø3 AD 77
CØ1Ø:43 Ø3 69 ØØ 8D 43 Ø3 AE 4F
CØ18:3C Ø3 2Ø 8A C3 A9 Ø1 8D 79
CØ2Ø:41 Ø3 2Ø C4 C3 A2 Ø2 A1 A1
CØ28:F7 C9 2Ø FØ 2Ø 2Ø 93 C3 97
CØ3Ø:2Ø 66 C5 2Ø DØ C2 AD 44 47
CØ38:Ø3 C9 ØØ FØ 53 AD 45 Ø3 9B
CØ4Ø:C9 ØØ FØ 4C 2Ø BE CØ 2Ø 27
CØ48:43 C4 4C Ø6 CØ 2Ø 93 C3 F7
CØ5Ø:A9 ØØ 8D 41 Ø3 2Ø C4 C3 52
CØ58:A2 ØØ A1 F7 C9 2Ø FØ 2Ø AF
CØ6Ø:2Ø 93 C3 2Ø 66 C5 2Ø DØ AC
CØ68:C2 AD 44 Ø3 C9 ØØ FØ 2Ø BF
CØ7Ø:AD 45 Ø3 C9 ØØ FØ 19 2Ø 2D
CØ78:BE CØ 2Ø 43 C4 4C Ø6 CØ E5
CØ8Ø:2Ø 93 C3 AD Ø1 DC 2D ØØ 2Ø
CØ88:DC 29 1Ø FØ Ø6 4C Ø6 CØ Ø2
CØ9Ø:4C F4 C4 AD Ø1 DC 2D ØØ BE
CØ98:DC 29 1Ø FØ F6 A2 FA 2Ø 3C
CØAØ:8A C3 AD Ø1 DC 2D ØØ DC 96
CØA8:29 1Ø DØ F6 AD Ø1 DC 2D A4
CØBØ:ØØ DC 29 1Ø FØ F6 A2 FA 33
CØB8:2Ø 8A C3 4C Ø6 CØ A9 93 44
CØCØ:2Ø D2 FF A9 ØF 8D 2Ø DØ 61
CØC8:8D 21 DØ A2 ØØ A9 AØ 9D 23
CØDØ:ØØ Ø4 A9 ØØ 9D ØØ D8 E8 1Ø
CØD8:EØ 28 DØ F1 A2 ØØ A9 AØ 17
CØEØ:9D CØ Ø7 A9 ØØ 9D CØ DB BØ
CØE8:E8 EØ 28 DØ F1 A9 ØØ 85 E4
```

```
CØFØ:FD A9 Ø4 85 FE A9 ØØ 85 D8
CØF8:FB A9 D8 85 FC A2 ØØ A9 72
C1ØØ:AØ AØ ØØ 91 FD A9 ØØ 91 3D
C1Ø8:FB AØ 27 A9 AØ 91 FD A9 22
C11Ø:ØØ 91 FB 18 A5 FD 69 28 19
C118:85 FD A5 FE 69 ØØ 85 FE D7
C12Ø:18 A5 FB 69 28 85 FB A5 24
C128:FC 69 ØØ 85 FC E8 EØ 19 43
C13Ø:DØ CD A9 Ø5 A2 1Ø 9D ØØ A5
C138:D8 E8 EØ 18 DØ F8 A9 83 41
C14Ø:8D 1Ø Ø4 A9 95 8D 11 Ø4 B2
C148:A9 94 8D 12 Ø4 A9 AD 8D 48
C15Ø:13 Ø4 A9 8F 8D 14 Ø4 A9 FA
C158:86 8D 15 Ø4 8D 16 Ø4 A9 DB
C16Ø:A1 8D 17 Ø4 A9 Ø6 8D C7 83
C168:D9 A9 Ø2 8D D1 D9 A9 51 F6
C17Ø:8D C7 Ø5 A9 57 8D D1 Ø5 81
C178:A9 ØØ 8D 42 Ø3 8D 43 Ø3 7E
C18Ø:A9 Ø7 8D 4B Ø3 A9 ØB 8D 63
C188:4A Ø3 A9 D1 8D 46 Ø3 A9 79
C19Ø:Ø5 8D 47 Ø3 A9 C7 8D 48 E2
C198:Ø3 A9 Ø5 8D 49 Ø3 18 A2 AA
C1AØ:ØØ AØ Ø7 2Ø FØ FF AE 3D 51
C1A8:Ø3 AD 3E Ø3 2Ø CD BD 18 DC
C1BØ:A2 ØØ AØ 1D 2Ø FØ FF AE DE
C1B8:3F Ø3 AD 4Ø Ø3 2Ø CD BD 48
C1CØ:6Ø A9 ØØ 8D 3D Ø3 8D 3E Ø7
C1C8:Ø3 8D 3F Ø3 8D 4Ø Ø3 A9 66
C1DØ:51 8D 4D Ø3 A9 57 8D 4C 4C
C1D8:Ø3 2Ø BE CØ A9 Ø5 A2 Ø9 79
C1EØ:9D FØ D8 E8 EØ 1E DØ F8 33
C1E8:A2 Ø9 9D 4Ø D9 E8 EØ 1E ØA
C1FØ:DØ F8 8D A3 D9 A2 ØØ BD 1E
C1F8:A4 C2 FØ Ø6 9D F9 Ø4 E8 C3
C2ØØ:DØ F5 A2 ØØ BD BA C2 FØ ØF
C2Ø8:Ø6 9D 49 Ø5 E8 DØ F5 A9 91
C21Ø:3Ø 8D 3C Ø3 8D A3 Ø5 A2 7Ø
C218:64 2Ø 8A C3 AD Ø1 DC 29 B9
C22Ø:ØF C9 ØE FØ 21 C9 ØD FØ AB
C228:ØA AD Ø1 DC 29 1Ø FØ 3F B6
C23Ø:4C 17 C2 AD 3C Ø3 38 E9 1D
C238:Ø1 C9 2F FØ 1C 8D 3C Ø3 38
C24Ø:8D A3 Ø5 4C 17 C2 AD 3C 36
C248:Ø3 18 69 Ø1 C9 3A FØ 14 BF
C25Ø:8D 3C Ø3 8D A3 Ø5 4C 17 C5
C258:C2 A9 39 8D 3C Ø3 8D A3 56
C26Ø:Ø5 4C 17 C2 A9 3Ø 8D 3C EF
C268:Ø3 8D A3 Ø5 4C 17 C2 AD 89
C27Ø:Ø1 DC 29 1Ø FØ F9 A2 FA 83
C278:2Ø 8A C3 AD 3C Ø3 38 E9 4C
```

```
C28Ø:3Ø AA A9 32 8D 3C Ø3 EØ 65
C288:ØØ FØ ØD AD 3C Ø3 38 E9 ØF
C29Ø:Ø5 8D 3C Ø3 CA 4C 87 C2 ØD
C298:2Ø BE CØ A9 ØA 8D 44 Ø3 A2
C2AØ:8D 45 Ø3 6Ø ØD ØF 16 Ø5 7A
C2A8:2Ø ØA ØF 19 13 14 Ø9 Ø3 32
C2BØ:ØB 2Ø 15 1Ø 2F Ø4 ØF 17 26
C2B8:ØE ØØ 14 ØF 2Ø Ø3 Ø8 ØF E4
C2CØ:ØF 13 Ø5 2Ø 13 ØB Ø9 ØC 18
C2C8:ØC 2Ø ØC Ø5 16 Ø5 ØC ØØ ØB
C2DØ:AD 46 Ø3 CD 48 Ø3 FØ Ø3 2F
C2D8:4C 5C C3 AD 47 Ø3 CD 49 1A
C2EØ:Ø3 DØ 79 AD 4B Ø3 AE 4A 34
C2E8:Ø3 C9 ØE FØ ØC C9 Ø7 FØ B9
C2FØ:ØF C9 ØD FØ 12 C9 ØB FØ DF
C2F8:15 EØ ØD FØ 18 4C 5C C3 6Ø
C3ØØ:EØ ØB FØ 11 4C 5C C3 EØ 26
C3Ø8:ØE FØ ØA 4C 5C C3 EØ Ø7 93
C31Ø:FØ Ø3 4C 5C C3 AD 3D Ø3 72
C318:18 6D 42 Ø3 8D 3D Ø3 AD 94
C32Ø:3E Ø3 69 ØØ 8D 3E Ø3 AD CD
C328:43 Ø3 18 6D 3E Ø3 8D 3E 43
C33Ø:Ø3 AE 44 Ø3 CA 8E 44 Ø3 B9
C338:AD 3F Ø3 18 6D 42 Ø3 8D 5Ø
C34Ø:3F Ø3 AD 4Ø Ø3 69 ØØ 8D 2D
C348:4Ø Ø3 AD 43 Ø3 18 6D 4Ø 2E
C35Ø:Ø3 8D 4Ø Ø3 AE 45 Ø3 CA 5Ø
C358:8E 45 Ø3 6Ø AD 41 Ø3 ØA 61
C36Ø:AA BD 3D Ø3 18 6D 42 Ø3 82
C368:9D 3D Ø3 BD 3E Ø3 69 ØØ 1B
C37Ø:9D 3E Ø3 AD 43 Ø3 18 7D 65
C378:3E Ø3 9D 3E Ø3 AE 41 Ø3 CF
C38Ø:BD 44 Ø3 38 E9 Ø1 9D 44 AE
C388:Ø3 6Ø AØ ØØ C8 DØ FD CA ØE
C39Ø:DØ F8 6Ø AE 41 Ø3 BC 4C 91
C398:Ø3 8A ØA AA B5 F7 9D 46 3F
C3AØ:Ø3 B5 F8 9D 47 Ø3 98 81 Ø9
C3A8:F7 B5 F8 18 69 D4 95 F8 FC
C3BØ:EØ Ø2 DØ Ø8 A9 Ø6 81 F7 24
C3B8:2Ø 99 C5 6Ø A9 Ø2 81 F7 C5
C3CØ:2Ø 99 C5 6Ø AE 41 Ø3 BD BB
C3C8:ØØ DC 29 ØF C9 ØE FØ 12 18
C3DØ:C9 Ø7 FØ 26 C9 ØD FØ 3A 1E
C3D8:C9 ØB FØ 4E BD 4A Ø3 4C 74
C3EØ:CC C3 9D 4A Ø3 8A ØA AA 19
C3E8:38 BD 46 Ø3 E9 28 95 F7 Ø8
C3FØ:BD 47 Ø3 E9 ØØ 95 F8 4C BC
C3F8:42 C4 9D 4A Ø3 8A ØA AA 2C
C4ØØ:18 BD 46 Ø3 69 Ø1 95 F7 7Ø
C4Ø8:BD 47 Ø3 69 ØØ 95 F8 4C CD
```

```
C41Ø:42 C4 9D 4A Ø3 8A ØA AA 45
C418:18 BD 46 Ø3 69 28 95 F7 25
C42Ø:BD 47 Ø3 69 ØØ 95 F8 4C E5
C428:42 C4 9D 4A Ø3 8A ØA AA 5D
C43Ø:38 BD 46 Ø3 E9 Ø1 95 F7 B4
C438:BD 47 Ø3 E9 ØØ 95 F8 4C Ø6
C44Ø:42 C4 6Ø A2 ØØ BD D4 C4 B7
C448:FØ Ø6 9D FF Ø4 E8 DØ F5 DA
C45Ø:A2 ØØ BD E1 C4 FØ Ø6 9D 94
C458:4F Ø5 E8 DØ F5 A2 ØØ BD EC
C46Ø:ED C4 FØ Ø6 9D 11 Ø6 E8 B6
C468:DØ F5 A2 ØØ BD ED C4 FØ 4C
C47Ø:Ø6 9D 21 Ø6 E8 DØ F5 A9 Ø9
C478:Ø5 A2 Ø9 9D FØ D8 E8 EØ C5
C48Ø:1E DØ F8 A2 Ø9 9D 4Ø D9 AF
C488:E8 EØ 1E DØ F8 A2 Ø1 9D 81
C49Ø:Ø8 DA E8 EØ 25 DØ F8 18 76
C498:A2 ØD AØ ØF 2Ø FØ FF AE 2F
C4AØ:45 Ø3 A9 ØØ 2Ø CD BD 18 8E
C4A8:A2 ØD AØ 1F 2Ø FØ FF AE 4Ø
C4BØ:44 Ø3 A9 ØØ 2Ø CD BD AD B3
C4B8:Ø1 DC 2D ØØ DC 29 1Ø DØ 1C
C4CØ:F6 AD Ø1 DC 2D ØØ DC 29 6B
C4C8:1Ø FØ F6 A2 ØØ 2Ø 8A C3 F8
C4DØ:2Ø BE CØ 6Ø 1Ø 12 Ø5 13 1E
C4D8:13 2Ø Ø2 15 14 14 ØF ØE A2
C4EØ:ØØ 14 ØF 2Ø Ø3 ØF ØE 14 D7
C4E8:Ø9 ØE 15 Ø5 ØØ ØC Ø9 16 C5
C4FØ:Ø5 13 3D ØØ 2Ø BE CØ A2 89
C4F8:ØØ BD 41 C5 FØ Ø6 9D ØØ 51
C5ØØ:Ø5 E8 DØ F5 A2 ØØ BD 4B 9D
C5Ø8:C5 FØ Ø6 9D 47 Ø5 E8 DØ 3E
C51Ø:F5 A9 Ø5 A2 Ø1 9D FØ D8 Ø5
C518:E8 EØ 1E DØ F8 A2 Ø1 9D 13
C52Ø:4Ø D9 E8 EØ 25 DØ F8 AD 79
C528:Ø1 DC 2D ØØ DC 29 1Ø DØ 8D
C53Ø:F6 AD Ø1 DC 2D ØØ DC 29 DC
C538:1Ø FØ F6 2Ø C1 C1 4C Ø6 9C
C54Ø:CØ Ø7 Ø1 ØD Ø5 2Ø ØF 16 BB
C548:Ø5 12 ØØ 1Ø 12 Ø5 13 13 B9
C55Ø:2Ø Ø2 15 14 14 ØF ØE 2Ø 69
C558:14 ØF 2Ø 1Ø ØC Ø1 19 2Ø 6D
C56Ø:Ø1 Ø7 Ø1 Ø9 ØE ØØ A9 ØF B1
C568:8D 18 D4 A9 81 8D Ø4 D4 15
C57Ø:A9 Ø9 8D Ø5 D4 A9 64 8D B8
C578:ØØ D4 A9 ØC 8D Ø1 D4 A9 F2
C58Ø:ØF 8D 2Ø DØ A2 28 2Ø 8A 88
C588:C3 38 E9 Ø1 C9 ØØ DØ F1 33
C59Ø:A9 ØØ 8D Ø4 D4 8D Ø5 D4 9E
```

```
C598:60 A9 08 8D 18 D4 A9 10 10
C5A0:8D 05 D4 A9 80 8D 06 D4 84
C5A8:A9 0A A2 00 18 6D 41 03 DB
C5B0:E8 E0 0A D0 F7 8D 01 D4 04
C5B8:A9 25 8D 00 D4 A9 21 8D 31
C5C0:04 D4 AE 3C 03 20 8A C3 8E
C5C8:A9 00 8D 04 D4 8D 05 D4 D6
C5D0:8D 06 D4 60 00 00 00 00 45
```

Program 2
Nessie

Once you've typed in and saved a copy of MLX to disk or tape, load and run it. As always, you have to provide the starting and ending addresses for the MLX listing. In Nessie's case, these are

Starting address: C000
Ending address: CBC7

You can now begin to type in the game. Refer to the first article in this chapter, "MLX: Machine Language Entry Program," for instructions on entering Nessie in more than one session and for saving the final program to disk.

Nessie

For mistake-proof program entry, use "MLX: Machine Language Entry Program" to type in this program.

```
CØØØ:A9 47 85 FB A9 C7 85 FC ØD
CØØ8:A9 ØØ 85 FD A9 38 85 FE 27
CØ1Ø:A2 Ø4 AØ FF B1 FB 91 FD 96
CØ18:88 CØ FF DØ F7 E6 FC E6 57
CØ2Ø:FE CA 3Ø Ø7 DØ EE AØ 7F 4D
CØ28:4C 14 CØ C6 FE AØ 8Ø A9 7E
CØ3Ø:ØØ 91 FD C8 DØ FB A9 FF 2C
CØ38:8D ØE D4 8D ØF D4 A9 8Ø 17
CØ4Ø:8D 12 D4 A9 ØØ 8D 18 D4 7D
CØ48:A9 FC 8D 1B DØ A2 Ø5 2Ø 7C
CØ5Ø:BD C1 A9 8Ø 9D CD CB 9D B7
CØ58:D3 CB A9 ØØ 9D C7 CB BD 4D
CØ6Ø:18 C7 9D 29 DØ BD 2Ø C7 AB
CØ68:9D FA Ø7 8A ØA A8 A9 ØØ 47
CØ7Ø:99 Ø5 DØ CA 1Ø D9 AD 16 2Ø
CØ78:C7 8D 27 DØ AD 17 C7 8D 1A
CØ8Ø:28 DØ AD 1E C7 8D F8 Ø7 4F
CØ88:AD 1F C7 8D F9 Ø7 A9 ØØ B9
CØ9Ø:8D EC CB A9 AØ 8D EB CB Ø7
CØ98:8D ØØ DØ 8D Ø2 DØ A9 8Ø FA
CØAØ:8D ED CB 8D Ø1 DØ 8D Ø3 2Ø
CØA8:DØ A9 FF 8D 15 DØ 8D 1C F8
CØBØ:DØ A9 Ø3 8D 1D DØ 8D 17 9C
CØB8:DØ A9 Ø6 8D 2Ø DØ 8D 21 27
CØCØ:DØ A9 Ø5 8D 25 DØ A9 Ø4 52
CØC8:8D 26 DØ A9 Ø1 8D F6 CB 47
CØDØ:2Ø E7 C1 78 A9 76 8D 14 72
```

```
CØD8:Ø3 A9 C4 8D 15 Ø3 58 AØ BD
CØEØ:ØØ 84 Ø7 98 ØA AA B9 C7 24
CØE8:CB 4A 66 Ø7 B9 CD CB 9D 5A
CØFØ:Ø4 DØ B9 D3 CB 9D Ø5 DØ CC
CØF8:C9 32 9Ø Ø7 C9 E5 BØ Ø3 B8
C1ØØ:4C ØE C1 B9 E5 CB 49 FF F1
C1Ø8:18 69 Ø1 99 E5 CB B9 C7 45
C11Ø:CB FØ ØA B9 CD CB C9 4Ø Ø4
C118:BØ ØD 4C 3E C1 B9 CD CB Ø1
C12Ø:C9 18 9Ø Ø3 4C 3E C1 A9 59
C128:ØØ 99 DF CB B9 D9 CB 49 EØ
C13Ø:FF 18 69 Ø1 99 D9 CB C9 8C
C138:FF DØ Ø3 99 DF CB B9 FA 86
C14Ø:Ø7 49 Ø1 99 FA Ø7 B9 D9 94
C148:CB AA B9 FA Ø7 Ø9 Ø2 99 3D
C15Ø:FA Ø7 EØ Ø1 FØ Ø5 49 Ø2 6F
C158:99 FA Ø7 18 B9 CD CB 79 DF
C16Ø:D9 CB 99 CD CB B9 C7 CB 74
C168:79 DF CB 99 C7 CB 18 B9 ØB
C17Ø:D3 CB 79 E5 CB 99 D3 CB 96
C178:C8 CØ Ø6 FØ Ø3 4C E3 CØ 32
C18Ø:AD 1Ø DØ 29 Ø3 Ø5 Ø7 8D 53
C188:1Ø DØ AE F5 CB FØ Ø8 AØ 5Ø
C19Ø:ØØ 88 DØ FD CA DØ FA AD 6D
C198:1B D4 C9 8Ø 9Ø ØB AD 1B 47
C1AØ:D4 C9 Ø6 BØ F9 AA 2Ø BD 45
C1A8:C1 2Ø E4 FF FØ ØC C9 88 85
C1BØ:DØ Ø3 4C DØ C5 2Ø E4 FF 6C
C1B8:FØ FB 4C DF CØ A9 ØØ 9D 85
C1CØ:DF CB AD 1B D4 29 Ø3 A8 88
C1C8:B9 3C C7 9D D9 CB C9 FF 9C
C1DØ:DØ Ø3 9D DF CB AD 1B D4 4F
C1D8:29 Ø3 A8 B9 3C C7 9D E5 84
C1EØ:CB 1D D9 CB FØ D7 6Ø A9 DA
C1E8:ØØ 8D EF CB 8D FØ CB 8D DF
C1FØ:F1 CB 8D F2 CB 8D F3 CB 89
C1F8:8D 4C CC A9 14 8D F4 CB 17
C2ØØ:A9 Ø5 8D F5 CB A9 BF AØ D1
C2Ø8:C2 2Ø 1E AB A9 FØ AØ C2 8A
C21Ø:2Ø 1E AB 2Ø E4 FF FØ FB A9
C218:C9 85 DØ 12 A9 EØ 8D F8 Ø4
C22Ø:Ø7 A9 E1 8D F9 Ø7 A9 ØØ E7
C228:8D F3 CB 4C 13 C2 C9 86 6D
C23Ø:DØ 12 A9 E2 8D F8 Ø7 A9 ØE
C238:E3 8D F9 Ø7 A9 Ø1 8D F3 23
C24Ø:CB 4C 13 C2 C9 87 DØ 1E 79
C248:EE F5 CB AD F5 CB 1Ø Ø5 1B
C25Ø:A9 Ø9 8D F5 CB C9 ØA 9Ø 28
C258:Ø5 A9 ØØ 8D F5 CB Ø9 3Ø C4
C26Ø:8D Ø6 Ø4 4C 13 C2 C9 88 33
```

```
C268:FØ 33 C9 8C DØ A5 EE 4C 7C
C27Ø:CC 2Ø 9D C2 A2 32 AØ 28 8B
C278:AD 1B D4 C9 A7 BØ F9 85 4C
C28Ø:FD AD 1B D4 29 Ø3 18 69 1Ø
C288:Ø4 85 FE A9 AØ 91 FD A5 D8
C29Ø:FE 69 D4 85 FE A9 ØØ 91 13
C298:FD CA DØ DC 6Ø A9 BF AØ 81
C2AØ:C2 2Ø 1E AB AD F5 CB Ø9 F3
C2A8:3Ø 8D Ø6 Ø4 38 A9 Ø9 ED 13
C2BØ:F5 CB 8D F5 CB ØE F5 CB 83
C2B8:ØE F5 CB ØE F5 CB 6Ø 93 5Ø
C2CØ:92 ØE 9F D3 5Ø 45 45 44 AA
C2C8:3A 9E 35 2Ø 2Ø 2Ø 2Ø 2Ø 9D
C2DØ:2Ø 2Ø 2Ø 9A CE 45 53 53 A1
C2D8:49 45 2Ø 2Ø 2Ø 2Ø 2Ø 2Ø 3C
C2EØ:9B C6 49 4C 4D 2Ø CC 45 9D
C2E8:46 54 3A 9E 32 3Ø ØD ØØ 44
C2FØ:99 12 46 31 92 3A D4 45 1Ø
C2F8:4C 45 5Ø 48 4F 54 4F 2Ø ØF
C3ØØ:12 46 33 92 3A D7 49 44 B9
C3Ø8:45 2Ø C1 4E 47 4C 45 2Ø 6D
C31Ø:12 46 35 92 3A D3 5Ø 45 Ø9
C318:45 44 ØD 12 46 37 92 3A 84
C32Ø:C5 41 53 59 2Ø C7 41 4D CA
C328:45 2Ø 12 46 38 92 3A C8 4A
C33Ø:41 52 44 2Ø C7 41 4D 45 9A
C338:ØØ AD ØØ DC 48 29 1Ø 8D 8D
C34Ø:EE CB 68 29 ØF 49 ØF AA 38
C348:18 AD EB CB 7D 31 C7 8D 4F
C35Ø:EB CB 8D ØØ DØ 8D Ø2 DØ Ø4
C358:AD EC CB 7D 26 C7 8D EC 9B
C36Ø:CB 4A Ø8 AD 1Ø DØ 4A 4A DE
C368:28 Ø8 2A 28 2A 8D 1Ø DØ 46
C37Ø:18 AD ED CB 7D 3C C7 8D E3
C378:ED CB 8D Ø1 DØ 8D Ø3 DØ 3F
C38Ø:AD EE CB FØ ØD A9 ØØ 8D BF
C388:F7 CB A9 Ø1 8D F6 CB 4C 7Ø
C39Ø:AØ C3 AD F7 CB FØ 17 AD 8C
C398:F6 CB FØ Ø4 CE F6 CB 6Ø 37
C3AØ:A9 Ø6 8D 2Ø DØ 8D 21 DØ Ø2
C3A8:A9 8Ø 8D Ø4 D4 6Ø CE 26 Ø3
C3BØ:Ø4 AD 26 Ø4 C9 3Ø BØ 14 2F
C3B8:A9 39 8D 26 Ø4 CE 25 Ø4 21
C3CØ:AD 25 Ø4 C9 31 BØ Ø5 A9 85
C3C8:2Ø 8D 25 Ø4 A9 1F 8D Ø5 92
C3DØ:D4 8D 18 D4 A9 Ø3 8D Ø6 FØ
C3D8:D4 A9 FF 8D ØØ D4 8D Ø1 7D
C3EØ:D4 8D Ø4 D4 A9 Ø4 8D 4A C6
C3E8:CC A2 Ø6 AC F4 CB 88 A9 9C
C3FØ:ØØ 99 2Ø CC AD 4A CC Ø9 E8
```

```
C3F8:Ø2 8D 15 DØ AD 1E DØ 2Ø 3C
C4ØØ:7F C4 AD 1E DØ 29 FD 19 52
C4Ø8:2Ø CC 99 2Ø CC ØE 4A CC ØA
C41Ø:CA DØ E1 A9 Ø1 8D 2Ø DØ 59
C418:8D 21 DØ 8D F7 CB A9 Ø4 E9
C42Ø:8D 4A CC A2 Ø6 A9 ØØ 99 37
C428:34 CC AD 4A CC Ø9 Ø1 8D 73
C43Ø:15 DØ AD 1E DØ 2Ø 7F C4 DA
C438:AD 1E DØ 29 FE 19 34 CC 5E
C44Ø:99 34 CC ØE 4A CC CA DØ ØA
C448:E1 AD 1F DØ 2Ø 7F C4 AD 55
C45Ø:1F DØ 99 F8 CB AD ØØ DC 52
C458:29 ØF 49 ØF 99 ØC CC A9 94
C46Ø:FF 8D 15 DØ CE F4 CB AD 8C
C468:F4 CB FØ 26 C9 Ø5 DØ Ø5 E8
C47Ø:A9 Ø7 8D 28 DØ 6Ø 2Ø 89 96
C478:C4 2Ø 39 C3 4C 31 EA 2Ø EC
C48Ø:82 C4 AD 12 DØ C9 FF DØ D1
C488:F9 EE F1 CB DØ Ø3 EE F2 29
C49Ø:CB 6Ø A9 93 2Ø D2 FF A9 7C
C498:ØØ 8D 2Ø DØ 8D 21 DØ 78 A1
C4AØ:A9 ØF 8D 14 Ø3 A9 C6 8D 8F
C4A8:15 Ø3 A9 Ø1 8D 1A DØ A9 E2
C4BØ:1B 8D 11 DØ A9 32 8D 12 9D
C4B8:DØ A9 7F 8D ØD DC A9 FF ØD
C4CØ:8D 1B DØ A9 ØØ 8D 4B CC 26
C4C8:A9 1F 8D 15 DØ A9 FF 8D AC
C4DØ:1D DØ 8D 17 DØ A9 1Ø 8D 1B
C4D8:1Ø DØ A2 ØØ 8A ØA A8 ØA CA
C4EØ:ØA ØA ØA ØA 18 69 1D 99 ØE
C4E8:ØØ DØ A9 Ø9 9D 27 DØ E8 8Ø
C4FØ:EØ Ø5 DØ E8 58 A2 13 AØ E8
C4F8:ØØ BD F8 CB 29 Ø3 1D ØC 69
C5ØØ:CC FØ ØA A9 F1 99 65 C6 91
C5Ø8:A9 ØØ 4C 56 C5 BD 34 CC B1
C51Ø:29 4Ø DØ 2C BD 34 CC DØ 46
C518:ØA A9 F2 99 65 C6 A9 ØØ A4
C52Ø:4C 56 C5 8E 48 CC BD 34 2E
C528:CC A2 ØØ 4A BØ Ø4 E8 4C 1B
C53Ø:2B C5 BD 1E C7 99 65 C6 92
C538:AE 48 CC A9 Ø5 4C 56 C5 2D
C54Ø:BD 2Ø CC FØ ØA A9 FØ 99 CD
C548:65 C6 A9 ØA 4C 56 C5 A9 FE
C55Ø:EC 99 65 C6 A9 C8 18 6D DF
C558:EF CB 8D EF CB AD FØ CB 42
C56Ø:69 ØØ 8D FØ CB CA C8 CØ 3D
C568:14 DØ 8E 2Ø 79 C6 AD EF 38
C57Ø:CB ØD FØ CB FØ 1D 38 AD 1A
C578:EF CB ED F2 CB 8D EF CB 1C
C58Ø:AD FØ CB E9 ØØ 8D FØ CB 1B
```

```
C588:AD 4C CC FØ Ø6 ØE EF CB BA
C59Ø:2E FØ CB A2 18 AØ ØØ 18 6E
C598:2Ø FØ FF A9 EC AØ C5 2Ø AØ
C5AØ:1E AB AE EF CB AD FØ CB BD
C5A8:2Ø CD BD A9 3Ø 2Ø D2 FF B1
C5BØ:AD F3 CB DØ Ø5 A9 3Ø 2Ø E5
C5B8:D2 FF A2 78 2Ø 7F C4 CA DC
C5CØ:DØ FA AD ØØ DC 29 1Ø FØ C5
C5C8:F9 AD ØØ DC 29 1Ø DØ F9 AF
C5DØ:78 A9 ØØ 8D 1A DØ A9 FF 43
C5D8:8D ØD DC A9 31 8D 14 Ø3 8F
C5EØ:A9 EA 8D 15 Ø3 58 A2 FF BD
C5E8:9A 4C ØØ CØ 9F DØ 4C 41 FA
C5FØ:59 2Ø 41 47 41 49 4E 3A D3
C5F8:D3 4E 41 5Ø 2Ø 53 48 55 63
C6ØØ:54 54 45 52 2Ø 9A D3 43 FØ
C6Ø8:4F 52 45 3A 9E 12 ØØ AD Ø9
C61Ø:12 DØ 18 69 Ø2 8D Ø1 DØ 8D
C618:8D Ø3 DØ 8D Ø5 DØ 8D Ø7 AD
C62Ø:DØ 8D Ø9 DØ 8D ØB DØ AE 9Ø
C628:4B CC BD 61 C6 A8 A2 ØØ 7A
C63Ø:B9 65 C6 9D F8 Ø7 C8 E8 Ø5
C638:EØ Ø5 DØ F4 EE 4B CC AD CC
C64Ø:4B CC C9 Ø4 DØ Ø5 A9 ØØ ØE
C648:8D 4B CC AA BD 5D C6 8D 32
C65Ø:12 DØ A9 Ø1 8D 19 DØ 68 3B
C658:A8 68 AA 68 4Ø 31 61 92 4C
C66Ø:C2 ØØ Ø5 ØA ØF EØ E1 E2 33
C668:E3 E4 E4 E3 E1 E2 E4 E1 42
C67Ø:E3 E2 E4 EØ E1 E3 E1 EØ 96
C678:E5 A2 ØØ A9 Ø7 9D ØØ D8 C3
C68Ø:9D ØØ D9 9D ØØ DA 9D ØØ 98
C688:DB E8 DØ F1 A9 ØØ 85 FD CD
C69Ø:A9 Ø4 8D 49 CC 85 FE A9 5E
C698:Ø5 8D 48 CC AØ ØØ A9 7Ø AA
C6AØ:91 FD C8 A2 Ø6 A9 4Ø 91 A2
C6A8:FD C8 CA DØ FA A9 6E 91 BA
C6BØ:FD C8 CE 48 CC AD 48 CC 48
C6B8:DØ E4 A2 Ø4 2Ø Ø8 C7 AØ CD
C6CØ:ØØ 18 A9 5D 91 FD 98 69 7E
C6C8:Ø7 A8 A9 5D 91 FD C8 CØ E5
C6DØ:28 9Ø EF CA DØ E6 2Ø Ø8 AB
C6D8:C7 A9 Ø5 8D 48 CC AØ ØØ E4
C6EØ:A9 6D 91 FD C8 A2 Ø6 A9 37
C6E8:4Ø 91 FD C8 CA DØ FA A9 8Ø
C6FØ:7D 91 FD C8 CE 48 CC AD CC
C6F8:48 CC DØ E4 2Ø Ø8 C7 CE C5
C7ØØ:49 CC AD 49 CC DØ 9Ø 6Ø DC
C7Ø8:A5 FD 18 69 28 85 FD A5 7C
C71Ø:FE 69 ØØ 85 FE 6Ø ØA ØE 6D
```

```
C718:Ø9 Ø9 Ø1 Ø1 Ø7 Ø9 EØ E1 9E
C72Ø:E4 E4 E8 E8 EC E4 ØØ ØØ Ø2
C728:ØØ ØØ FF FF FF ØØ ØØ ØØ B7
C73Ø:ØØ ØØ ØØ ØØ ØØ FF FF FF BF
C738:ØØ Ø1 Ø1 Ø1 ØØ FF Ø1 ØØ 3A
C74Ø:ØØ FF Ø1 ØØ ØØ FF Ø1 ØØ F1
C748:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ D7
C75Ø:ØØ 8Ø ØØ ØØ 8Ø ØØ Ø2 AØ A8
C758:ØØ ØØ 8Ø ØØ ØØ 8Ø ØØ ØØ F9
C76Ø:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ EF
C768:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ F7
C77Ø:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ FF
C778:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ Ø8
C78Ø:ØØ ØØ ØØ ØØ ØØ ØØ 93 AA E1
C788:AA 8Ø 8Ø ØØ 8Ø 8Ø ØØ 8Ø 24
C79Ø:8Ø ØØ 8Ø 8Ø ØØ 8Ø 8Ø ØØ 7B
C798:8Ø 8Ø ØØ 8Ø 8Ø ØØ 8Ø 8Ø 16
C7AØ:ØØ 8Ø 8Ø ØØ 8Ø 8Ø ØØ 8Ø E6
C7A8:AA AA 8Ø ØØ ØØ ØØ ØØ ØØ 48
C7BØ:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 4Ø
C7B8:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 48
C7CØ:ØØ ØØ ØØ ØØ ØØ ØØ 93 ØØ 77
C7C8:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 58
C7DØ:ØØ ØØ ØØ ØØ ØØ ØØ ØØ Ø8 68
C7D8:ØØ ØØ Ø8 ØØ ØØ Ø8 ØØ ØØ 89
C7EØ:AA 8Ø ØØ Ø8 ØØ ØØ Ø8 ØØ 76
C7E8:ØØ Ø8 ØØ ØØ ØØ ØØ ØØ ØØ 7A
C7FØ:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 8Ø
C7F8:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 88
C8ØØ:ØØ ØØ ØØ ØØ ØØ ØØ 3Ø 2A 1C
C8Ø8:AA AA 2Ø ØØ Ø2 2Ø ØØ Ø2 3Ø
C81Ø:2Ø ØØ Ø2 2Ø ØØ Ø2 2Ø ØØ 3C
C818:Ø2 2Ø ØØ Ø2 2Ø ØØ Ø2 2Ø F7
C82Ø:ØØ Ø2 2Ø ØØ Ø2 2Ø ØØ Ø2 C8
C828:2Ø ØØ Ø2 2Ø ØØ Ø2 2Ø ØØ 54
C83Ø:Ø2 2Ø ØØ Ø2 2Ø ØØ Ø2 2A 1A
C838:AA AA ØØ ØØ ØØ ØØ ØØ ØØ C9
C84Ø:ØØ ØØ ØØ ØØ ØØ ØØ 3Ø ØØ 32
C848:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ D9
C85Ø:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ E1
C858:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ E9
C86Ø:ØØ ØØ 5Ø AA ØØ 76 AA AØ 76
C868:56 AE A8 16 EA BA Ø6 AE 44
C87Ø:AA ØB AA E9 ØA AA 95 Ø5 39
C878:4Ø 14 Ø1 ØØ 5Ø ØØ 4Ø 4Ø 92
C88Ø:ØØ ØØ ØØ ØØ ØØ ØØ E9 ØØ E5
C888:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 1A
C89Ø:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 22
C898:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 2A
C8AØ:ØØ ØØ ØØ AA ØØ 52 AA AØ 1C
```

```
C8A8:76 AE A8 56 EA BA 16 AE B8
C8BØ:AA ØB AA E9 ØA AA 94 Ø1 73
C8B8:4Ø 14 ØØ 5Ø Ø5 ØØ 1Ø ØØ BC
C8CØ:ØØ ØØ ØØ ØØ ØØ ØØ E9 ØØ 26
C8C8:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 5A
C8DØ:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 62
C8D8:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 6A
C8EØ:ØØ ØØ ØØ AA Ø5 ØA AA 9D 6Ø
C8E8:2A BA 95 AE AB 94 AA BA 9B
C8FØ:9Ø 6B AA EØ 56 AA AØ 14 BB
C8F8:Ø1 5Ø Ø5 ØØ 4Ø Ø1 Ø1 ØØ C7
C9ØØ:ØØ ØØ ØØ ØØ ØØ ØØ E9 ØØ 67
C9Ø8:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 9B
C91Ø:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ A3
C918:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ AB
C92Ø:ØØ ØØ ØØ AA ØØ ØA AA 85 61
C928:2A BA 9D AE AB 95 AA BA E1
C93Ø:94 6B AA EØ 16 AA AØ 14 FC
C938:Ø1 4Ø 5Ø Ø5 ØØ ØØ Ø4 ØØ BE
C94Ø:ØØ ØØ ØØ ØØ ØØ ØØ 16 ØØ FF
C948:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ DB
C95Ø:ØØ ØØ ØØ ØØ ØØ ØØ 1Ø ØØ Ø4
C958:ØØ 74 ØØ ØØ 14 ØØ ØØ Ø5 AE
C96Ø:ØØ ØØ Ø1 ØØ ØØ Ø1 44 4Ø EØ
C968:Ø1 95 9Ø Ø1 59 5Ø ØØ 44 54
C97Ø:64 ØØ ØØ 14 ØØ ØØ Ø5 ØØ 81
C978:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØC
C98Ø:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 14
C988:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 1C
C99Ø:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 24
C998:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 2C
C9AØ:ØØ ØØ 1Ø ØØ ØØ 74 91 Ø1 2C
C9A8:15 56 55 Ø9 65 94 ØØ 11 26
C9BØ:4Ø ØØ ØØ ØØ ØØ ØØ ØØ ØØ 64
C9B8:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 4C
C9CØ:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 54
C9C8:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 5C
C9DØ:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 64
C9D8:Ø4 ØØ ØØ 1D ØØ ØØ 14 ØØ 68
C9EØ:ØØ 5Ø ØØ ØØ 4Ø Ø1 11 8Ø 31
C9E8:Ø5 95 4Ø Ø9 59 4Ø 15 11 Ø4
C9FØ:ØØ 24 ØØ ØØ 5Ø ØØ ØØ ØØ 1Ø
C9F8:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 8C
CAØØ:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 95
CAØ8:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 9D
CA1Ø:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ A5
CA18:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ AD
CA2Ø:ØØ ØØ ØØ ØØ Ø4 4Ø 45 1D 7E
CA28:65 65 54 16 56 4Ø Ø1 44 AF
CA3Ø:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ C5
```

```
CA38:00 00 00 00 00 00 00 00 CD
CA40:00 00 00 00 00 00 00 00 D5
CA48:00 00 00 00 00 00 00 00 DD
CA50:00 00 00 00 00 00 14 00 0E
CA58:00 5D 00 00 25 00 00 05 73
CA60:00 00 05 40 00 05 44 40 77
CA68:01 55 50 02 56 50 05 65 61
CA70:54 05 55 95 01 55 50 01 74
CA78:40 50 00 50 14 00 00 00 E7
CA80:00 00 00 00 00 00 FF 00 16
CA88:00 00 00 00 00 00 00 00 1E
CA90:00 00 00 14 00 00 5D 00 22
CA98:00 25 00 00 05 00 00 05 A4
CAA0:00 00 05 40 00 01 44 40 A7
CAA8:01 55 50 02 56 54 05 65 B1
CAB0:55 05 55 90 01 55 50 01 E4
CAB8:01 40 01 01 40 00 40 50 E1
CAC0:00 00 00 00 00 00 FF 00 56
CAC8:00 00 00 00 00 00 00 00 5E
CAD0:00 00 00 00 00 00 00 00 66
CAD8:14 00 00 75 00 00 58 00 80
CAE0:00 50 00 01 50 01 11 50 93
CAE8:05 55 40 05 95 80 15 59 E0
CAF0:50 56 55 50 05 55 40 05 F6
CAF8:01 40 14 05 00 00 00 00 F1
CB00:00 00 00 00 00 00 00 00 97
CB08:00 00 00 00 00 00 00 00 9F
CB10:00 00 00 00 00 14 00 00 F7
CB18:75 00 00 58 00 00 50 00 90
CB20:00 50 00 01 50 01 11 40 C4
CB28:05 55 40 15 95 80 55 59 A3
CB30:50 06 55 50 05 55 40 01 20
CB38:40 40 01 40 40 05 01 00 3C
CB40:00 00 00 00 00 00 00 00 D7
CB48:00 00 00 00 00 00 00 00 DF
CB50:00 00 00 00 00 00 00 00 E7
CB58:00 00 00 00 00 00 00 00 EF
CB60:00 00 00 00 00 00 00 00 F7
CB68:00 00 00 00 00 00 40 00 80
CB70:00 50 00 00 51 10 00 55 3C
CB78:54 00 59 64 00 55 55 00 AB
CB80:56 59 00 55 54 00 7D 00 8C
CB88:00 00 00 00 00 00 2A A0 15
CB90:00 00 00 00 00 00 FF FF 28
CB98:FF 00 AA A5 0F FF FF 00 58
CBA0:00 50 2A 81 40 01 11 50 22
CBA8:05 5A A0 F0 3F FF 15 59 F9
CBB0:50 56 AA AA 05 55 40 FF 04
CBB8:FF 40 14 05 00 AA 28 00 2E
CBC0:00 00 00 00 00 00 00 00 58
```

Program 3
Campaign Manager

Before beginning to enter Campaign Manager, you *must* have a copy of MLX on tape or disk.

Load MLX and answer the starting and ending address prompts with

Starting address: 0801
Ending address: 2530

and begin typing. The program uses about 10K, which was crunched down to about 7K to make typing it in a little easier. Since it's such a long program, you may want to enter it in parts.

When you've finished typing Campaign Manager, make sure to save it to tape or disk (maybe with a couple of backup copies as well).

Campaign Manager

For mistake-proof program entry, use "MLX: Machine Language Entry Program" to type in this program.

```
Ø8Ø1:ØB Ø8 ØA ØØ 9E 32 3Ø 36 2E
Ø8Ø9:31 ØØ ØØ ØØ 2Ø 6E ØC 2Ø A4
Ø811:F1 ØC 2Ø 7A 11 2Ø 6C 1F C9
Ø819:45 FA CC CC CC CC DC CØ Ø5
Ø821:ØØ ØØ ØØ Ø5 E5 FF A7 FF FF
Ø829:FF FF FF B2 1E 8Ø ØØ ØØ 57
Ø831:DB 5F FA 1F FF FF FF FF 58
Ø839:8F 2D ØØ Ø4 F5 25 FF FF EØ
Ø841:1F FF FF FF F1 C5 FA 4C C9
Ø849:FF F8 5F 5F FF FF FF FF 79
Ø851:8F BF AF F5 FF 73 25 F5 7C
Ø859:FF FF FF FC CC FE FA F7 89
Ø861:23 4C 2Ø ØF F7 FF FF FF CA
Ø869:FF FA FE A2 FA 2F 12 ØØ FA
Ø871:5F 15 FF FF E3 FF 5C FC 4B
Ø879:CC 3C CC ØØ ØØ 7F AF FF F5
Ø881:FF FF FA FF CC 3C CF BØ 98
Ø889:ØØ Ø1 F2 FF FF BF FF EF 27
Ø891:AF FA F7 EØ ØØ ØØ ØØ Ø1 46
Ø899:33 7F FF FF F2 FF FF FØ AB
Ø8A1:ØØ ØØ 4F AØ 8Ø ØØ 77 FF 98
Ø8A9:2F E1 23 7F ØØ ØØ ØD FF 4Ø
Ø8B1:ØØ 6Ø ØØ Ø7 BØ ØØ ØØ ØØ CF
```

```
Ø8B9:7E ØØ ØØ D3 3A ØØ 7Ø ØØ F8
Ø8C1:Ø2 ØØ ØØ ØØ Ø1 FA ØØ 1Ø D6
Ø8C9:ØØ AØ ØØ ØØ ØØ ØØ ØØ ØØ Ø2
Ø8D1:ØØ Ø1 2Ø ØØ ØØ ØØ ØØ ØØ 26
Ø8D9:2Ø E3 Ø8 2Ø 29 Ø9 2Ø 4E F1
Ø8E1:Ø9 6Ø AD ØE DC 29 FE 8D 3C
Ø8E9:ØE DC A5 Ø1 29 FB 85 Ø1 42
Ø8F1:A9 D1 85 FC A9 39 85 FE Ø8
Ø8F9:AØ ØØ 84 FB 84 FD B1 FB 26
Ø9Ø1:91 FD 88 DØ F9 C6 FC C6 25
Ø9Ø9:FE A9 37 C5 FE DØ EF A5 Ø9
Ø911:Ø1 Ø9 Ø4 85 Ø1 AD ØE DC 76
Ø919:Ø9 Ø1 8D ØE DC AD 18 DØ 21
Ø921:29 FØ Ø9 ØE 8D 18 DØ 6Ø D4
Ø929:A9 39 85 FC 85 FE A9 5Ø AA
Ø931:85 FB A9 DØ 85 FD 2Ø 44 EF
Ø939:Ø9 A9 18 85 FB A9 D8 85 53
Ø941:FD C6 FE AØ 27 B1 FB 91 77
Ø949:FD 88 1Ø F9 6Ø A9 FF 8D 55
Ø951:Ø3 38 A9 FØ 8D Ø2 38 A9 C5
Ø959:ØF 8D Ø1 38 A2 ØØ 8E ØØ 2C
Ø961:38 86 FB 8A 2Ø 75 Ø9 8A CC
Ø969:2Ø 72 Ø9 E8 EØ 1Ø DØ F3 B4
Ø971:6Ø EA 4A 4A 29 Ø3 A8 B9 BC
Ø979:ØØ 38 AØ Ø3 91 FB 88 1Ø 7B
Ø981:FB E6 FB E6 FB E6 FB E6 93
Ø989:FB 6Ø A9 36 85 FC A9 ØØ BD
Ø991:85 FB A8 AA EØ BC DØ Ø1 C1
Ø999:6Ø BD 19 Ø8 48 4A 38 6A 35
Ø9A1:4A 4A 91 FB 2Ø B5 Ø9 68 AF
Ø9A9:29 ØF Ø9 2Ø 91 FB 2Ø B5 A9
Ø9B1:Ø9 E8 DØ EØ C9 2Ø DØ Ø4 1F
Ø9B9:Ø9 CØ 91 FB C8 CØ 19 FØ DE
Ø9C1:Ø1 6Ø A9 ØØ 91 FB A8 18 87
Ø9C9:A9 1A 65 FB 85 FB 9Ø Ø2 E2
Ø9D1:E6 FC 6Ø ØC Ø1 Ø3 Ø2 ØE 89
Ø9D9:AØ Ø4 B9 D4 Ø9 99 2Ø DØ 81
Ø9E1:88 1Ø F7 AD 11 DØ Ø9 4Ø 34
Ø9E9:8D 11 DØ 6Ø 2Ø F7 Ø9 2Ø 3A
Ø9F1:6E ØA 2Ø B9 ØA 6Ø A9 93 16
Ø9F9:2Ø D2 FF AØ Ø3 2Ø 57 ØA 2C
ØAØ1:A9 9Ø 2Ø D2 FF A9 AB 2Ø 5D
ØAØ9:D2 FF A9 A3 2Ø 65 ØA A9 4A
ØA11:A7 2Ø D2 FF A2 ØF AØ Ø3 FØ
ØA19:2Ø 52 ØA A9 AA 2Ø D2 FF 29
ØA21:A9 9A 2Ø D2 FF A9 AØ 2Ø E9
ØA29:65 ØA A9 9Ø 2Ø D2 FF A9 A6
ØA31:A5 2Ø D2 FF CA DØ DF AØ 74
ØA39:Ø3 2Ø 52 ØA A9 AE 2Ø D2 DC
ØA41:FF A9 AC 2Ø 65 ØA A9 AD AB
```

```
ØA49:2Ø D2 FF A9 92 2Ø D2 FF 77
ØA51:6Ø A9 ØD 2Ø D2 FF A9 2Ø AD
ØA59:2Ø D2 FF 88 DØ FA A9 12 92
ØA61:2Ø D2 FF 6Ø AØ 19 2Ø D2 BC
ØA69:FF 88 DØ FA 6Ø A9 Ø4 85 AØ
ØA71:FE A9 2C 85 FD A9 36 85 D5
ØA79:FC A9 ØØ 85 FB A9 ØØ A8 FD
ØA81:A2 ØF B1 FB DØ Ø7 2Ø AØ 24
ØA89:ØA CA DØ F6 6Ø 91 FD C8 EC
ØA91:DØ FØ 29 3F AA BD CØ 37 68
ØA99:29 CØ 11 F7 91 F7 6Ø 18 59
ØAA1:A9 1A 65 FB 85 FB 9Ø Ø2 BC
ØAA9:E6 FC A9 28 18 65 FD 85 FF
ØAB1:FD 9Ø Ø2 E6 FE AØ ØØ 6Ø 72
ØAB9:A9 ØF 85 F9 A9 D8 85 FE 71
ØAC1:A9 2C 85 FD 85 F7 A9 Ø4 A9
ØAC9:85 F8 A9 22 85 FC A9 AD 57
ØAD1:85 FB AØ 18 B1 FB C9 ØØ 4E
ØAD9:FØ 2B 85 Ø2 29 3F AA BD 5B
ØAE1:CØ 37 29 ØF 91 FD A9 CØ D2
ØAE9:24 Ø2 FØ 19 3Ø Ø8 BD 78 D5
ØAF1:22 2Ø 93 ØA DØ ØF 5Ø Ø7 9C
ØAF9:A9 CØ 2Ø 9B ØA DØ Ø6 BD 2E
ØBØ1:79 22 2Ø 93 ØA EA 88 1Ø B6
ØBØ9:CB A9 19 18 65 FB 85 FB 36
ØB11:9Ø Ø2 E6 FC C6 F9 DØ Ø1 5D
ØB19:6Ø A9 28 18 65 F7 85 F7 5E
ØB21:9Ø Ø2 E6 F8 A9 28 18 65 EF
ØB29:FD 85 FD 9Ø A5 E6 FE DØ FF
ØB31:A1 AD 12 DØ 48 65 A2 4A 3A
ØB39:4A 4A A8 68 E5 A2 4A 8D 7E
ØB41:2Ø DØ 8C 24 DØ 6Ø 1F 43 F8
ØB49:41 4D 5Ø 41 49 47 4E 2Ø 95
ØB51:4D 41 4E 41 47 45 52 ØD 3D
ØB59:ØØ A2 12 AØ Ø8 18 2Ø FØ 36
ØB61:FF A2 ØØ BD 47 ØB FØ Ø6 4A
ØB69:2Ø D2 FF E8 DØ F5 AØ Ø5 77
ØB71:A9 Ø1 8D 86 Ø2 A9 12 2Ø B1
ØB79:D2 FF A2 28 AD 86 Ø2 49 A4
ØB81:Ø3 8D 86 Ø2 A9 A3 2Ø D2 5C
ØB89:FF CA DØ FA 88 DØ EB A9 25
ØB91:92 4C D2 FF A9 92 85 FE FF
ØB99:A9 ØØ 85 FD A2 ØØ E8 EC E8
ØBA1:89 24 FØ 2F BD 89 24 85 78
ØBA9:F9 29 Ø7 85 F7 A5 F9 4A D4
ØBB1:4A 4A 4A 29 Ø7 85 F8 AØ 3C
ØBB9:Ø2 2Ø E6 ØB A5 F7 AØ Ø1 B5
ØBC1:2Ø E6 ØB A9 FF AØ Ø5 91 BB
ØBC9:FD A9 Ø5 18 65 FD 85 FD 97
ØBD1:4C 9F ØB A9 ØØ AA A8 B9 A7
```

```
ØBD9:44 22 9D ØØ 78 E8 E8 C8 5Ø
ØBE1:CØ 34 DØ F3 6Ø 91 FD C8 CC
ØBE9:C8 91 FD 6Ø A9 91 85 F8 26
ØBF1:A9 ØØ 85 F7 E6 F7 85 FE 2E
ØBF9:AA A2 ØØ BD BD 24 85 F9 6D
ØCØ1:4A 4A 4A 4A 85 FA BD FØ 43
ØCØ9:24 85 FB 4A 4A 85 FC 4A 65
ØC11:4A 85 FD AØ Ø4 A2 Ø4 B5 E1
ØC19:F9 48 29 Ø3 18 69 Ø1 91 8F
ØC21:F7 68 4A 4A 29 Ø3 18 69 2C
ØC29:Ø3 ØA ØA ØA ØA 11 F7 91 3D
ØC31:F7 88 CA 1Ø E2 E6 F7 AØ Ø5
ØC39:Ø2 A9 ØF 31 F7 AA E8 8A 78
ØC41:ØA ØA ØA ØA 85 Ø2 8A Ø5 11
ØC49:Ø2 91 F7 88 DØ EB E6 F7 4A
ØC51:E6 F7 E6 F7 E6 F7 E6 FE 1B
ØC59:A6 FE EØ 33 DØ 9D 6Ø A9 3B
ØC61:FF 8D ØF D4 A9 8Ø 8D 12 88
ØC69:D4 8D 18 D4 6Ø A2 4Ø A9 57
ØC71:ØØ 9D ØØ 8F 9D 4Ø 8F CA C1
ØC79:DØ F7 A9 8Ø 8D 8A Ø2 A9 79
ØC81:Ø8 2Ø D2 FF 2Ø 95 ØB 2Ø 8D
ØC89:FA 1A 2Ø 6C 1B 2Ø 8Ø 17 E1
ØC91:2Ø 8B Ø9 2Ø 1E 1C 2Ø D9 3B
ØC99:Ø8 2Ø ED ØB 2Ø D9 Ø9 2Ø C6
ØCA1:ED Ø9 A9 9E 2Ø D2 FF 2Ø 7E
ØCA9:5A ØB 2Ø 1E 14 2Ø 32 ØB 28
ØCB1:2Ø 1A 1F AD 6B 1F FØ F5 CE
ØCB9:2Ø D9 Ø9 2Ø 6Ø ØC A2 Ø4 F7
ØCC1:AØ Ø5 2Ø A3 1C 8D 15 8F 7A
ØCC9:8D 23 25 A2 Ø7 AØ Ø9 2Ø 2D
ØCD1:A3 1C A2 ØØ AØ ØØ C9 ØØ AF
ØCD9:FØ Ø7 29 Ø1 FØ Ø2 CA C8 4F
ØCE1:88 8E ØF 8F 8C 4F 8F 2Ø 9D
ØCE9:2E 11 DØ Ø3 4C 9D ØC 6Ø F8
ØCF1:A9 ØØ 8D 24 25 A9 8Ø 85 29
ØCF9:F7 A9 8F 85 F8 A9 Ø5 85 CØ
ØDØ1:Ø2 AØ Ø5 A2 Ø3 AD 1B D4 E8
ØDØ9:29 Ø3 95 F9 CA DØ F6 A9 FB
ØD11:Ø1 25 FA 18 69 Ø1 65 FB EB
ØD19:65 FC 91 F7 88 DØ E4 AØ C8
ØD21:Ø6 AD 1B D4 29 Ø3 AA CØ C5
ØD29:Ø8 FØ ØA CØ Ø9 FØ Ø6 AD 96
ØD31:15 8F FØ Ø2 E8 E8 E8 8A 3F
ØD39:91 F7 C8 CØ ØB DØ E2 AD 4E
ØD41:1B D4 29 3F FØ F9 C9 34 6E
ØD49:BØ F5 91 F7 C8 AD ØF 8F 95
ØD51:91 F7 DØ Ø9 AD 1B D4 29 89
ØD59:Ø7 ØA ØA 91 F7 2Ø DC ØE DB
ØD61:C6 Ø2 DØ 9D AØ ØØ 8C 2D 9E
```

```
ØD69:11 A9 8Ø 85 F7 A9 8F 85 E9
ØD71:F8 AD 2D 11 C9 Ø5 BØ EC DA
ØD79:AA FØ Ø6 2Ø DC ØE CA DØ 6D
ØD81:FA EE 2D 11 AØ Ø5 B1 F7 FF
ØD89:99 ØF 8F 88 DØ F8 AØ Ø6 6Ø
ØD91:A2 ØØ B1 F7 9D 1B 8F C8 F3
ØD99:E8 EØ Ø5 DØ F5 B1 F7 8D Ø2
ØDA1:ØC 8F 8D ØA 8F C8 B1 F7 F2
ØDA9:8D ØD 8F 2Ø E4 ØE 2Ø EE 5Ø
ØDB1:ØE DØ Ø3 4C 6A ØD 2Ø 2E 22
ØDB9:11 FØ A9 A2 ØØ 86 F8 AØ A4
ØDC1:Ø6 18 2Ø FØ FF AD 15 8F 68
ØDC9:CD 23 25 FØ Ø2 A2 ØC 86 8Ø
ØDD1:F7 BD 9E 14 FØ Ø6 2Ø D2 1F
ØDD9:FF E8 DØ F5 A9 3F 2Ø D2 Ø5
ØDE1:FF A6 F7 AØ ØA A9 2C 9D 9B
ØDE9:9E 14 E8 88 DØ F9 2Ø E4 91
ØDF1:FF FØ FB C9 ØD FØ 27 C9 A8
ØDF9:2Ø FØ Ø8 C9 41 9Ø EF C9 F3
ØEØ1:5B BØ EB E6 F8 A6 F8 EØ 18
ØEØ9:ØB FØ 13 A4 F7 99 9E 14 ØB
ØE11:29 3F 9D Ø5 Ø4 E6 F7 A9 EA
ØE19:2F 9D Ø6 Ø4 DØ DØ 2Ø 26 65
ØE21:ØF 2Ø 2E 11 FØ 95 AD ØF EB
ØE29:8F 29 Ø2 18 6D 1Ø 8F ØA ED
ØE31:6D 12 8F 8D 16 8F AD 11 AE
ØE39:8F ØA ØA 69 Ø9 38 ED 13 8F
ØE41:8F 8D 17 8F AD 1B D4 29 11
ØE49:1F ØA 6D 17 8F 69 2Ø 8D 86
ØE51:Ø8 8F AD ØF 8F 29 Ø4 6D 92
ØE59:13 8F ØA 6D 13 8F 6D 14 CØ
ØE61:8F 8D 18 8F ØA 6D 12 8F 5E
ØE69:69 3Ø 8D Ø9 8F AD 14 8F 73
ØE71:Ø9 Ø8 6D 1Ø 8F 8D 19 8F 37
ØE79:AD ØF 8F 29 Ø7 18 6D 12 3A
ØE81:8F 6D 11 8F 8D 1A 8F A2 72
ØE89:ØØ AD ØC 8F E8 DD 7F 24 6D
ØE91:BØ FA 8E 2Ø 8F 8E ØB 8F F4
ØE99:8E 21 8F 2Ø 84 1B AD 15 3A
ØEA1:8F CD 23 25 FØ Ø3 4C F1 CD
ØEA9:ØC AD ØF 8F 29 Ø3 8D 81 Ø4
ØEB1:8F 2Ø F3 1B A9 ØØ 8D 81 B7
ØEB9:8F AE 21 8F BD 7F 24 A8 43
ØEC1:CA BD 7F 24 AA CA 2Ø F7 9D
ØEC9:1B 2Ø 84 1B AD 15 8F CD 6C
ØED1:23 25 DØ D5 2Ø FA 1A 2Ø 81
ØED9:6C 1B 6Ø A9 1Ø 18 65 F7 3D
ØEE1:85 F7 6Ø 2Ø ED Ø9 2Ø CD 6E
ØEE9:15 2Ø 26 ØF 6Ø A9 ØF 85 9B
ØEF1:FD A9 16 85 FE A9 1D 85 FØ
```

```
ØEF9:A7 A2 FØ AØ 1Ø 2Ø B8 14 41
ØFØ1:AD 15 8F FØ ØD A2 ØA BD Ø1
ØFØ9:75 14 29 3F 9D 9C Ø6 CA 36
ØF11:DØ F5 AD 15 8F CD 23 25 3B
ØF19:FØ Ø3 EE 7D Ø6 A2 14 AØ A9
ØF21:15 2Ø A3 1C 6Ø AE 15 8F 7F
ØF29:BD 28 25 2Ø D2 FF A9 11 D1
ØF31:85 FD A9 19 85 FE A9 ØØ D3
ØF39:85 A7 A2 51 AØ 1Ø 2Ø B8 AB
ØF41:14 A9 1F 2Ø D2 FF A9 14 B7
ØF49:85 FD A9 19 85 FE A9 Ø9 F4
ØF51:85 A7 A2 AB AØ 1Ø 2Ø B8 69
ØF59:14 A2 13 E8 AØ ØF 18 2Ø AC
ØF61:FØ FF A2 31 8A 2Ø D2 FF D9
ØF69:E8 EØ 37 DØ F7 38 2Ø FØ F9
ØF71:FF EØ 18 DØ E6 AD ØC 8F 6D
ØF79:ØA AA BD DC 21 29 3F 8D 86
ØF81:AB Ø6 E8 BD DC 21 29 3F EC
ØF89:8D AC Ø6 A2 12 AØ Ø2 18 B3
ØF91:2Ø FØ FF AE 15 8F BD 28 71
ØF99:25 2Ø D2 FF AE ØD 8F 3Ø A5
ØFA1:ØE AØ Ø4 BD 31 1Ø 2Ø D2 28
ØFA9:FF E8 88 DØ F6 FØ ØD A2 58
ØFB1:ØØ BD E4 1Ø FØ Ø6 2Ø D2 8F
ØFB9:FF E8 DØ F5 A9 9E 85 F7 56
ØFC1:A9 14 85 F8 AØ ØØ AD 15 6F
ØFC9:8F CD 23 25 FØ Ø2 AØ ØC B6
ØFD1:B1 F7 FØ Ø6 2Ø D2 FF C8 5A
ØFD9:DØ F6 AD 15 8F FØ 2Ø A2 48
ØFE1:ØA BD 75 14 29 3F 9D F8 DE
ØFE9:Ø6 CA DØ F5 A9 14 85 FD DD
ØFF1:A9 16 85 FE A9 Ø9 85 A7 2F
ØFF9:A2 D2 AØ 1Ø 2Ø B8 14 A2 E1
1ØØ1:Ø4 AØ AØ BD 1Ø 8F Ø9 3Ø 3C
1ØØ9:99 27 Ø7 98 38 E9 28 A8 8C
1Ø11:CA 1Ø FØ A2 Ø4 AØ AØ 98 5F
1Ø19:18 7D 1B 8F A8 B9 2E Ø7 9Ø
1Ø21:Ø9 4Ø 99 2E Ø7 98 38 E9 EØ
1Ø29:28 29 F8 A8 CA 1Ø E8 6Ø 1A
1Ø31:53 45 4E 2Ø 47 4F 56 2Ø 5C
1Ø39:52 45 5Ø 2Ø 52 45 56 2Ø 54
1Ø41:2Ø 4D 53 2Ø 44 52 3E 2Ø 39
1Ø49:56 3D 5Ø 2Ø 47 45 4E 2Ø FB
1Ø51:1B 2C ØØ 1B 2C ØØ FF 2C 49
1Ø59:44 45 4D 4F 43 52 41 54 C5
1Ø61:49 43 2Ø 43 41 4E 44 49 44
1Ø69:44 41 54 45 2Ø 2C 2C 2C 11
1Ø71:2C 2C 2C ØØ FF 9C 2F 2Ø 29
1Ø79:43 48 41 52 2Ø 58 ØØ FF FC
1Ø81:2F 2Ø 53 54 41 4D 2Ø 58 C8
```

```
1Ø89:ØØ FF 2F 2Ø 49 4E 54 4C ØA
1Ø91:2Ø 58 ØØ FF 2F 2Ø 45 58 B4
1Ø99:5Ø 52 2Ø 58 ØØ FF 2F 2Ø 7E
1ØA1:41 5Ø 5Ø 4C 2Ø 58 ØØ ØØ A7
1ØA9:ØØ ØØ FF 1F 55 4E 45 4D 77
1ØB1:5Ø ØØ FF 5Ø 4F 56 54 59 D4
1ØB9:ØØ FF 41 47 52 49 43 ØØ B4
1ØC1:FF 45 44 55 43 4E ØØ FF 64
1ØC9:44 46 45 4E 53 ØØ ØØ ØØ C5
1ØD1:ØØ FF 1F 49 4E 46 4C 4E DC
1ØD9:ØØ FF 43 52 49 4D 45 ØØ 91
1ØE1:ØØ ØØ ØØ 9D 5Ø 52 45 53 85
1ØE9:49 44 45 4E 54 2Ø ØØ FF 7Ø
1ØF1:97 5Ø 4C 41 59 45 52 2Ø 34
1ØF9:5B 31 ØØ FF 44 45 4D 4F 35
11Ø1:43 52 41 54 49 43 ØØ FF 1E
11Ø9:43 4F 4E 56 45 4E 54 49 25
1111:4F 4E ØØ ØA A6 ØØ Ø5 2Ø 6E
1119:ØØ FF 1E 2Ø 2F 2Ø 4E 4F E6
1121:ØØ FF 2Ø 2F 2Ø 59 45 53 7E
1129:ØØ ØØ ØØ ØØ ØØ AD Ø5 Ø4 1Ø
1131:48 A9 ØØ 85 A2 85 C6 A9 9C
1139:2Ø C5 A2 DØ FC A2 17 BD 9C
1141:62 11 29 3F 9D Ø4 Ø4 CA C1
1149:1Ø F5 2Ø 1A 1F AD 6B 1F 3C
1151:FØ F8 A2 17 68 9D Ø4 Ø4 B5
1159:CA 1Ø FA AD 6B 1F 29 1Ø 59
1161:6Ø 3A 46 49 52 45 42 55 21
1169:54 54 4F 4E 2Ø 54 4F 2Ø AA
1171:43 4F 4E 54 49 4E 55 45 8B
1179:3A AD 23 25 CD 15 8F DØ 8D
1181:ØB EE 24 25 AD 24 25 C9 CD
1189:ØA DØ Ø1 6Ø 2Ø ED Ø9 2Ø F5
1191:CD 15 2Ø 26 ØF A9 Ø7 8D Ø1
1199:ØØ 8F 2Ø F4 14 A2 Ø5 AØ C8
11A1:ØC 2Ø A3 1C AA DØ Ø3 4C F2
11A9:Ø3 12 CA DØ Ø3 4C 93 12 BA
11B1:CA DØ Ø3 4C C5 12 CA DØ 6F
11B9:Ø3 4C EF 12 CA DØ Ø6 2Ø 55
11C1:2B 1D 4C 9B 11 CA DØ Ø8 61
11C9:2Ø 16 13 DØ CD 4C Ø3 12 A8
11D1:CA DØ ØE 2Ø 2E 11 FØ C5 AE
11D9:AD ØB 8F 8D 2Ø 8F 4C 8D C5
11E1:11 CA FØ Ø3 4C 9B 11 4C CC
11E9:C8 13 A2 ØØ A9 8Ø 18 6D 76
11F1:2Ø 8F A8 A9 ØØ 85 FD A9 73
11F9:ØE 85 FE A9 1E 85 A7 4C A1
12Ø1:B8 14 2Ø EB 11 AE 2Ø 8F 5C
12Ø9:BD 7F 24 CA 38 FD 7F 24 F9
1211:48 69 Ø3 A8 A2 Ø3 2Ø A3 A3
```

```
1219:1C C9 ØØ DØ Ø7 2Ø 2B 1D F6
1221:68 4C Ø6 12 85 Ø2 68 C5 39
1229:Ø2 BØ Ø3 4C 9B 11 C6 Ø2 5Ø
1231:AE 2Ø 8F CA BD 7F 24 18 9F
1239:65 Ø2 AE ØØ 8F 9D ØØ 8F E8
1241:85 FB 86 FC 2Ø FA 13 A9 84
1249:1E 2Ø D2 FF A5 FB ØA AA BA
1251:BD DC 21 2Ø D2 FF BD DD A1
1259:21 2Ø D2 FF A9 2Ø 2Ø D2 51
1261:FF BD ØØ 78 48 AA A9 ØØ BC
1269:2Ø CD BD 68 C9 ØA BØ Ø5 2C
1271:A9 2Ø 2Ø D2 FF A9 2Ø 2Ø AA
1279:D2 FF A9 98 2Ø D2 FF AD BF
1281:2Ø 8F Ø9 3Ø 2Ø D2 FF CE D8
1289:ØØ 8F DØ Ø3 4C Ø7 14 4C CE
1291:Ø6 12 AE ØØ 8F A9 FØ 9D B5
1299:ØØ 8F 86 FC 2Ø FA 13 A9 FE
12A1:81 2Ø D2 FF A2 ØØ BD 7Ø E9
12A9:15 FØ Ø6 2Ø D2 FF E8 DØ 9Ø
12B1:F5 AD 2Ø 8F Ø9 3Ø 2Ø D2 55
12B9:FF CE ØØ 8F DØ Ø3 4C Ø7 BC
12C1:14 4C 9B 11 AE ØØ 8F A9 C5
12C9:FF 9D ØØ 8F 86 FC 2Ø FA B1
12D1:13 A9 9A 2Ø D2 FF A2 ØØ 1B
12D9:BD 7D 15 FØ Ø6 2Ø D2 FF 44
12E1:E8 DØ F5 CE ØØ 8F DØ Ø3 3D
12E9:4C Ø7 14 4C 9B 11 AE ØØ BB
12F1:8F A9 ØØ 9D ØØ 8F 86 FC 6A
12F9:2Ø FA 13 A9 9B 2Ø D2 FF EC
13Ø1:A9 5A A2 Ø5 2Ø D2 FF CA 4E
13Ø9:DØ FA CE ØØ 8F DØ Ø3 4C 42
1311:Ø7 14 4C 9B 11 AD Ø9 8F E3
1319:C9 ØA BØ Ø1 6Ø CE Ø9 8F AC
1321:2Ø 99 21 AE 2Ø 8F BD 7F Ø7
1329:24 85 F8 CA BD 7F 24 85 48
1331:F7 A9 96 85 F9 A9 Ø4 85 EC
1339:FA A9 2F 85 FB 85 FC A6 1C
1341:F7 E4 F8 DØ Ø3 4C 2E 11 7F
1349:A5 F9 18 69 28 85 F9 9Ø 36
1351:Ø2 E6 FA BD ØØ 9Ø 85 FD B8
1359:85 FE A2 Ø4 Ø6 FE CA DØ 29
1361:FB Ø6 FE BØ 1C A9 25 85 49
1369:FB Ø6 FE BØ 14 A9 2Ø 85 Ø7
1371:FB Ø6 FE BØ ØC A9 25 85 D8
1379:FC Ø6 FE BØ Ø4 A9 2Ø 85 17
1381:FC AØ ØØ A9 2Ø 91 F9 C8 EC
1389:A5 FB 91 F9 C8 A5 FC 91 BB
1391:F9 A9 2F 85 FB 85 FC Ø6 53
1399:FD BØ 1C A9 2A 85 FC Ø6 7Ø
13A1:FD BØ 14 A9 2Ø 85 FC Ø6 27
```

```
13A9:FD B0 0C A9 2A 85 FB 06 7C
13B1:FD B0 04 A9 20 85 FB A0 CD
13B9:07 A5 FB 91 F9 C8 A5 FC A0
13C1:91 F9 E6 F7 4C 3A 13 20 1D
13C9:67 17 AE 20 8F E8 E8 E8 1C
13D1:A0 1F 18 20 F0 FF A9 3A 2A
13D9:20 D2 FF A2 03 A0 0D 20 C3
13E1:A3 1C C9 00 D0 06 20 2B 24
13E9:1D 4C C8 13 C9 0A D0 03 17
13F1:4C 9B 11 8D 20 8F 4C 9B 93
13F9:11 A9 16 38 E5 FC AA A0 72
1401:20 18 20 F0 FF 60 20 2E 42
1409:11 D0 03 4C D9 11 20 68 CE
1411:19 20 FA 1A 20 6C 1B 20 D7
1419:84 1B 4C 7A 11 A9 00 85 30
1421:FD A9 0A 85 FE A9 1E 85 AC
1429:A7 A2 31 A0 14 4C B8 14 55
1431:FF 12 90 A0 D5 D3 C5 A0 24
1439:CA CF D9 BD A0 00 FF A0 77
1441:D3 D4 C9 C3 CB A0 CF D2 51
1449:A0 00 FF C9 CA CB CC 92 10
1451:CD 12 A0 CB C5 D9 D3 00 F2
1459:FF 12 9B 50 4C 5B 31 20 D0
1461:50 41 52 54 59 92 00 FF A6
1469:1F 44 45 4D 4F 43 52 41 1D
1471:54 49 43 00 FF 52 45 50 A2
1479:55 42 4C 49 43 41 4E 00 B6
1481:FF 12 9B 20 49 4E 43 55 03
1489:4D 42 45 4E 54 92 00 FF 63
1491:1F 20 20 20 4E 4F 4E 45 E8
1499:20 20 20 00 FF 20 50 4C 4B
14A1:41 59 45 52 20 31 20 00 94
14A9:FF 20 50 4C 41 59 45 52 F4
14B1:20 32 20 00 00 00 00 86 01
14B9:FB 84 FC D0 0B C8 98 18 72
14C1:65 FB 85 FB 90 02 E6 FC 63
14C9:A6 FD E4 FE D0 01 60 E6 83
14D1:FD A4 A7 18 20 F0 FF A0 FD
14D9:00 A2 FF B1 FB 10 10 C8 CE
14E1:B1 FB F0 D9 20 D2 FF CA B4
14E9:10 FA F0 D1 C8 D0 F1 AA 2C
14F1:C8 D0 ED A9 00 85 FD A9 C6
14F9:0E 85 FE A9 92 20 D2 FF BF
1501:A9 90 20 D2 FF A9 1E 85 BD
1509:A7 A2 48 A0 15 20 B8 14 71
1511:AE 15 8F BD 25 25 29 3F F4
1519:8D 47 04 AD 24 25 09 30 2F
1521:8D 4E 04 AD 20 8F 0A 0A 5E
1529:0A 18 6D 20 8F AA AD 20 B0
1531:8F 09 30 8D 96 04 A0 00 4A
```

```
1539:BD 25 24 29 3F FØ Ø7 99 Ø8
1541:98 Ø4 E8 C8 DØ F2 6Ø Ø9 7E
1549:3A ØØ FF 2Ø 2Ø 2Ø 57 45 Ø8
1551:45 4B 2Ø 2Ø 2Ø ØØ Ø9 3A 44
1559:ØØ Ø9 2Ø ØØ Ø9 2Ø ØØ FF 92
1561:1F 2Ø 2Ø 43 41 4D 5Ø 41 7C
1569:49 47 4E ØØ FF 2Ø 2Ø 54 E8
1571:56 2Ø 41 44 53 2Ø 2Ø 9D 34
1579:ØØ FF 2Ø 2Ø 46 55 4E 44 12
1581:52 41 49 53 ØØ FF 2Ø 2Ø E3
1589:52 45 53 54 2Ø 2Ø 2Ø 2Ø BF
1591:ØØ FF 12 9B 2Ø 4D 41 5Ø CØ
1599:2Ø 2Ø 2Ø 2Ø 2Ø 2Ø ØØ FF 63
15A1:2Ø 5Ø 4F 4C 4C 2Ø 2Ø 2Ø E1
15A9:2Ø 2Ø ØØ FF 92 96 52 45 C4
15B1:43 4F 4E 53 49 44 45 52 88
15B9:ØØ FF 54 52 41 56 45 4C CD
15C1:2Ø 2Ø 2Ø 2Ø 9A ØØ Ø9 2Ø 11
15C9:ØØ ØØ ØØ ØØ A9 ØE 85 FD 82
15D1:A9 19 85 FE A9 1C 85 A7 28
15D9:A9 1E 2Ø D2 FF A2 22 AØ Ø1
15E1:17 2Ø B8 14 A9 2B 8D 4C 59
15E9:Ø6 A9 2E 8D 8C Ø7 AD 24 2Ø
15F1:25 DØ ØB A9 14 A2 Ø8 2Ø 3A
15F9:D2 FF CA DØ FA 6Ø AE Ø8 B2
16Ø1:8F A9 ØØ 2Ø CD BD A2 17 23
16Ø9:AØ 1E 18 2Ø FØ FF A9 1E ØB
1611:2Ø D2 FF AE Ø9 8F EØ 64 99
1619:BØ 16 A9 2Ø 2Ø D2 FF EØ 87
1621:ØA BØ ØD 2Ø D2 FF 2Ø D2 CB
1629:FF 8A Ø9 3Ø 2Ø D2 FF 6Ø C8
1631:A9 ØØ 2Ø CD BD AD B8 Ø7 3Ø
1639:8D B9 Ø7 A2 Ø6 AD 1B D4 97
1641:29 ØF C9 ØA BØ F7 Ø9 3Ø 47
1649:9D B9 Ø7 CA DØ EF A9 3C 16
1651:8D B8 Ø7 8D BC Ø7 2Ø 7A E8
1659:16 A9 34 85 F8 A9 ØØ 85 CD
1661:F7 A8 A2 Ø2 95 FC CA 1Ø 6E
1669:FB 2Ø A6 16 A9 2Ø A2 Ø2 E6
1671:95 F9 CA 1Ø FB 2Ø D3 16 5F
1679:6Ø A9 11 85 FC A9 ØØ 85 CE
1681:FB A6 FB E4 FC DØ Ø1 6Ø BØ
1689:AØ ØØ 18 2Ø FØ FF A2 ØØ D7
1691:BD 28 25 2Ø D2 FF A9 25 5C
1699:2Ø D2 FF E8 EØ Ø3 DØ FØ BE
16A1:E6 FB 4C 82 16 A6 F7 E8 16
16A9:E8 86 F7 C8 C4 F8 DØ Ø1 24
16B1:6Ø BD ØØ 78 4A 85 Ø2 B9 2B
16B9:ØØ 9Ø A2 Ø2 29 EE FØ Ø6 6B
16C1:CA 29 ØE FØ Ø1 CA B5 FC ØA
```

```
16C9:18 65 Ø2 BØ D8 95 FC 4C ØA
16D1:A6 16 A9 Ø4 85 F8 A9 ØØ AF
16D9:85 F7 AØ Ø2 A9 ØF 85 Ø2 91
16E1:B9 25 25 29 3F 91 F7 88 24
16E9:1Ø F6 A9 28 18 65 F7 85 5F
16F1:F7 A9 ØØ 65 F8 85 F8 AØ 4B
16F9:Ø2 A2 Ø2 A9 1Ø 18 75 FC 73
17Ø1:95 FC BØ ØD B5 F9 91 F7 DØ
17Ø9:CA 88 1Ø EF C6 Ø2 1Ø DA F8
1711:6Ø 48 A9 25 95 F9 68 4A B8
1719:4A 4A 29 Ø1 Ø9 24 4C Ø7 AC
1721:17 ØB 23 ØØ FF 2A 2Ø 53 3E
1729:ØØ FF 2A 2Ø 4D ØØ FF 2A 33
1731:2Ø 54 ØØ FF 2A 2Ø 57 ØØ Ø5
1739:FF 2A 2Ø 54 ØØ FF 2A 2Ø AF
1741:46 ØØ FF 2A 2Ø 53 ØØ ØB 8E
1749:2C ØØ FF 5C 2Ø 3Ø 3Ø 3C B1
1751:3Ø 3Ø 3Ø 3C 3Ø 3Ø 3Ø ØØ 1Ø
1759:FF 9A 2Ø 48 45 41 4C 54 D2
1761:48 2Ø ØØ ØØ ØØ ØØ A9 1C 2B
1769:2Ø D2 FF A9 ØØ 85 FD A9 B2
1771:ØF 85 FE A9 1E 85 A7 A2 FB
1779:ØØ AØ 8Ø 2Ø B8 14 6Ø A9 62
1781:8Ø 85 F8 A9 ØØ 85 F7 A9 BA
1789:ØØ 85 FA 85 F9 A9 ØØ 85 CC
1791:FD 85 FE A8 A2 Ø1 2Ø F1 D5
1799:17 A2 ØØ AØ Ø4 BD 25 24 8B
17A1:91 F7 C8 E8 EØ Ø8 DØ F5 FC
17A9:A2 Ø8 AC 36 19 88 E6 FD DA
17B1:A5 FD C9 ØA DØ Ø3 4C E1 19
17B9:17 A9 FF 91 F7 C8 A9 1C 49
17C1:91 F7 C8 A9 2A 91 F7 C8 BA
17C9:A5 FD Ø9 3Ø 91 F7 C8 A9 16
17D1:1F 91 F7 C8 E8 BD 25 24 2C
17D9:91 F7 DØ F7 C8 4C AF 17 56
17E1:2Ø Ø2 18 A9 ØØ A2 Ø4 91 62
17E9:F7 C8 CA DØ FA 4C 11 18 EF
17F1:A2 Ø1 AØ ØØ BD 36 19 91 5Ø
17F9:F7 C8 E8 EC 36 19 DØ F4 EE
18Ø1:6Ø A2 Ø1 BD 59 19 91 F7 5Ø
18Ø9:C8 E8 EC 59 19 DØ F4 6Ø 61
1811:A9 Ø1 85 FD 85 FE DØ Ø9 B9
1819:E6 FD A5 FD C9 ØA DØ Ø1 E9
1821:6Ø E6 F8 A9 Ø9 18 65 F9 62
1829:85 F9 A9 ØØ 65 FA 85 FA EC
1831:2Ø F1 17 A6 F9 AØ Ø4 A5 3B
1839:FD 49 3Ø 91 F7 C8 C8 BD ØC
1841:25 24 FØ Ø6 91 F7 E8 C8 92
1849:DØ F5 A6 FD BD 7F 24 85 CD
1851:FA AC 36 19 88 A5 FE ØA 65
```

```
1859:AA A9 30 85 FB 85 FC A9 41
1861:FF 91 F7 C8 A9 1C 91 F7 5A
1869:C8 A9 2A 91 F7 C8 A9 9A 97
1871:91 F7 C8 A5 FE C9 0A 90 9F
1879:07 E6 FC E9 0A 4C 76 18 AB
1881:65 FB 85 FB A5 FC 91 F7 10
1889:C8 A5 FB 91 F7 C8 A9 20 76
1891:91 F7 C8 A9 97 91 F7 C8 F7
1899:BD DC 21 91 F7 C8 E8 BD 8F
18A1:DC 21 91 F7 C8 CA A9 20 1F
18A9:91 F7 C8 A9 30 85 FB 85 69
18B1:FC BD 00 78 C9 0A 90 07 F5
18B9:E6 FC E9 0A 4C B5 18 65 49
18C1:FB 85 FB A5 FC 91 F7 C8 12
18C9:A5 FB 91 F7 C8 A9 20 91 3C
18D1:F7 C8 A9 00 91 F7 C8 E6 4A
18D9:FE A5 FE C5 FA F0 03 4C 1D
18E1:56 18 20 02 18 A5 FA 85 3A
18E9:FE A6 FD CA BD 7F 24 85 69
18F1:02 E8 BD 7F 24 38 E5 02 DC
18F9:85 02 A9 08 E5 02 85 02 67
1901:30 26 A9 09 91 F7 C8 A9 42
1909:23 91 F7 C8 A9 00 91 F7 25
1911:C8 C6 02 30 13 A9 09 91 7F
1919:F7 C8 A9 20 91 F7 C8 A9 58
1921:00 91 F7 C8 C6 02 10 ED 8F
1929:A9 00 A2 04 91 F7 C8 CA 8D
1931:D0 FA 4C 19 18 23 09 23 28
1939:00 FF 20 20 20 20 20 20 53
1941:20 20 20 20 00 09 2C 00 0E
1949:FF 12 9A 25 90 CD C1 D0 B5
1951:A0 A0 A0 A0 A0 A0 92 00 C6
1959:0E FF 1C 2A 4D 45 4E 55 2A
1961:20 20 20 20 20 00 00 AD 60
1969:0B 8F 8D 20 8F A9 08 8D 79
1971:00 8F CE 00 8F D0 01 60 83
1979:AE 00 8F BD 00 8F D0 09 B9
1981:20 B1 1A 20 D0 1A 4C 73 70
1989:19 10 17 6A B0 03 4C A6 A6
1991:1B AD 18 8F 0A 6D 09 8F 60
1999:90 02 A9 FF 8D 09 8F 4C C5
19A1:73 19 AC 09 8F F0 CB 48 1A
19A9:A2 00 E8 DD 7F 24 B0 FA 11
19B1:EC 0B 8F F0 09 8E 20 8F 70
19B9:8E 0B 8F CE 09 8F 68 20 4C
19C1:CF 19 20 E9 19 20 5D 1A E2
19C9:20 8C 1A 4C 73 19 85 02 44
19D1:85 FB C6 FB A5 FB 0A 0A 99
19D9:18 65 FB 85 FB 85 FD A9 E4
19E1:92 85 FC A9 91 85 FE 60 F9
```

```
19E9:AD Ø8 8F 29 F8 DØ Ø5 A9 38
19F1:Ø1 85 FF 6Ø A9 Ø3 18 6D Ø3
19F9:15 8F A8 B1 FB 85 FF AD 6E
1AØ1:ØA 8F 1Ø Ø3 E6 FF 6Ø C5 ØE
1AØ9:Ø2 DØ Ø9 A9 Ø2 2Ø 57 1A 87
1A11:A9 FF 85 Ø2 A5 Ø2 8D ØA 45
1A19:8F AD Ø8 8F AØ Ø5 4A 88 BØ
1A21:DØ FC 2Ø 57 1A AD 16 8F B9
1A29:2Ø 57 1A AØ Ø6 88 DØ Ø1 85
1A31:6Ø B9 1A 8F D1 FD DØ Ø7 6F
1A39:A9 Ø3 2Ø 57 1A DØ EE AA 19
1A41:CA 8A D1 FD DØ Ø7 A9 Ø1 8E
1A49:2Ø 57 1A DØ EØ E8 E8 8A BA
1A51:D1 FD DØ D9 FØ FØ 18 65 86
1A59:FF 85 FF 6Ø AØ Ø5 B1 FB 6D
1A61:38 E5 FF BØ Ø4 C6 FF DØ 42
1A69:F3 91 FB A5 FF AA AC 15 EE
1A71:8F C8 18 71 FB 9Ø Ø2 A9 89
1A79:FF 91 FB 98 49 Ø3 A8 8A 4D
1A81:4A 4A 71 FB 9Ø Ø2 A9 FF 3B
1A89:91 FB 6Ø 46 FF DØ Ø1 6Ø 9B
1A91:AD Ø8 8F 38 E5 FF BØ Ø2 A6
1A99:A9 ØØ 8D Ø8 8F 46 FF DØ 3B
1AA1:Ø1 6Ø AD Ø9 8F 38 E5 FF DD
1AA9:BØ Ø2 A9 ØØ 8D Ø9 8F 6Ø FB
1AB1:AØ ØØ A2 ØF AD 15 8F FØ 4D
1AB9:Ø2 A2 FØ 86 FB A2 34 CA BB
1AC1:DØ Ø3 84 Ø2 6Ø BD ØØ 9Ø 5A
1AC9:25 FB FØ F3 C8 DØ FØ A5 FD
1AD1:Ø2 18 6D 17 8F ØA 6D 16 C1
1AD9:8F 6D Ø8 8F 9Ø Ø3 18 A9 95
1AE1:FF 8D Ø8 8F AD ØA 8F DØ F8
1AE9:ØA A9 1Ø 6D Ø8 8F BØ Ø3 49
1AF1:8D Ø8 8F A9 ØØ 8D ØA 8F 55
1AF9:6Ø A9 92 85 FC A9 ØØ 85 87
1BØ1:FB A9 ØØ AA FØ Ø7 AØ Ø5 34
1BØ9:E6 FB 88 DØ FB E8 EØ 34 49
1B11:DØ Ø1 6Ø AØ Ø1 B1 FB C8 95
1B19:38 F1 FB DØ Ø6 2Ø 58 1B FØ
1B21:4C Ø7 1B BØ ØA EA 49 FF 3C
1B29:18 69 Ø1 AØ 8Ø DØ Ø2 AØ DB
1B31:Ø8 85 FD 84 FE 29 EØ FØ 24
1B39:Ø2 DØ 14 46 FE A5 FD 29 3F
1B41:1Ø FØ Ø2 DØ ØA 46 FE A5 16
1B49:FD 29 Ø8 DØ Ø2 46 FE A5 A3
1B51:FE 9D ØØ 9Ø 4C Ø7 1B AD D9
1B59:ØØ 9Ø 29 FØ FØ Ø4 A9 Ø1 D3
1B61:DØ Ø2 A9 1Ø 8D ØØ 9Ø 9D E1
1B69:ØØ 9Ø 6Ø A2 34 CA FØ 12 BA
1B71:BD ØØ 9Ø 29 ØF FØ Ø4 A9 19
```

```
1B79:43 DØ Ø2 A9 82 9D CØ 37 A3
1B81:DØ EB 6Ø AD 15 8F 48 A2 1C
1B89:3F BD 4Ø 8F 9D 8Ø 8F BD 9B
1B91:ØØ 8F 9D 4Ø 8F BD 8Ø 8F 67
1B99:9D ØØ 8F CA DØ EB 68 49 8D
1BA1:Ø1 8D 15 8F 6Ø AD Ø9 8F B2
1BA9:C9 28 9Ø 43 AE 2Ø 8F BD E7
1BB1:7F 24 85 FA CA BD 7F 24 81
1BB9:85 F9 C6 F9 AD 24 25 ØA FB
1BC1:18 6D 19 8F 85 FF E6 F9 6F
1BC9:A5 F9 C5 FA FØ ØE 2Ø 17 DØ
1BD1:1C 46 FF 2Ø 5D 1A 2Ø 8C C9
1BD9:1A 4C BD 1B AD Ø9 8F 38 82
1BE1:ED 19 8F 9Ø Ø5 ED 19 8F F1
1BE9:BØ Ø2 A9 Ø1 8D Ø9 8F 4C 3A
1BF1:73 19 A2 ØØ AØ 34 86 F9 59
1BF9:84 FA E6 F9 A5 F9 C5 FA 49
1CØ1:FØ 13 AE 81 8F 86 FF 2Ø 1B
1CØ9:17 1C 2Ø 2C 1A 46 FF 2Ø A4
1C11:5D 1A 4C FB 1B 6Ø 2Ø CF 32
1C19:19 2Ø 26 1A 6Ø A2 ØØ A9 83
1C21:ØØ 9D ØØ 3F CA DØ FA A9 ED
1C29:ØØ AA A8 B9 ØA 1F 9D ØØ C4
1C31:3F B9 12 1F 9D 4Ø 3F E8 Ø1
1C39:E8 E8 C8 CØ Ø7 DØ EC B9 54
1C41:ØA 1F 9D ØØ 3F 9D Ø1 3F AB
1C49:9D Ø2 3F B9 12 1F 9D 4Ø DC
1C51:3F A9 FC 8D F8 Ø7 A9 FD 41
1C59:8D F9 Ø7 A2 Ø7 A9 ØC 9D 76
1C61:27 DØ CA 1Ø FA A9 Ø1 8D C9
1C69:1D DØ A9 Ø1 8D 1Ø DØ A9 A1
1C71:Ø4 8D ØØ DØ A9 32 8D Ø1 4E
1C79:DØ A9 36 8D Ø2 DØ A9 38 Ø3
1C81:8D Ø3 DØ A9 ØØ AØ Ø4 99 1A
1C89:Ø2 1F 88 1Ø FA A9 22 8D EC
1C91:Ø7 1F A9 AD 8D Ø6 1F A9 91
1C99:36 8D Ø9 1F A9 ØØ 8D Ø8 D3
1CA1:1F 6Ø A9 ØØ 85 FD A9 Ø4 32
1CA9:8D ØØ DØ 98 2Ø FA 1E 85 FA
1CB1:FC 8A 2Ø FA 1E 85 FB 8D 4B
1CB9:Ø1 DØ A9 ØC 8D 27 DØ AD F4
1CC1:1Ø DØ Ø9 Ø1 8D 1Ø DØ AD 63
1CC9:15 DØ Ø9 Ø1 8D 15 DØ 2Ø 74
1CD1:1A 1F AD 6B 1F FØ F8 29 23
1CD9:13 FØ F4 AA 29 1Ø DØ 27 73
1CE1:8A 29 Ø1 FØ 11 AD Ø1 DØ EA
1CE9:C5 FB FØ E3 C6 FD 38 E9 E8
1CF1:Ø8 8D Ø1 DØ DØ D9 AD Ø1 Ø9
1CF9:DØ C5 FC FØ D2 E6 FD 18 Ø1
1DØ1:69 Ø8 8D Ø1 DØ DØ C8 A9 B8
```

```
1DØ9:ØØ 8D 27 DØ 2Ø 1A 1F AD ED
1D11:6B 1F FØ F8 29 1Ø DØ Ø7 A8
1D19:A9 ØC 8D 27 DØ DØ BØ AD 28
1D21:15 DØ 29 FE 8D 15 DØ A5 37
1D29:FD 6Ø A2 Ø7 BD Ø2 1F 95 Ø9
1D31:F7 CA 1Ø F8 A9 Ø1 8D 28 4Ø
1D39:DØ AD 15 DØ Ø9 Ø2 8D 15 77
1D41:DØ 2Ø 1A 1F AD 6B 1F FØ 6B
1D49:F8 6A BØ 14 6A BØ 43 6A F8
1D51:BØ 6E 6A BØ Ø5 6A BØ Ø5 1Ø
1D59:9Ø E7 4C FØ 1D 4C E7 1E 76
1D61:A5 F8 FØ DD AD Ø3 DØ 38 FB
1D69:E9 Ø4 8D Ø3 DØ C6 F8 A5 B4
1D71:F8 6A BØ Ø3 4C 24 1E A5 DD
1D79:FD E9 1A 85 FD BØ Ø2 C6 46
1D81:FE A5 FB 38 E9 19 85 FB 62
1D89:9Ø Ø3 4C 24 1E C6 FC 4C EA
1D91:24 1E A5 F8 C9 1D FØ A9 F7
1D99:AD Ø3 DØ 18 69 Ø4 8D Ø3 8Ø
1DA1:DØ E6 F8 A5 F8 6A BØ 7B C5
1DA9:A5 FD 69 1A 85 FD 9Ø Ø2 4C
1DB1:E6 FE A5 FB 18 69 19 85 B1
1DB9:FB 9Ø 68 E6 FC 4C 24 1E 11
1DC1:A5 F7 DØ Ø3 4C 42 1D AD 6A
1DC9:Ø2 DØ 38 E9 Ø4 8D Ø2 DØ ØA
1DD1:C6 F7 A5 F7 6A 9Ø 4C A5 75
1DD9:FD E9 Ø1 85 FD BØ Ø2 C6 83
1DE1:FE A5 FB 38 E9 Ø1 85 FB 62
1DE9:BØ 39 C6 FC 4C 24 1E A5 48
1DF1:F7 C9 31 DØ Ø3 4C 42 1D B8
1DF9:AD Ø2 DØ 18 69 Ø4 8D Ø2 9F
1EØ1:DØ E6 F7 A5 F7 6A 9Ø Ø3 46
1EØ9:4C 24 1E A5 FD 69 Ø1 85 AF
1E11:FD 9Ø Ø2 E6 FE A5 FB 18 BD
1E19:69 Ø1 85 FB 9Ø Ø5 E6 FC 1E
1E21:4C 24 1E A9 Ø1 85 F9 A5 A2
1E29:F8 4A 9Ø Ø4 Ø6 F9 Ø6 F9 E4
1E31:A5 F7 6A BØ Ø2 Ø6 F9 AØ 53
1E39:ØØ B1 FB 85 Ø2 A5 F9 31 85
1E41:FD DØ 26 A9 CØ 24 Ø2 3Ø DA
1E49:ØD A5 Ø2 29 3F AA BD 78 EØ
1E51:22 85 Ø2 4C 6A 1E 5Ø Ø7 78
1E59:A9 ØØ 85 Ø2 4C 6A 1E A5 29
1E61:Ø2 29 3F AA BD 79 22 85 19
1E69:Ø2 A2 ØF AØ 1Ø 18 2Ø FØ 4D
1E71:FF A9 95 2Ø D2 FF A9 2Ø D6
1E79:A2 Ø7 2Ø D2 FF CA 1Ø FA 4Ø
1E81:A9 9D A2 Ø7 2Ø D2 FF CA D5
1E89:1Ø FA A5 Ø2 DØ Ø3 4C 42 CE
1E91:1D 29 3F ØA AA BD DC 21 56
```

```
1E99:2Ø D2 FF BD DD 21 2Ø D2 FC
1EA1:FF A9 2Ø 2Ø D2 FF BD ØØ 6Ø
1EA9:78 AA C9 ØA BØ Ø5 A9 2Ø B3
1EB1:2Ø D2 FF A9 ØØ 2Ø CD BD 27
1EB9:A9 1D 2Ø D2 FF A9 9Ø 2Ø 2B
1EC1:D2 FF A9 12 2Ø D2 FF A9 B3
1EC9:AØ 2Ø D2 FF A5 Ø2 29 3F 7F
1ED1:A2 ØØ E8 DD 7F 24 BØ FA 43
1ED9:8A 69 BØ 2Ø D2 FF A9 92 4A
1EE1:2Ø D2 FF 4C 42 1D AD 15 9E
1EE9:DØ 29 FD 8D 15 DØ A2 Ø7 A9
1EF1:B5 F7 9D Ø2 1F CA 1Ø F8 18
1EF9:6Ø EA ØA ØA ØA 18 69 32 B8
1FØ1:6Ø ØØ ØØ ØØ ØØ ØØ ØØ ØØ 6F
1FØ9:ØØ CØ CØ EØ FØ EØ CØ C8 F2
1F11:FF FF 99 81 C3 C3 81 99 64
1F19:FF A9 ØØ 8D 6B 1F AD ØØ CD
1F21:DC 29 1F 49 1F DØ 2D AD D4
1F29:Ø1 DC 29 1F 49 1F DØ 24 C2
1F31:2Ø E4 FF DØ Ø1 6Ø 38 E9 A9
1F39:49 9Ø DE AA E8 E9 Ø5 BØ 7Ø
1F41:D8 8A 29 Ø2 FØ Ø4 8A 49 C9
1F49:Ø1 AA A9 ØØ 8D 6B 1F 38 78
1F51:2A CA DØ FC 8D 6B 1F AD 47
1F59:ØØ DC 2D Ø1 DC 29 1Ø FØ 21
1F61:F6 A9 Ø6 65 A2 C5 A2 DØ DE
1F69:FC 6Ø ØØ 2Ø FA 1A 2Ø B1 72
1F71:1A A5 Ø2 C9 1A 9Ø Ø3 2Ø 3C
1F79:84 1B 2Ø 7A 21 2Ø 84 1B 1A
1F81:2Ø 7A 21 A9 Ø1 2Ø CF 19 6E
1F89:AØ Ø5 B1 FB 4A 4A AA AØ CØ
1F91:Ø2 8A 18 71 FB 9Ø Ø2 A5 59
1F99:FF 91 FB 88 DØ F3 AØ Ø2 DD
1FA1:D1 FB DØ 13 AØ Ø3 B1 FB 83
1FA9:C8 38 F1 FB A9 8Ø 2A A8 A4
1FB1:C8 B1 FB E9 Ø1 91 FB A5 CA
1FB9:FB 18 69 Ø5 85 FB C9 FF 29
1FC1:DØ C6 2Ø FA 1A 2Ø B1 1A 9C
1FC9:A2 33 BD ØØ 9Ø 29 11 FØ 1A
1FD1:Ø3 1E ØØ 9Ø CA DØ F3 2Ø C3
1FD9:6C 1B 2Ø ED Ø9 2Ø 57 16 85
1FE1:2Ø 5A ØB 2Ø CD 15 A2 ØF 41
1FE9:86 Ø2 AØ 1D 18 2Ø FØ FF F4
1FF1:A9 98 2Ø D2 FF A9 2Ø A2 E5
1FF9:ØB 2Ø D2 FF CA DØ FA E6 96
2ØØ1:Ø2 A6 Ø2 EØ 18 DØ E3 AD B3
2ØØ9:64 Ø7 8D 8C Ø7 8D B4 Ø7 96
2Ø11:8D DC Ø7 A9 2Ø A2 ØB 9D ØA
2Ø19:DC Ø7 CA DØ FA A9 14 8D 24
2Ø21:E2 Ø7 A9 ØØ A2 Ø3 95 Ø3 19
```

```
2Ø29:CA 1Ø FB A9 Ø9 85 AE A9 52
2Ø31:ØØ 8D 2Ø 8F EE 2Ø 8F AD 96
2Ø39:2Ø 8F C9 ØA DØ Ø3 4C 3Ø A2
2Ø41:2Ø 2Ø 99 21 A9 ØØ 85 B2 E9
2Ø49:85 B3 A2 Ø4 86 FB AØ 1F 52
2Ø51:84 FC A9 BE 85 F7 85 F9 45
2Ø59:A9 Ø4 85 F8 85 FA A6 A7 BC
2Ø61:AØ Ø3 A9 2Ø 91 F7 88 1Ø 77
2Ø69:FB A5 F7 18 69 28 85 F7 8Ø
2Ø71:9Ø Ø2 E6 F8 CA DØ E9 AE Ø3
2Ø79:2Ø 8F BD 7F 24 85 FE CA 5D
2Ø81:BD 7F 24 85 FD A6 FB A4 84
2Ø89:FC 18 2Ø FØ FF A6 FD BD B5
2Ø91:ØØ 9Ø 29 ØF DØ Ø3 4C 6F A6
2Ø99:21 BD 44 22 AA 18 65 B2 B7
2ØA1:85 B2 8A C9 ØA BØ Ø5 A9 Ø6
2ØA9:2Ø 2Ø D2 FF A9 9A 2Ø D2 27
2ØB1:FF A9 ØØ 2Ø CD BD A6 FB ØD
2ØB9:AØ 25 18 2Ø FØ FF AØ Ø3 64
2ØC1:A9 2Ø 2Ø D2 FF 88 DØ FA CE
2ØC9:E6 FB E6 FD C6 A7 DØ B5 65
2ØD1:A5 AE DØ Ø3 4C 69 21 AD D2
2ØD9:2Ø 8F 18 69 ØE AA AØ 1F 23
2ØE1:18 2Ø FØ FF A9 9A 2Ø D2 1F
2ØE9:FF A5 B2 AA C9 ØA BØ Ø5 71
2ØF1:A9 2Ø 2Ø D2 FF A9 ØØ 2Ø Ø7
2ØF9:CD BD A9 9C 2Ø D2 FF A9 85
21Ø1:2Ø 48 2Ø D2 FF AD 2Ø 8F 1D
21Ø9:Ø9 3Ø 2Ø D2 FF 68 2Ø D2 C1
2111:FF 2Ø D2 FF A5 B3 AA C9 DØ
2119:ØA BØ Ø5 A9 2Ø 2Ø D2 FF EE
2121:A9 1C 2Ø D2 FF A9 ØØ 2Ø 37
2129:CD BD A2 18 AØ 1E 18 2Ø 65
2131:FØ FF A9 98 2Ø D2 FF A5 9C
2139:B2 18 65 Ø3 85 Ø3 A9 ØØ 43
2141:65 Ø4 85 Ø4 A5 B3 65 Ø5 F3
2149:85 Ø5 A9 ØØ 65 Ø6 85 Ø6 19
2151:A6 Ø3 A5 Ø4 2Ø CD BD A2 F2
2159:18 AØ 24 18 2Ø FØ FF A6 41
2161:Ø5 A5 Ø6 2Ø CD BD C6 AE F3
2169:2Ø 2E 11 4C 35 2Ø BD 44 18
2171:22 18 65 B3 85 B3 4C C9 1Ø
2179:2Ø AD 1A 8F 8D 81 8F 2Ø 25
2181:F3 1B A9 ØØ 8D 81 8F AE F9
2189:2Ø 8F BD 7F 24 A8 CA BD 86
2191:7F 24 AA CA 2Ø F7 1B 6Ø 16
2199:A9 9C 2Ø D2 FF 2Ø EB 11 72
21A1:A2 Ø3 AØ 1E 18 2Ø FØ FF ØF
21A9:2Ø C1 21 AE 2Ø 8F BD 7F B5
21B1:24 CA 38 FD 7F 24 85 A7 DE
```

```
21B9:69 Ø3 AA AØ 1E 2Ø FØ FF 24
21C1:A2 ØØ BD CF 21 DØ Ø1 6Ø B8
21C9:2Ø D2 FF E8 DØ F4 9A 44 33
21D1:45 4D 2Ø 2Ø 2Ø 2Ø 1C 52 1C
21D9:45 5Ø ØØ 2Ø 2Ø 4D 45 4E E3
21E1:48 56 54 4D 41 52 49 43 66
21E9:54 4E 59 4E 4A 5Ø 41 4F 5F
21F1:48 49 4E 49 4C 4D 49 57 8A
21F9:49 4D 4E 49 41 4D 4F 4E BE
22Ø1:44 53 44 4E 45 4B 53 44 EB
22Ø9:45 4D 44 44 43 56 41 57 5D
2211:56 4E 43 53 43 47 41 46 B1
2219:4C 4B 59 54 4E 41 4C 4D 24
2221:53 41 52 4C 41 4F 4B 54 AØ
2229:58 4D 54 49 44 57 59 43 81
2231:4F 4E 4D 41 5A 55 54 4E 8D
2239:56 57 41 4F 52 43 41 41 FE
2241:4B 48 49 ØØ Ø4 Ø4 Ø3 ØD A9
2249:Ø4 Ø8 24 1Ø 19 17 ØC 18 6C
2251:14 ØB ØA Ø8 ØB Ø3 Ø3 Ø5 93
2259:Ø7 Ø3 ØA Ø3 ØC Ø6 ØD Ø8 ED
2261:ØC 15 Ø9 ØB Ø9 Ø7 Ø6 ØA 3D
2269:Ø8 1D Ø4 Ø4 Ø3 Ø8 Ø5 Ø7 Ø3
2271:Ø5 Ø4 ØA Ø7 2F Ø3 Ø4 ØØ 78
2279:Ø1 Ø3 Ø3 Ø4 Ø5 Ø8 Ø9 Ø8 Ø2
2281:ØA 1A 1F ØB ØE 1Ø ØE ØC BE
2289:25 ØC 13 1Ø 11 14 19 Ø9 DA
2291:2Ø 19 19 1D 1B 1E 1A 1F C5
2299:1D 2Ø 11 22 26 23 28 28 EE
22A1:29 2A 26 25 2E 31 28 28 CA
22A9:2C ØØ ØØ ØØ EF EF 2F 6F 11
22B1:E7 E7 E7 E7 E7 D2 D2 D2 62
22B9:CF CF CF ØØ ØØ ØØ ØØ ØØ D3
22C1:ØØ ØØ ØØ C1 C1 EF 2F 2F 7D
22C9:6F 67 27 27 27 27 12 12 Ø3
22D1:12 ØF 4F 4F 4D CD CD ØØ FE
22D9:ØØ ØØ ØØ ØØ C1 C1 FØ 3Ø 45
22E1:3Ø 7Ø 67 27 27 27 27 13 FØ
22E9:13 13 ØF ØF 4F ØE CE CD 6D
22F1:ØØ ØØ ØØ C7 Ø7 42 C1 FØ 68
22F9:3Ø 3Ø 28 28 67 29 29 29 45
23Ø1:13 13 13 1Ø 1Ø 4E 8C CD 99
23Ø9:ØD CD C9 C7 Ø7 Ø7 Ø4 C4 2Ø
2311:F1 31 6E 2E 2E 2D 29 29 EE
2319:29 14 14 14 54 1Ø 5Ø ØC 4C
2321:4C ØB ØA 49 Ø9 Ø9 47 46 67
2329:C5 F1 31 6E 2E 2E 2D 2D 8D
2331:2A 2A 2A 95 95 95 11 91 6C
2339:ØC 4C ØB 8A 4A 89 58 97 63
2341:C8 ØØ ØØ 31 31 6E 2E 2D CB
```

```
2349:2D 2A 2A 2A 15 15 15 55 15
2351:11 91 4C 4B 1F 5F 5A 19 Ø7
2359:57 D6 ØØ ØØ F1 31 6E AC DE
2361:2C 2C 2B 2B AB 65 25 25 43
2369:51 63 11 6Ø 6Ø 6Ø 59 5B EB
2371:5B 5B ØØ ØØ ØØ ØØ F1 31 51
2379:71 2C 2C 2B 2B 2B 26 25 33
2381:25 A5 23 23 62 A1 AØ 5D 92
2389:9C 1B DB ØØ ØØ ØØ ØØ F1 52
2391:31 71 2C 2C 2B 2B 6B 26 18
2399:26 26 66 23 A3 22 21 61 C4
23A1:1D 5C 9C ØØ ØØ ØØ ØØ ØØ 21
23A9:ØØ ØØ ØØ EC EC EB E6 26 C9
23B1:26 26 26 26 24 64 22 21 D3
23B9:21 1D 1D ØØ ØØ ØØ ØØ ØØ 7B
23C1:F2 32 F2 ØØ F3 ØØ ØØ ØØ ØC
23C9:E6 E6 26 26 E6 24 E4 E2 D8
23D1:E1 DE DE 1E ØØ ØØ ØØ ØØ 7E
23D9:ØØ F2 32 32 ØØ ØØ F3 ØØ 2E
23E1:ØØ ØØ ØØ E6 E6 ØØ ØØ ØØ CD
23E9:ØØ ØØ ØØ ØØ DE DE ØØ ØØ A2
23F1:ØØ ØØ F2 F2 F2 F2 ØØ ØØ 29
23F9:F3 ØØ ØØ ØØ ØØ E6 ØØ ØØ D5
24Ø1:ØØ ØØ ØØ ØØ ØØ DE 1E DE DF
24Ø9:ØØ ØØ F2 ØØ ØØ ØØ F2 ØØ 95
2411:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 59
2419:ØØ ØØ ØØ ØØ ØØ ØØ ØØ DE 4Ø
2421:DE ØØ ØØ FF 2Ø 52 45 47 F4
2429:49 4F 4E 53 ØØ 4E 45 57 Ø4
2431:2Ø 45 4E 47 4C ØØ 55 52 78
2439:42 41 4E 2Ø 4E 45 ØØ 48 8E
2441:45 41 52 54 4C 4E 44 ØØ 3Ø
2449:47 2Ø 5Ø 4C 41 49 4E 53 2B
2451:ØØ 41 54 4C 41 4E 54 49 6E
2459:43 ØØ 53 4F 55 54 48 45 74
2461:52 4E ØØ 41 52 4B 4C 41 14
2469:54 45 58 ØØ 4D 4F 55 4E D8
2471:54 41 49 4E ØØ 5Ø 41 43 49
2479:49 46 49 43 2Ø ØØ Ø1 Ø7 5F
2481:ØA ØF 16 1F 23 27 2F 34 8F
2489:34 DC F3 F3 2F 3F DC 4E 5F
2491:E4 4D 4D E4 E3 CE 5C 3E E8
2499:F3 5C E3 F2 E3 F3 63 3F 93
24A1:2F E4 3F 45 64 BE 45 46 E5
24A9:64 4D 4D 46 46 D4 4E D4 7C
24B1:F3 F3 C5 D4 E4 F3 C5 EB 65
24B9:F2 E4 F2 2F 21 3B 21 F6 35
24C1:68 7E CA BD 24 61 59 BD 2A
24C9:DC 34 76 7A 51 26 Ø3 AB D8
24D1:BA EE FE CC AB Ø2 5Ø 46 2C
```

```
24D9:46 EB ØØ 91 45 Ø1 Ø1 86 1Ø
24E1:57 CB 61 6Ø 77 DF 42 EA A5
24E9:AA F6 F5 EA 9E 7C FE 6F Ø7
24F1:F7 39 43 9F D3 42 1B 5F 24
24F9:1D 68 A4 B3 Ø5 41 34 E9 3A
25Ø1:2C 38 Ø4 88 11 D2 42 E6 B7
25Ø9:3F A9 AF 4D 9A 39 3D 5C B8
2511:8C 3E 2F 78 D8 25 3B Ø5 75
2519:91 D5 91 F3 BB F2 ØB E6 B9
2521:83 C1 ØØ ØØ 44 52 55 9F 53
2529:1C 98 ØØ ØØ ØØ ØØ ØØ ØØ A7
```

Program 4

Heat Seeker

You need a copy of MLX on tape or disk before you can type in Heat Seeker. Load and run MLX, then respond to the questions with

Starting address: 0801
Ending address: 1828

After you've finished typing in the data for Heat Seeker, be sure to save a copy to tape or disk before exiting MLX.

Heat Seeker

For mistake-proof program entry, use "MLX: Machine Language Entry Program" to type in this program.

```
Ø8Ø1:ØD Ø8 Ø1 ØØ 9E 32 3Ø 36 ØE
Ø8Ø9:31 ØØ ØØ ØØ A9 ØE 8D 21 73
Ø811:DØ A9 Ø2 8D 2Ø DØ AØ 18 AA
Ø819:A9 ØØ 99 FF D3 88 DØ FA 8E
Ø821:A9 Ø2 8D 17 D4 A9 1F 8D C2
Ø829:18 D4 A9 Ø8 8D 16 D4 A9 48
Ø831:Ø3 8D Ø8 D4 A9 3D 8D ØC DD
Ø839:D4 A9 ØØ 8D ØF D4 8D ØE EB
Ø841:D4 A9 2Ø 8D 13 D4 A9 7F C1
Ø849:8D 14 D4 A9 81 8D 12 D4 95
Ø851:A9 Ø1 8D Ø3 D4 A9 19 8D 65
Ø859:Ø5 D4 A9 ØØ 8D 24 18 2Ø A3
Ø861:6B 13 2Ø E9 11 A9 3Ø AØ BE
Ø869:Ø6 99 C8 Ø7 88 DØ FA 8C 76
Ø871:2Ø 18 AC 28 18 3Ø 12 AØ F5
Ø879:Ø6 99 E1 Ø7 88 DØ FA A9 C6
Ø881:32 8D C6 Ø7 A9 31 8D DF 64
Ø889:Ø7 A9 FC 8D Ø3 18 A9 ØB D6
Ø891:A2 Ø4 9D DB 17 E8 E8 EØ 74
Ø899:1Ø DØ F7 2Ø 7F ØD A9 Ø8 72
Ø8A1:8D 21 18 8D 22 18 4C 3D E3
Ø8A9:ØB A9 ØØ 8D Ø1 18 8D Ø2 Ø8
Ø8B1:18 2Ø 61 ØD AD ØE 18 C9 72
Ø8B9:FF DØ 22 2Ø 3C ØD AD 19 CE
Ø8C1:18 C9 Ø8 9Ø Ø4 C9 F8 9Ø 24
Ø8C9:14 AD 1A 18 C9 Ø8 9Ø Ø4 A7
Ø8D1:C9 F8 9Ø Ø9 2Ø 46 ØD 2Ø FB
Ø8D9:79 ØD 4C 14 ØB AD Ø1 18 DD
Ø8E1:FØ Ø3 2Ø 83 ØA 78 A9 FD EA
Ø8E9:8D ØØ DC AD Ø1 DC 29 8Ø 85
```

```
Ø8F1:FØ F3 A9 F7 8D ØØ DC 58 AA
Ø8F9:A9 Ø4 8D 1D 18 A2 ØØ BD 6C
Ø9Ø1:FB 17 C9 7F 9Ø Ø6 AD 1D 1D
Ø9Ø9:18 2Ø 6C ØA ØE 1D 18 E8 5B
Ø911:EØ Ø6 DØ EB A5 A1 CD 1F 5C
Ø919:18 FØ Ø6 8D 1F 18 2Ø 71 18
Ø921:ØC AD ØE 18 1Ø 21 C9 CØ 41
Ø929:FØ 1D C9 FF FØ 19 2Ø F4 55
Ø931:ØB 9Ø Ø7 A9 CØ 8D ØE 18 D8
Ø939:DØ ØD A9 FF 8D ØE 18 A9 AA
Ø941:8Ø 8D 19 18 8D 1A 18 AØ 41
Ø949:Ø9 A9 FF D9 Ø4 18 FØ ØD 57
Ø951:88 DØ F8 AD ØE 18 C9 CØ FA
Ø959:DØ Ø3 4C Ø4 ØB AD 1F DØ 7C
Ø961:8D ØF 18 29 Ø1 FØ Ø9 2Ø 91
Ø969:39 ØA 2Ø 46 ØD 4C 3D ØB 22
Ø971:AD ØF 18 29 Ø2 FØ Ø3 2Ø AD
Ø979:13 ØA AD ØF 18 29 FC DØ 6E
Ø981:Ø3 4C B2 Ø8 8D ØF 18 A9 81
Ø989:Ø4 8D 1E 18 AA AD 1E 18 A6
Ø991:2D ØF 18 FØ Ø3 2Ø A5 Ø9 FC
Ø999:ØE 1E 18 E8 E8 EØ 1Ø DØ 87
Ø9A1:EC 4C B2 Ø8 2D 1Ø DØ FØ 5Ø
Ø9A9:Ø2 38 24 18 BD ØØ DØ 6A CA
Ø9B1:38 E9 Ø8 BØ Ø2 A9 ØØ C9 E6
Ø9B9:AØ 9Ø Ø2 A9 9Ø 4A 4A 4A A7
Ø9C1:4A A8 B9 Ø4 18 C9 FF DØ 53
Ø9C9:48 A9 CØ 99 Ø4 18 A9 ØØ EF
Ø9D1:9D C5 17 9D C6 17 A9 E2 A9
Ø9D9:9D Ø1 DØ BD ØØ DØ 38 E9 8E
Ø9E1:1Ø 29 EØ 18 69 1C 9D ØØ DA
Ø9E9:DØ 2Ø 8C ØC 8A 4A AA A9 3B
Ø9F1:4Ø 9D F9 17 A9 FF 9D F8 BD
Ø9F9:Ø7 8A ØA AA 2Ø 31 ØD 98 96
ØAØ1:ØA ØA A8 A9 6Ø 99 71 Ø7 9F
ØAØ9:99 72 Ø7 99 99 Ø7 99 9A B7
ØA11:Ø7 6Ø AD ØE 18 C9 FF DØ 1Ø
ØA19:1E 2Ø A7 ØC 2Ø 31 ØD A9 83
ØA21:ØØ 8D C7 17 8D C8 17 8D 4E
ØA29:ØE 18 A9 E2 8D Ø3 DØ A9 71
ØA31:Ø2 8D 28 DØ 2Ø 46 ØD 6Ø 5Ø
ØA39:A9 ØØ 8D C5 17 8D C6 17 C3
ØA41:A9 E2 8D Ø1 DØ A9 Ø2 8D 63
ØA49:27 DØ A9 Ø1 ØD 1C DØ 8D 72
ØA51:1C DØ 2Ø 79 ØD 2Ø 31 ØD 9B
ØA59:AØ CØ 84 A2 AD 1F DØ 29 5D
ØA61:Ø2 FØ Ø3 2Ø 13 ØA A4 A2 C1
ØA69:DØ F2 6Ø ØD Ø3 18 8D Ø3 16
ØA71:18 AD 1D 18 49 FF 2D 15 DB
ØA79:DØ 8D 15 DØ A9 FE 9D FA 88
```

```
ØA81:Ø7 6Ø AD Ø2 18 C9 1C BØ D7
ØA89:Ø6 A9 ØØ 8D Ø1 18 6Ø AD BA
ØA91:Ø3 18 DØ Ø1 6Ø A9 ØØ 8D 8E
ØA99:Ø1 18 8D Ø2 18 A9 Ø4 AA 2Ø
ØAA1:A8 2D Ø3 18 DØ ØA 98 ØA 21
ØAA9:A8 E8 E8 EØ 1Ø DØ F2 6Ø 81
ØAB1:8D 1D 18 ØD 15 DØ 8D 15 C3
ØAB9:DØ AD 1D 18 49 FF A8 2D 8F
ØAC1:Ø3 18 8D Ø3 18 AD 1Ø DØ A7
ØAC9:29 Ø1 FØ ØC AD 1D 18 ØD BØ
ØAD1:1Ø DØ 8D 1Ø DØ 4C EØ ØA 58
ØAD9:98 2D 1Ø DØ 8D 1Ø DØ AD 9Ø
ØAE1:C5 17 9D C5 17 AD C6 17 C2
ØAE9:9D C6 17 AD ØØ DØ 9D ØØ BA
ØAF1:DØ AD Ø1 DØ 9D Ø1 DØ 8A 24
ØAF9:4A A8 A9 ØØ 99 F9 17 58 CD
ØBØ1:4C 4C ØD AE 2Ø 18 FE 21 5D
ØBØ9:18 AØ ØA 2Ø A7 ØC 88 DØ E5
ØB11:FA FØ 29 A9 ØØ 8D C5 17 79
ØB19:8D C7 17 8D C6 17 8D C8 1A
ØB21:17 2Ø 31 ØD AD 1C DØ Ø9 4A
ØB29:Ø1 8D 1C DØ A9 Ø2 8D 27 4B
ØB31:DØ 8D 28 DØ A9 CØ 85 A2 23
ØB39:A5 A2 DØ FC 78 A9 64 8D 75
ØB41:ØØ DØ A9 64 8D Ø1 DØ A9 C2
ØB49:ØØ 8D 1Ø DØ A9 Ø1 8D 15 53
ØB51:DØ A9 FØ 8D F8 Ø7 A9 ØF 77
ØB59:8D 27 DØ A9 FE 2D 1C DØ 6A
ØB61:8D 1C DØ A9 ØØ 85 AØ 85 D6
ØB69:A1 85 A2 8D 1F 18 2Ø 52 CA
ØB71:13 A2 Ø9 A9 FF 9D Ø4 18 ØC
ØB79:CA 1Ø FA A9 ØØ 8D ØE 18 5D
ØB81:AD 1F DØ AD 1E DØ 2C 28 DF
ØB89:18 3Ø 1C AØ Ø9 B9 C5 Ø7 Ø7
ØB91:AA B9 DE Ø7 99 C5 Ø7 8A 34
ØB99:99 DE Ø7 88 DØ EF A9 Ø1 38
ØBA1:38 ED 2Ø 18 8D 2Ø 18 AE AØ
ØBA9:2Ø 18 BD 21 18 DØ 13 AØ 6A
ØBB1:ØØ 2C 28 18 3Ø Ø1 C8 B9 2A
ØBB9:21 18 DØ CA 88 1Ø F8 4C EF
ØBC1:AE ØC DE 21 18 BD 21 18 32
ØBC9:18 69 31 8D D6 Ø7 AD Ø1 74
ØBD1:DC 2D ØØ DC 29 1Ø DØ F6 91
ØBD9:AD Ø1 DC 2D ØØ DC 29 1Ø 4B
ØBE1:FØ F6 A9 ØØ 8D C6 17 8D A6
ØBE9:C7 17 A9 55 8D C5 17 58 3E
ØBF1:4C AA Ø8 AØ Ø9 B9 Ø4 18 33
ØBF9:C9 FF FØ ØD 88 1Ø F6 A9 FF
ØCØ1:FD 2D 15 DØ 8D 15 DØ 38 AD
ØCØ9:6Ø A9 CØ 99 Ø4 18 98 ØA 29
```

```
ØC11:ØA A8 A9 6Ø 99 71 Ø7 99 CD
ØC19:72 Ø7 99 99 Ø7 99 9A Ø7 D3
ØC21:98 ØA ØA ØA 48 9Ø ØA A9 2C
ØC29:Ø2 ØD 1Ø DØ 8D 1Ø DØ DØ B3
ØC31:Ø8 A9 FD 2D 1Ø DØ 8D 1Ø 39
ØC39:DØ 68 18 69 1C 8D Ø2 DØ 59
ØC41:A9 E2 8D Ø3 DØ A9 F7 8D 73
ØC49:F9 Ø7 A9 AB 8D C8 17 AD 7B
ØC51:1F DØ 2C 11 DØ 3Ø FB AD BØ
ØC59:12 DØ C9 F2 DØ F4 A9 Ø7 CB
ØC61:8D 28 DØ A9 Ø2 ØD 15 DØ 3E
ØC69:8D 15 DØ AD 1F DØ 18 6Ø 4F
ØC71:48 8A 48 A2 Ø5 FE C8 Ø7 4Ø
ØC79:A9 3A DD C8 Ø7 DØ Ø8 A9 72
ØC81:3Ø 9D C8 Ø7 CA DØ EE 68 82
ØC89:AA 68 6Ø 48 8A 48 AD CD 4Ø
ØC91:Ø7 18 69 Ø5 C9 3A BØ Ø5 4E
ØC99:8D CD Ø7 DØ EA E9 ØA 8D 7A
ØCA1:CD Ø7 A2 Ø4 DØ CF 48 8A D7
ØCA9:48 A2 Ø4 DØ C8 78 A9 31 C8
ØCB1:8D 14 Ø3 A9 EA 8D 15 Ø3 4B
ØCB9:58 A9 3Ø 8D D6 Ø7 2Ø 84 DE
ØCC1:FF AØ 1Ø B9 2Ø ØD C9 4Ø A8
ØCC9:9Ø Ø3 38 E9 4Ø 99 ØB Ø4 13
ØCD1:A9 Ø3 99 ØB D8 88 DØ EB D9
ØCD9:C8 8C 25 18 AD 25 18 DØ A2
ØCE1:ØC A9 2D 8D 19 Ø4 A9 3E 53
ØCE9:8D 1A Ø4 DØ ØA A9 3C 8D D9
ØCF1:19 Ø4 A9 2D 8D 1A Ø4 AD 2A
ØCF9:ØØ DC 2D Ø1 DC 4A 4A 4A ED
ØDØ1:BØ Ø5 AØ Ø1 8C 25 18 4A 4C
ØDØ9:BØ Ø5 AØ ØØ 8C 25 18 4A 44
ØD11:BØ CA 2Ø 81 FF AD 25 18 6B
ØD19:FØ Ø3 4C ØD Ø8 85 C6 ØØ AA
ØD21:5Ø 4C 41 59 2Ø 41 47 41 ØA
ØD29:49 4E 3F 2Ø 59 2Ø 2Ø 4E 3F
ØD31:A9 8Ø 8D ØB D4 A9 81 8D 8Ø
ØD39:ØB D4 6Ø A9 FF 38 ED Ø3 74
ØD41:DØ 8D ØF D4 6Ø A9 ØØ 8D 8D
ØD49:ØF D4 6Ø A9 18 8D 24 18 1E
ØD51:A5 A2 8D 23 18 A9 4Ø 8D 4Ø
ØD59:Ø4 D4 A9 41 8D Ø4 D4 6Ø 7A
ØD61:A5 A2 CD 23 18 FØ 1Ø 8D 15
ØD69:23 18 AD 24 18 C9 3Ø BØ ØC
ØD71:Ø7 8D Ø1 D4 EE 24 18 6Ø 78
ØD79:A9 ØØ 8D Ø1 D4 6Ø 78 A9 EC
ØD81:91 8D 14 Ø3 A9 ØD 8D 15 2C
ØD89:Ø3 A9 ØE 8D Ø5 DC 58 6Ø D6
ØD91:AD ØØ DC 2D Ø1 DC 8D 26 AD
ØD99:18 AD C5 17 ØD C6 17 DØ D7
```

```
ØDA1:Ø3 4C 39 ØE AD 26 18 29 B7
ØDA9:1Ø DØ Ø3 EE Ø1 18 AD 26 39
ØDB1:18 29 Ø3 C9 Ø3 DØ Ø3 4C CC
ØDB9:39 ØE AD D9 17 8D 14 18 76
ØDC1:AD C5 17 8D 12 18 2Ø A7 B8
ØDC9:11 AD 26 18 29 Ø2 DØ 16 27
ØDD1:38 AD D6 17 ED 1Ø 18 8D 2D
ØDD9:D6 17 AD C6 17 ED 11 18 F1
ØDE1:8D C6 17 4C FA ØD 18 AD Ø6
ØDE9:D6 17 6D 1Ø 18 8D D6 17 9F
ØDF1:AD C6 17 6D 11 18 8D C6 19
ØDF9:17 AD D9 17 8D 14 18 AD 52
ØEØ1:C6 17 8D 12 18 2Ø A7 11 BA
ØEØ9:AD 26 18 29 Ø2 DØ 16 18 B2
ØE11:AD D5 17 6D 1Ø 18 8D D5 Ø5
ØE19:17 AD C5 17 6D 11 18 8D C3
ØE21:C5 17 4C 39 ØE 38 AD D5 85
ØE29:17 ED 1Ø 18 8D D5 17 AD 6F
ØE31:C5 17 ED 11 18 8D C5 17 5E
ØE39:AD ØØ DØ 38 ED Ø2 DØ 9Ø 73
ØE41:Ø2 18 24 38 6A 8D 19 18 4Ø
ØE49:AD Ø1 DØ 38 ED Ø3 DØ 9Ø C7
ØE51:Ø2 18 24 38 6A 8D 1A 18 52
ØE59:A9 Ø3 2D 1Ø DØ FØ 2E C9 22
ØE61:Ø3 FØ 2A C9 Ø1 FØ 13 AD BC
ØE69:19 18 38 E9 8Ø C9 A5 BØ E4
ØE71:Ø3 18 69 AC 8D 19 18 4C 5A
ØE79:A6 ØE AD 19 18 18 69 8Ø 28
ØE81:C9 55 9Ø Ø3 38 E9 AC 8D 6A
ØE89:19 18 4C A6 ØE AD 19 18 9D
ØE91:3Ø Ø9 C9 55 3Ø ØC 38 E9 A2
ØE99:AC 9Ø Ø7 C9 A5 1Ø Ø3 18 39
ØEA1:69 AC 8D 19 18 AD ØE 18 8C
ØEA9:3Ø Ø3 4C FD ØF A9 ØØ 2C 53
ØEB1:C8 17 3Ø Ø9 2C C7 17 1Ø 4D
ØEB9:ØE A9 Ø1 DØ ØA 2C C7 17 1C
ØEC1:1Ø Ø3 A9 Ø2 2C A9 Ø3 8D 97
ØEC9:17 18 A9 ØØ 2C 1A 18 3Ø D6
ØED1:Ø9 2C 19 18 1Ø ØE A9 Ø1 2F
ØED9:DØ ØA 2C 19 18 1Ø Ø3 A9 A8
ØEE1:Ø2 2C A9 Ø3 8D 18 18 AD 1A
ØEE9:17 18 38 ED 18 18 4A 9Ø C3
ØEF1:Ø8 29 Ø1 8D 16 18 4C 83 82
ØEF9:ØF FØ 12 A9 ØØ 38 ED 19 8C
ØFØ1:18 8D 19 18 A9 ØØ 38 ED DE
ØFØ9:1A 18 8D 1A 18 A9 ØØ 8D 82
ØF11:1B 18 8D 1C 18 AD 19 18 F7
ØF19:ØD 1A 18 DØ Ø3 4C FD ØF A8
ØF21:AD 1B 18 18 6D 19 18 8D EE
ØF29:1B 18 AD 1C 18 18 6D 1A 68
```

```
ØF31:18 8D 1C 18 AD 1B 18 38 Ø6
ØF39:ED C7 17 4D C7 17 1Ø ØE CØ
ØF41:AD 1C 18 38 ED C8 17 4D D1
ØF49:C8 17 3Ø D4 1Ø 17 AD 1C 39
ØF51:18 38 4D C8 17 3Ø ØB AC FB
ØF59:17 18 CC 18 18 DØ Ø3 4C 7A
ØF61:FD ØF A9 ØØ 2C A9 Ø1 2C AD
ØF69:19 18 1Ø Ø2 49 Ø1 2C 1A FC
ØF71:18 1Ø Ø2 49 Ø1 AC 17 18 75
ØF79:CC 18 18 FØ Ø5 49 Ø1 8D F2
ØF81:16 18 AD DA 17 8D 14 18 43
ØF89:AD C7 17 8D 12 18 2Ø A7 Ø5
ØF91:11 AD 16 18 DØ 16 38 AD E4
ØF99:D8 17 ED 1Ø 18 8D D8 17 68
ØFA1:AD C8 17 ED 11 18 8D C8 57
ØFA9:17 4C CØ ØF 18 AD D8 17 AF
ØFB1:6D 1Ø 18 8D D8 17 AD C8 AD
ØFB9:17 6D 11 18 8D C8 17 AD CD
ØFC1:DA 17 8D 14 18 AD C8 17 26
ØFC9:8D 12 18 2Ø A7 11 AD 16 2B
ØFD1:18 DØ 16 18 AD D7 17 6D DC
ØFD9:1Ø 18 8D D7 17 AD C7 17 4B
ØFE1:6D 11 18 8D C7 17 4C FD Ø8
ØFE9:ØF 38 AD D7 17 ED 1Ø 18 79
ØFF1:8D D7 17 AD C7 17 ED 11 12
ØFF9:18 8D C7 17 A2 ØØ BD DB 5E
1ØØ1:17 8D 14 18 BD C5 17 8D D4
1ØØ9:12 18 2Ø A7 11 A9 ØØ 8D 73
1Ø11:13 18 BD B5 17 2D 1Ø DØ 32
1Ø19:FØ Ø3 EE 13 18 18 BD EB ØA
1Ø21:17 6D 1Ø 18 9D EB 17 BD 34
1Ø29:ØØ DØ 6D 11 18 9D ØØ DØ 44
1Ø31:9Ø Ø3 EE 13 18 2C 11 18 15
1Ø39:1Ø Ø3 CE 13 18 2C 13 18 DC
1Ø41:1Ø ØE A9 Ø1 8D 13 18 18 33
1Ø49:BD ØØ DØ 69 58 9D ØØ DØ Ø3
1Ø51:A9 Ø1 CD 13 18 DØ 13 BD 59
1Ø59:ØØ DØ C9 58 9Ø ØC CE 13 D1
1Ø61:18 38 BD ØØ DØ E9 58 9D CF
1Ø69:ØØ DØ AD 13 18 DØ ØC BD 7E
1Ø71:B6 17 2D 1Ø DØ 8D 1Ø DØ Ø7
1Ø79:4C 85 1Ø BD B5 17 ØD 1Ø 33
1Ø81:DØ 8D 1Ø DØ BD DB 17 8D 95
1Ø89:14 18 BD C6 17 8D 12 18 Ø9
1Ø91:2Ø A7 11 18 BD EC 17 6D 8C
1Ø99:1Ø 18 9D EC 17 BD Ø1 DØ CC
1ØA1:6D 11 18 C9 1D BØ Ø2 A9 B5
1ØA9:1D C9 FA 9Ø Ø2 A9 FA 9D 7D
1ØB1:Ø1 DØ E8 E8 EØ 1Ø FØ Ø3 5E
1ØB9:4C FF ØF 2C Ø2 18 3Ø Ø3 78
```

```
10C1:EE Ø2 18 A2 Ø5 FE FB 17 3A
1ØC9:CA 1Ø FA AD ØE 18 3Ø Ø3 C1
1ØD1:EE ØE 18 AD C7 17 DØ ØD 14
1ØD9:AD C8 17 DØ Ø8 A9 FF 8D 67
1ØE1:F9 Ø7 4C 3C 11 A9 ØØ 8D CA
1ØE9:15 18 AD C7 17 3Ø ØC C9 28
1ØF1:2Ø 9Ø 11 A9 Ø4 8D 15 18 9B
1ØF9:4C Ø5 11 C9 EØ BØ Ø5 A9 BD
11Ø1:Ø6 8D 15 18 AD C8 17 3Ø 9C
11Ø9:1Ø C9 2Ø 9Ø 19 18 A9 Ø8 37
1111:6D 15 18 8D 15 18 4C 27 D3
1119:11 C9 EØ BØ Ø9 18 A9 Ø9 62
1121:6D 15 18 8D 15 18 AD 15 94
1129:18 C9 Ø4 DØ Ø2 A9 ØA C9 EB
1131:Ø6 DØ Ø2 A9 ØB 18 69 EE DF
1139:8D F9 Ø7 AD C5 17 DØ ØD 95
1141:AD C6 17 DØ Ø8 A9 FF 8D 5Ø
1149:F8 Ø7 4C A4 11 A9 ØØ 8D 3A
1151:15 18 AD C5 17 3Ø ØC C9 71
1159:23 9Ø 11 A9 Ø4 8D 15 18 86
1161:4C 6D 11 C9 EØ BØ Ø5 A9 41
1169:Ø6 8D 15 18 AD C6 17 3Ø FC
1171:1Ø C9 2Ø 9Ø 19 18 A9 Ø8 9F
1179:6D 15 18 8D 15 18 4C 8F A4
1181:11 C9 EØ BØ Ø9 18 A9 Ø9 CA
1189:6D 15 18 8D 15 18 AD 15 FC
1191:18 C9 Ø4 DØ Ø2 A9 ØA C9 54
1199:Ø6 DØ Ø2 A9 ØB 18 69 E6 4Ø
11A1:8D F8 Ø7 4C 31 EA AD 12 11
11A9:18 8D 1Ø 18 3Ø Ø3 A9 ØØ 9F
11B1:2C A9 FF 8D 11 18 AØ Ø8 5F
11B9:88 ØE 14 18 9Ø FA CØ ØØ 99
11C1:FØ 25 ØE 1Ø 18 2E 11 18 1C
11C9:ØE 14 18 9Ø 17 18 AD 1Ø 88
11D1:18 6D 12 18 8D 1Ø 18 9Ø 8C
11D9:Ø3 EE 11 18 AD 12 18 1Ø D2
11E1:Ø3 CE 11 18 88 DØ DB 6Ø 7C
11E9:2Ø B5 FF 78 AD 16 DØ Ø9 81
11F1:1Ø 8D 16 DØ A9 1D 8D 18 44
11F9:DØ A9 Ø7 8D 23 DØ A9 ØØ 58
12Ø1:8D 25 DØ A9 Ø7 8D 26 DØ 75
12Ø9:A9 ØF 8D 27 DØ A9 Ø5 85 A6
1211:FE A9 15 85 FF A9 8Ø 85 47
1219:FC A9 3B 85 FD AØ ØØ B1 ØA
1221:FE DØ 18 E6 FE DØ Ø2 E6 9Ø
1229:FF B1 FE AA A9 ØØ 91 FC B1
1231:E6 FC DØ Ø2 E6 FD CA DØ D7
1239:F3 FØ Ø8 91 FC E6 FC DØ FB
1241:Ø2 E6 FD E6 FE DØ Ø2 E6 74
1249:FF A5 FC C9 4Ø DØ DØ A9 A3
```

```
1251:FE  8D  1C  DØ  A5  Ø1  29  FB  68
1259:85  Ø1  AØ  ØØ  B9  ØØ  DC  99  B5
1261:ØØ  3Ø  B9  ØØ  DD  99  ØØ  31  4F
1269:C8  DØ  F1  A5  Ø1  Ø9  Ø4  85  78
1271:Ø1  AØ  ØØ  B9  11  13  99  ØØ  E1
1279:32  C8  CØ  2Ø  DØ  F5  AØ  ØØ  A2
1281:98  99  ØØ  33  C8  CØ  Ø8  DØ  B5
1289:F8  A9  Ø4  85  FF  A9  D8  85  4B
1291:FD  A9  ØØ  85  FE  85  FC  A8  28
1299:A9  6Ø  91  FE  A9  Ø8  91  FC  5A
12A1:C8  DØ  F5  E6  FF  E6  FD  A5  C8
12A9:FF  C9  Ø7  DØ  EB  A9  6Ø  99  8E
12B1:ØØ  Ø7  A9  Ø8  99  ØØ  DB  C8  9A
12B9:CØ  CØ  DØ  F1  2Ø  52  13  AØ  B8
12C1:ØØ  A9  FE  99  FA  Ø7  A9  Ø2  13
12C9:99  29  DØ  C8  CØ  Ø6  DØ  F1  5D
12D1:A9  FF  8D  F8  Ø7  8D  F9  Ø7  75
12D9:AØ  ØØ  B9  31  13  C9  4Ø  9Ø  69
12E1:Ø3  38  E9  4Ø  99  CØ  Ø7  A9  5E
12E9:Ø3  99  CØ  DB  C8  2C  28  18  2B
12F1:1Ø  Ø6  CØ  18  DØ  E4  FØ  Ø4  39
12F9:CØ  21  DØ  DE  A9  Ø3  99  CØ  1C
13Ø1:DB  C8  CØ  28  DØ  F8  58  6Ø  5D
13Ø9:6Ø  42  4Ø  6Ø  6Ø  43  41  6Ø  FØ
1311:ØØ  ØØ  ØØ  ØØ  ØØ  Ø3  Ø3  Ø3  4C
1319:Ø3  Ø3  ØB  ØB  2B  2B  28  28  12
1321:ØØ  ØØ  ØØ  ØØ  ØØ  CØ  CØ  CØ  8C
1329:CØ  CØ  EØ  EØ  E8  E8  28  28  6D
1331:5Ø  4C  41  59  45  52  31  3A  6Ø
1339:2Ø  2Ø  2Ø  2Ø  2Ø  2Ø  2Ø  2Ø  5F
1341:53  48  49  5Ø  53  3A  2Ø  2Ø  35
1349:2Ø  5Ø  4C  41  59  45  52  32  E7
1351:3A  AØ  ØØ  A2  Ø4  BD  Ø8  13  21
1359:99  7Ø  Ø7  BD  ØC  13  99  98  9D
1361:Ø7  C8  CA  DØ  FØ  CØ  28  DØ  4F
1369:EA  6Ø  A9  ØØ  8D  27  18  AØ  2C
1371:6E  A2  1E  A9  14  2Ø  BB  14  82
1379:AD  27  18  ØA  AA  BD  E7  Ø4  Ø4
1381:49  8Ø  9D  E7  Ø4  2Ø  C8  14  E4
1389:FØ  1E  BD  E7  Ø4  49  8Ø  9D  C9
1391:E7  Ø4  98  18  6D  27  18  C9  43
1399:FF  DØ  Ø2  A9  ØØ  C9  Ø3  DØ  CC
13A1:Ø2  A9  Ø1  8D  27  18  DØ  DØ  38
13A9:AD  27  18  18  69  Ø4  8D  DB  47
13B1:17  69  Ø2  8D  DD  17  69  Ø3  F7
13B9:8D  DA  17  69  Ø1  8D  D9  17  DF
13C1:A9  Ø2  CD  27  18  DØ  Ø3  EE  62
13C9:D9  17  A9  FF  8D  28  18  AØ  B5
13D1:19  A2  8C  A9  14  2Ø  BB  14  Ø6
13D9:AD  28  18  ØA  AA  E8  E8  BD  ØD
```

```
13E1:2C Ø5 49 8Ø 9D 2C Ø5 2Ø 58
13E9:C8 14 FØ 23 18 6D 28 18 A8
13F1:BD 2C Ø5 49 8Ø 9D 2C Ø5 ØF
13F9:98 18 6D 28 18 C9 FE DØ 59
14Ø1:Ø2 A9 FF C9 Ø1 DØ Ø2 A9 2A
14Ø9:ØØ 8D 28 18 4C D9 13 AØ AB
1411:15 A2 A5 A9 14 2Ø BB 14 68
1419:2Ø C8 14 DØ FB 6Ø C5 2Ø 2Ø
1421:C9 2Ø CE 2Ø 3A 54 52 45 1F
1429:5Ø 58 45 2Ø 2C 45 54 41 9A
1431:49 44 45 4D 52 45 54 4E 2B
1439:49 2Ø 2C 45 43 49 56 4F 23
1441:4E 2Ø 11 ØD 2D 54 43 45 12
1449:4C 45 53 2Ø 4F 54 2Ø 4E AF
1451:4F 54 54 55 42 2Ø 44 4E 7F
1459:41 2Ø 4B 43 49 54 53 59 63
1461:4F 4A 2Ø 45 53 55 2Ø 11 5D
1469:ØD 35 38 39 31 2Ø 21 45 91
1471:54 55 5Ø 4D 4F C3 2Ø 2C ED
1479:D2 C5 CB C5 C5 D3 2Ø D4 E4
1481:C1 C5 C8 2Ø 2Ø 2Ø 2Ø 2Ø F8
1489:9B 11 93 ØE 32 2Ø 31 2Ø AB
1491:3A 53 52 45 59 41 4C 5Ø Ø3
1499:2Ø 46 4F 2Ø 52 45 42 4D C8
14A1:55 4E 2Ø 11 ØD 4E 49 47 98
14A9:45 42 2Ø 4F 54 2Ø 45 52 FD
14B1:49 46 2Ø 53 53 45 52 5Ø ED
14B9:11 ØD 86 FE 85 FF B1 FE FC
14C1:2Ø D2 FF 88 DØ F8 6Ø AD 1Ø
14C9:ØØ DC 2D Ø1 DC 29 1C C9 6C
14D1:1C DØ F4 A9 ØØ 85 A2 A9 7A
14D9:1C C5 A2 DØ FC AD ØØ DC 5E
14E1:2D Ø1 DC 29 Ø4 DØ Ø3 AØ 19
14E9:FF 6Ø AD ØØ DC 2D Ø1 DC 5A
14F1:29 Ø8 DØ Ø3 AØ Ø1 6Ø AD 72
14F9:ØØ DC 2D Ø1 DC 29 1Ø DØ 8B
15Ø1:DC AØ ØØ 6Ø ØØ Ø8 CØ ØØ 69
15Ø9:Ø1 1B CØ ØØ Ø1 3F CØ ØØ 19
1511:Ø1 3F 8Ø ØØ Ø1 3E ØØ Ø2 9E
1519:3C ØØ Ø2 3C ØØ Ø2 3C ØØ E5
1521:Ø2 3E ØØ Ø2 3F ØØ Ø2 3F 39
1529:ØØ Ø2 3F ØØ Ø2 3E ØØ Ø2 C6
1531:3E ØØ Ø2 3C ØØ Ø2 3C ØØ FE
1539:Ø2 38 ØØ Ø2 3Ø ØØ Ø9 ØC 32
1541:ØØ Ø2 1C ØØ Ø2 3C ØØ Ø2 72
1549:3C ØØ Ø2 7C ØØ Ø2 7C ØØ 9A
1551:Ø2 FC ØØ Ø2 FC ØØ Ø2 FC C4
1559:ØØ Ø2 7C ØØ Ø2 3C ØØ Ø2 96
1561:3C ØØ Ø2 3C ØØ Ø2 7C ØØ AE
1569:Ø1 Ø1 FC ØØ Ø1 Ø3 FC ØØ Ø2
```

```
1571:Ø1 Ø3 D8 ØØ Ø1 Ø3 ØØ 1B 27
1579:EØ ØØ Ø2 F8 ØØ Ø2 7E Ø3 EB
1581:EØ 3F FF FC 7F FF FF 3F F6
1589:FF FF ØØ 37 FF FF FC FF 21
1591:FF FE 3F FF FC Ø7 CØ 7E 67
1599:ØØ Ø2 1F ØØ Ø2 Ø7 ØØ 16 6A
15A1:Ø3 ØØ Ø2 Ø7 ØØ Ø2 ØF ØØ 24
15A9:Ø2 ØF ØØ Ø2 3F ØØ Ø2 3F F5
15B1:8Ø ØØ Ø1 ØF EØ ØØ Ø1 Ø3 39
15B9:F8 ØØ Ø2 FF 8Ø ØØ Ø1 3F E5
15C1:EØ ØØ Ø1 ØF EØ ØØ Ø1 Ø3 79
15C9:F8 ØØ Ø2 F8 ØØ Ø2 3C ØØ CØ
15D1:Ø2 ØC ØØ 2Ø Ø3 FC ØØ Ø1 ØF
15D9:ØF FC ØØ Ø1 1F FØ ØØ Ø1 98
15E1:1F CØ EØ 7F ØØ Ø1 F9 FC D4
15E9:ØØ Ø1 7F FØ ØØ Ø1 7F CØ 17
15F1:ØØ Ø1 3F ØØ Ø2 3C ØØ Ø2 47
15F9:1Ø ØØ 23 Ø8 ØØ Ø2 3C ØØ 91
16Ø1:Ø2 FC ØØ Ø1 Ø3 FE ØØ Ø1 92
16Ø9:ØF FE ØØ Ø1 3F 9F ØØ Ø1 Ø6
1611:FE Ø7 Ø3 F8 ØØ Ø1 ØF F8 89
1619:ØØ Ø1 3F FØ ØØ Ø1 3F CØ BF
1621:ØØ 2Ø 3Ø ØØ Ø2 3C ØØ Ø2 5E
1629:1F ØØ Ø2 1F CØ ØØ Ø1 Ø7 26
1631:FØ ØØ Ø1 Ø7 FC ØØ Ø1 Ø1 51
1639:FF ØØ Ø2 1F CØ ØØ Ø1 Ø7 A6
1641:FØ ØØ Ø1 Ø1 FC ØØ Ø2 FC FE
1649:ØØ Ø2 FØ ØØ Ø2 FØ ØØ Ø2 E9
1651:EØ ØØ Ø2 CØ ØØ 13 28 28 FE
1659:ØØ Ø1 28 28 ØØ Ø1 29 68 ØC
1661:ØØ Ø1 29 68 ØØ Ø1 Ø9 6Ø EF
1669:ØØ Ø1 Ø9 6Ø ØØ Ø1 Ø1 4Ø 43
1671:ØØ Ø1 Ø1 4Ø ØØ Ø1 Ø1 4Ø 48
1679:ØØ Ø1 Ø1 4Ø ØØ Ø1 Ø1 4Ø 5Ø
1681:ØØ 2Ø Ø1 4Ø ØØ Ø1 Ø1 4Ø 2Ø
1689:ØØ Ø1 Ø1 4Ø ØØ Ø1 Ø1 4Ø 6Ø
1691:ØØ Ø1 Ø1 4Ø ØØ Ø1 Ø9 6Ø 98
1699:ØØ Ø1 Ø9 6Ø ØØ Ø1 29 68 EB
16A1:ØØ Ø1 29 68 ØØ Ø1 28 28 36
16A9:ØØ Ø1 28 28 ØØ 23 AA ØØ 7F
16B1:Ø2 AA 8Ø ØØ Ø1 15 55 ØØ AØ
16B9:Ø1 15 55 ØØ Ø1 15 55 ØØ 5D
16C1:Ø1 AA 8Ø ØØ Ø1 AA ØØ 2E ØA
16C9:AA ØØ Ø1 Ø2 AA ØØ Ø1 55 38
16D1:54 ØØ Ø1 55 54 ØØ Ø1 55 97
16D9:54 ØØ Ø1 Ø2 AA ØØ Ø2 AA 74
16E1:ØØ 2C Ø8 ØØ Ø2 ØA ØØ Ø2 54
16E9:Ø6 8Ø ØØ Ø1 15 8Ø ØØ Ø1 F4
16F1:A5 4Ø ØØ Ø1 29 5Ø ØØ Ø1 9C
16F9:ØA 54 ØØ Ø2 15 ØØ Ø2 Ø5 12
```

```
1701:00 27 05 00 02 15 00 01 FE
1709:0A 54 00 01 29 50 00 01 EC
1711:A5 40 00 01 15 80 00 01 DD
1719:06 80 00 01 0A 00 02 08 D6
1721:00 28 20 00 02 A0 00 01 F0
1729:02 90 00 01 02 54 00 01 EE
1731:01 5A 00 01 05 68 00 01 51
1739:15 A0 00 01 54 00 02 50 21
1741:00 27 50 00 02 54 00 02 A6
1749:15 A0 00 01 05 68 00 01 05
1751:01 5A 00 01 02 54 00 01 09
1759:02 90 00 02 A0 00 02 20 F5
1761:00 1A 08 80 00 01 0A A8 DF
1769:00 01 2B E0 00 01 0B E0 46
1771:00 01 0B E8 00 01 2A A0 C8
1779:00 01 02 20 00 33 88 00 08
1781:01 02 AA 00 01 02 AE 00 73
1789:01 02 EE 80 0A FF A0 0A 3A
1791:FF E0 0B FE A8 0B FF E0 9B
1799:2A FF A8 2B FF E0 2B FF 7E
17A1:E8 0B FF E0 2F FF A0 2A F9
17A9:FF E0 2F FF F8 2B BB E0 C2
17B1:0A AA A8 FF 01 FE 02 FD AA
17B9:04 FB 08 F7 10 EF 20 DF C9
17C1:40 BF 80 7F 00 00 00 00 08
17C9:00 00 00 00 00 00 00 00 F7
17D1:00 00 00 00 00 00 00 00 FF
17D9:00 00 00 00 00 00 00 00 08
17E1:00 00 00 00 00 00 00 00 10
17E9:00 00 00 00 00 00 00 00 18
17F1:00 00 00 00 00 00 00 00 20
17F9:00 00 00 00 00 00 00 00 28
1801:00 00 00 00 00 00 00 00 31
1809:00 00 00 00 00 00 00 00 39
1811:00 00 00 00 00 00 00 00 41
1819:00 00 00 00 00 00 00 00 49
1821:00 00 00 00 00 00 00 00 51
```

Program 5
Laser Beam

Load and run your copy of MLX, and enter the following two addresses to the displayed prompts:

Starting address: C000
Ending address: CDDF

After entering the game (in one session or several), be sure to save a copy to tape or disk.

Laser Beam

For mistake-proof program entry, use "MLX: Machine Language Entry Program" to type in this program.

```
C000:A9 04 8D B5 02 20 3A C5 2F
C008:20 CC C4 A9 00 8D 15 D0 31
C010:A9 93 20 D2 FF A9 00 8D B0
C018:21 D0 8D 20 D0 A9 2E 8D 29
C020:A0 CE 85 0A 20 CA CC A2 5F
C028:09 A0 11 18 20 F0 FF A9 68
C030:DD A0 CC 20 1E AB A9 6D C4
C038:8D A0 CE 85 0A 20 CA CC 0E
C040:A2 0C A0 12 18 20 F0 FF 6E
C048:A9 E9 A0 CC 20 1E AB A2 6D
C050:0B A5 A2 C5 A2 F0 FC CA 0F
C058:D0 F7 AD B5 02 09 30 8D 73
C060:70 06 AD 00 DC 4A B0 0A CC
C068:AE B5 02 E0 09 F0 03 EE FD
C070:B5 02 4A B0 08 AE B5 02 0A
C078:F0 F8 CE B5 02 4A 4A 4A FD
C080:B0 CD AD B5 02 29 0F 8D 3F
C088:B5 02 A9 03 8D B3 02 A9 B3
C090:00 8D B9 02 8D BA 02 A9 D1
C098:0A 38 ED B5 02 0A 85 4F D8
C0A0:85 4E A9 07 8D B2 02 20 79
C0A8:E2 CA 20 19 C7 20 CC C4 01
C0B0:20 46 C3 A9 14 8D B7 02 2F
C0B8:20 7B C7 AC B3 02 C8 A9 CD
C0C0:20 99 DF 07 88 A9 28 99 F9
C0C8:DF 07 A9 07 99 DF DB 88 2E
C0D0:D0 F3 20 A9 C5 A5 A2 C5 26
C0D8:A2 F0 FC AD 1E D0 AD 1F 11
C0E0:D0 20 A9 C5 C6 41 D0 15 56
C0E8:A5 42 85 41 20 E7 C5 20 DE
C0F0:85 CC 20 45 CD A9 00 8D 63
```

```
CØF8:B4 Ø2 2Ø 48 C1 C6 4E DØ 74
C1ØØ:Ø7 A5 4F 85 4E 2Ø C2 C3 EE
C1Ø8:EA 2Ø 1F C1 2Ø 5F CB 2Ø 3F
C11Ø:37 CC AD 8D Ø2 FØ CA AD Ø8
C118:8D Ø2 DØ FB 4C E1 CØ A2 CA
C12Ø:Ø2 AD 1F DØ 4A 4A 9Ø 1A B7
C128:48 BD 26 DØ 29 ØF C9 Ø1 2B
C13Ø:FØ ØF A9 Ø1 9D 26 DØ CA 27
C138:E4 A7 DØ Ø4 A9 18 85 A9 D4
C14Ø:E8 68 E8 EØ Ø9 DØ DE 6Ø 27
C148:A5 ØA DØ Ø3 4C 16 C2 AD 59
C15Ø:B4 Ø2 DØ Ø8 2Ø EC C6 9Ø 1C
C158:Ø3 4C 1A C2 A5 53 DØ 2D 29
C16Ø:AE AØ CE BD ØØ CF 85 AA ØE
C168:85 FB BD 8Ø CF 85 AB 85 DE
C17Ø:FC BD ØØ CE 85 54 4A 9Ø 71
C178:Ø8 A9 4Ø 85 95 A9 24 DØ 37
C18Ø:Ø6 A9 42 85 95 A9 16 85 17
C188:93 85 94 85 53 A5 54 C9 C5
C19Ø:23 BØ 17 C9 21 DØ Ø4 A9 4F
C198:Ø1 DØ Ø2 A9 28 18 65 AA C2
C1AØ:85 AA A5 AB 69 ØØ 85 AB Ø3
C1A8:DØ 19 C9 24 FØ Ø4 A9 Ø1 42
C1BØ:DØ Ø2 A9 28 85 1Ø A5 AA 37
C1B8:38 E5 1Ø 85 AA A5 AB E9 59
C1CØ:ØØ 85 AB A5 95 AØ ØØ 91 36
C1C8:AA A5 AA 18 69 ØØ 85 FD 36
C1DØ:A5 AB 69 D4 85 FE A9 ØA 12
C1D8:AØ ØØ 91 FD A9 8Ø 8D ØB 34
C1EØ:D4 A9 Ø8 8D ØC D4 A5 93 A5
C1E8:18 69 ØA ØA 8D Ø8 D4 A9 94
C1FØ:81 8D ØB D4 C6 93 DØ 1E 8B
C1F8:A5 95 C9 2Ø FØ 12 A9 2Ø 33
C2ØØ:85 95 A5 FB 85 AA A5 FC 41
C2Ø8:85 AB A5 94 85 93 DØ Ø6 5B
C21Ø:A9 ØØ 85 ØA 85 53 6Ø 4C 42
C218:D4 C2 A9 ØØ 85 ØA A6 A6 36
C22Ø:BD 96 Ø3 C9 Ø2 9Ø FØ AD Ø9
C228:5Ø Ø3 FØ EB AD 3C Ø3 C9 A1
C23Ø:37 DØ E4 AD 46 Ø3 C9 D6 A5
C238:9Ø DD 8A 48 2Ø 17 C3 68 AØ
C24Ø:AA A9 1A 85 A9 86 1Ø A2 4B
C248:ØØ AØ Ø6 B9 28 DØ 29 ØF 38
C25Ø:C9 Ø1 DØ Ø1 E8 88 1Ø F3 A2
C258:BD 32 C5 18 6D B9 Ø2 8D 67
C26Ø:B9 Ø2 AD BA Ø2 69 ØØ 8D E7
C268:BA Ø2 2Ø 78 CD A6 1Ø A9 2A
C27Ø:FF 9D 46 Ø3 CE B2 Ø2 DØ 6C
C278:5Ø A9 ØØ 8D 15 DØ A9 93 3C
C28Ø:2Ø D2 FF A2 ØA AØ ØF 18 FD
```

```
C288:2Ø FØ FF 2Ø D5 C2 A9 2Ø 89
C29Ø:AØ CD 2Ø 1E AB AD B7 Ø2 45
C298:ØA 85 1Ø ØA ØA 18 65 1Ø B2
C2AØ:AA 48 A9 ØØ 2Ø CD BD 68 DE
C2A8:18 6D B9 Ø2 8D B9 Ø2 AD F1
C2BØ:BA Ø2 69 ØØ 8D BA Ø2 A9 46
C2B8:ØØ 85 A2 A5 A2 C9 78 DØ 4C
C2CØ:FA 68 68 EE B5 Ø2 4C 97 BF
C2C8:CØ 2Ø 23 C3 A9 Ø1 9D 96 7A
C2DØ:Ø3 2Ø FF C2 6Ø A9 14 8D 6B
C2D8:Ø4 D4 8D ØB D4 A9 ØA 8D E6
C2EØ:Ø5 D4 8D ØC D4 AD B5 Ø2 5B
C2E8:ØA 18 69 Ø5 8D Ø1 D4 A9 BA
C2FØ:23 8D Ø8 D4 A9 11 8D Ø4 6A
C2F8:D4 A9 15 8D ØB D4 6Ø E6 22
C3ØØ:A8 A6 A8 BD F9 C7 86 A7 1A
C3Ø8:8D 2Ø DØ AØ Ø7 B9 F9 C7 5D
C31Ø:99 27 DØ 88 1Ø F7 6Ø A2 94
C318:18 AØ Ø5 8A 2Ø A5 CC CA 19
C32Ø:DØ F7 6Ø A9 14 8D ØB D4 76
C328:AD B2 Ø2 18 69 Ø4 8D Ø1 6C
C33Ø:D4 AD B3 Ø2 ØA ØA ØA ØA BA
C338:8D Ø8 D4 A9 ØC 8D ØC D4 41
C34Ø:A9 15 8D ØB D4 6Ø EE AØ EA
C348:CE AD AØ CE C9 7A DØ 11 8E
C35Ø:A9 ØØ 8D AØ CE FØ ØA CE 85
C358:AØ CE 1Ø Ø5 A9 79 8D AØ 25
C36Ø:CE AD AØ CE A8 2Ø 6F C3 24
C368:B9 ØØ CE 2Ø 9C C3 6Ø AD ØB
C37Ø:9B C3 C9 19 BØ Ø5 EE 9B 94
C378:C3 DØ Ø5 A9 Ø5 8D 9B C3 AA
C38Ø:A9 14 8D ØB D4 A9 Ø7 8D 2D
C388:Ø1 D4 AD 9B C3 8D Ø8 D4 6E
C39Ø:A9 26 8D ØC D4 A9 15 8D ED
C398:ØB D4 6Ø Ø5 48 AD AØ CE 4Ø
C3AØ:A8 B9 ØØ CF 85 Ø4 B9 8Ø 18
C3A8:CF 85 Ø5 68 AØ ØØ 91 Ø4 CC
C3BØ:A5 Ø4 18 69 ØØ 85 Ø4 A5 69
C3B8:Ø5 69 D4 85 Ø5 A9 Ø7 91 7E
C3CØ:Ø4 6Ø A2 Ø7 BD 96 Ø3 FØ 66
C3C8:Ø3 4C 83 C4 BD 46 Ø3 C9 78
C3DØ:37 DØ Ø5 FE 46 Ø3 DØ Ø7 9F
C3D8:C9 D7 DØ 11 DE 46 Ø3 BD 3A
C3EØ:64 Ø3 49 Ø1 9D 64 Ø3 2Ø 39
C3E8:8A C4 2Ø B7 C4 BD 3C Ø3 FE
C3FØ:C9 37 DØ ØA BD 5Ø Ø3 DØ EB
C3F8:Ø5 FE 3C Ø3 DØ ØF BD 3C F4
C4ØØ:Ø3 C9 37 DØ 16 BD 5Ø Ø3 BC
C4Ø8:FØ 11 DE 3C Ø3 BD 5A Ø3 B4
C41Ø:49 Ø1 9D 5A Ø3 2Ø 8A C4 4A
```

```
C418:2Ø B7 C4 BD 5A Ø3 DØ 1D B1
C42Ø:BD 82 Ø3 18 7D 6E Ø3 9D 54
C428:82 Ø3 BD 3C Ø3 69 ØØ 9D 8A
C43Ø:3C Ø3 BD 5Ø Ø3 69 ØØ 9D BØ
C438:5Ø Ø3 4C 57 C4 BD 82 Ø3 CE
C44Ø:38 FD 6E Ø3 9D 82 Ø3 BD 1E
C448:3C Ø3 E9 ØØ 9D 3C Ø3 BD 8F
C45Ø:5Ø Ø3 E9 ØØ 9D 5Ø Ø3 BD F1
C458:64 Ø3 DØ 15 BD 8C Ø3 18 7E
C46Ø:7D 78 Ø3 9D 8C Ø3 BD 46 33
C468:Ø3 69 ØØ 9D 46 Ø3 4C 83 Ø2
C47Ø:C4 BD 8C Ø3 38 FD 78 Ø3 3B
C478:9D 8C Ø3 BD 46 Ø3 E9 ØØ 42
C48Ø:9D 46 Ø3 CA FØ Ø3 4C C4 68
C488:C3 6Ø 2Ø A7 C4 18 7D 6E 7A
C49Ø:Ø3 C9 64 9Ø Ø3 9D 6E Ø3 12
C498:2Ø A7 C4 18 7D 78 Ø3 C9 D3
C4AØ:96 9Ø Ø3 9D 78 Ø3 6Ø AD 12
C4A8:1B D4 29 1F AC 1B D4 1Ø 97
C4BØ:Ø5 49 FF 18 69 Ø1 6Ø A9 4A
C4B8:8Ø 8D Ø4 D4 A9 4Ø 8D Ø5 22
C4CØ:D4 A9 14 8D Ø1 D4 A9 81 AA
C4C8:8D Ø4 D4 6Ø 2Ø 54 CC A9 5Ø
C4DØ:ØØ 85 ØC 85 ØA 85 53 A9 4C
C4D8:Ø9 85 4Ø 8D AA Ø2 A9 8Ø 5A
C4EØ:8D ØF D4 8D 12 D4 A9 Ø8 A7
C4E8:85 41 85 42 A9 Ø3 8D B8 87
C4FØ:Ø2 A9 Ø7 8D A7 Ø2 8D A8 A8
C4F8:Ø2 AØ Ø7 2Ø 2A C5 99 6E 98
C5ØØ:Ø3 2Ø 2A C5 99 78 Ø3 AD 19
C5Ø8:1B D4 29 Ø1 99 5A Ø3 AD 75
C51Ø:1B D4 29 Ø1 99 64 Ø3 88 8Ø
C518:DØ E1 A9 Ø9 85 A4 A9 ØØ 5C
C52Ø:85 A8 2Ø FF C2 A9 Ø1 85 EØ
C528:3F 6Ø AD 1B D4 C9 96 9Ø 5E
C53Ø:F9 6Ø ØØ Ø5 ØA 14 28 5Ø 62
C538:78 AØ A2 FF A9 Ø3 85 Ø2 E2
C54Ø:A9 Ø4 85 Ø3 A9 22 2Ø 9A 33
C548:C5 E6 Ø2 A5 Ø2 C9 28 DØ 63
C55Ø:F3 C6 Ø2 A9 23 2Ø 9A C5 F6
C558:A5 Ø2 18 69 28 85 Ø2 A5 D1
C56Ø:Ø3 69 ØØ 85 Ø3 A5 Ø2 C9 9C
C568:E7 DØ E8 A9 BF 85 Ø2 A9 95
C57Ø:24 2Ø 9A C5 C6 Ø2 A5 Ø2 51
C578:C9 9A DØ F3 E6 Ø2 A9 21 9C
C58Ø:2Ø 9A C5 A5 Ø2 38 E9 28 C2
C588:85 Ø2 A5 Ø3 E9 ØØ 85 Ø3 99
C59Ø:C9 Ø3 DØ EA A9 14 8D AØ E3
C598:CE 6Ø E8 9D ØØ CE A5 Ø2 23
C5AØ:9D ØØ CF A5 Ø3 9D 8Ø CF AE
```

```
C5A8:6Ø A9 ØØ 85 1Ø A2 Ø7 AØ EØ
C5BØ:ØE BD 96 Ø3 C9 Ø2 DØ 12 BF
C5B8:AD 3C Ø3 9D 3C Ø3 AD 46 F3
C5CØ:Ø3 9D 46 Ø3 AD 5Ø Ø3 9D 8Ø
C5C8:5Ø Ø3 BD 5Ø Ø3 4A 26 1Ø 97
C5DØ:BD 3C Ø3 99 ØØ DØ BD 46 49
C5D8:Ø3 99 Ø1 DØ 88 88 CA 1Ø 85
C5EØ:DØ A5 1Ø 8D 1Ø DØ 6Ø A9 47
C5E8:ØØ 85 Ø6 AØ ØØ A5 53 FØ CE
C5FØ:Ø1 6Ø AD ØØ DC 4A BØ ØF 4B
C5F8:AE 46 Ø3 EØ 37 FØ Ø8 CE 38
C6ØØ:46 Ø3 A2 Ø1 86 Ø6 C8 4A FD
C6Ø8:BØ ØF AE 46 Ø3 EØ D7 FØ 28
C61Ø:Ø8 EE 46 Ø3 A2 Ø2 86 Ø6 86
C618:C8 4A BØ 3B 2Ø EC C6 BØ 59
C62Ø:ØF DØ 34 48 A9 2Ø 2Ø 9C 1F
C628:C3 2Ø 57 C3 68 4C 57 C6 BØ
C63Ø:AE 5Ø Ø3 DØ Ø7 AE 3C Ø3 Ø5
C638:EØ 19 FØ 1B 48 AD 3C Ø3 CØ
C64Ø:38 E9 Ø1 8D 3C Ø3 AD 5Ø F6
C648:Ø3 E9 ØØ 8D 5Ø Ø3 CØ ØØ BA
C65Ø:DØ Ø4 A9 Ø3 85 Ø6 68 4A ØC
C658:BØ 3B 2Ø EC C6 BØ ØF DØ C7
C66Ø:34 48 A9 2Ø 2Ø 9C C3 2Ø 6C
C668:46 C3 68 4C 95 C6 AE 5Ø 51
C67Ø:Ø3 FØ Ø7 AE 3C Ø3 EØ 37 6E
C678:FØ 1B 48 AD 3C Ø3 18 69 BØ
C68Ø:Ø1 8D 3C Ø3 AD 5Ø Ø3 69 C7
C688:ØØ 8D 5Ø Ø3 CØ ØØ DØ Ø4 5F
C69Ø:A9 Ø4 85 Ø6 68 4A BØ Ø4 D6
C698:A9 Ø1 85 ØA A5 Ø6 DØ ØA 7D
C6AØ:A6 3F BD 1Ø C7 8D F8 Ø7 77
C6A8:DØ 3F C6 4Ø DØ 3Ø A9 Ø9 EE
C6BØ:85 4Ø A5 ØC 49 Ø1 85 ØC EB
C6B8:A9 4Ø 8D Ø4 D4 A5 Ø6 CD 34
C6CØ:83 CC FØ Ø8 8D 83 CC A9 9F
C6C8:ØØ 8D 84 CC EE 84 CC AD E7
C6DØ:84 CC 18 69 Ø5 ØA 8D Ø1 D9
C6D8:D4 A9 41 8D Ø4 D4 A5 ØC Ø7
C6EØ:A6 Ø6 18 7D 14 C7 8D F8 F1
C6E8:Ø7 86 3F 6Ø AE 3C Ø3 EØ D6
C6FØ:19 DØ 1C AE 46 Ø3 EØ 37 E4
C6F8:DØ 15 AE 5Ø Ø3 DØ 1Ø CE 59
C7ØØ:A7 Ø2 DØ Ø8 AE A8 Ø2 8E 29
C7Ø8:A7 Ø2 A2 ØØ 18 9Ø Ø1 38 7D
C71Ø:6Ø CC CF D3 D5 CD DØ D2 94
C718:D4 A9 93 2Ø D2 FF A9 ØØ DA
C72Ø:8D 21 DØ A9 ØB 8D 2Ø DØ 13
C728:A9 18 85 1Ø A9 69 AØ C7 4Ø
C73Ø:2Ø 1E AB C6 1Ø DØ F5 AØ 89
```

```
C738:25 98 99 BF Ø7 A9 Ø1 99 2A
C74Ø:BF DB C8 CØ 28 DØ F2 A2 D8
C748:13 A9 AØ 9D CØ Ø7 A9 ØA 39
C75Ø:9D CØ DB CA 1Ø F3 A2 18 B4
C758:AØ 14 18 2Ø FØ FF A9 7Ø 8D
C76Ø:AØ C7 2Ø 1E AB 2Ø 78 CD B4
C768:6Ø 1E 12 2Ø 2Ø 2Ø ØD ØØ 8F
C77Ø:9F 12 53 43 4F 52 45 92 D3
C778:2Ø 3Ø ØØ AØ ØØ B9 22 C8 22
C78Ø:99 ØØ 33 B9 22 C9 99 ØØ 4A
C788:34 B9 22 CA 99 ØØ 35 88 51
C79Ø:DØ EB A9 FF 8D 15 DØ A9 C4
C798:Ø7 A8 38 ED B2 Ø2 85 1Ø 74
C7AØ:B9 F9 C7 99 27 DØ B9 Ø1 ØF
C7A8:C8 99 F8 Ø7 AD Ø1 C8 8D 23
C7BØ:F8 Ø7 B9 Ø9 C8 99 3C Ø3 6E
C7B8:AD Ø9 C8 8D 3C Ø3 B9 11 C5
C7CØ:C8 99 46 Ø3 AD 11 C8 8D E4
C7C8:46 Ø3 B9 19 C8 99 5Ø Ø3 55
C7DØ:A9 ØØ 8D 5Ø Ø3 A9 ØØ 99 44
C7D8:96 Ø3 8D 96 Ø3 88 C4 1Ø 63
C7EØ:DØ BE A9 FF 8D 1C DØ A9 E5
C7E8:Ø7 8D 25 DØ A9 Ø6 8D 26 B7
C7FØ:DØ A9 1A 85 A9 2Ø 17 C3 AE
C7F8:6Ø Ø2 Ø2 Ø3 Ø4 ØD Ø6 Ø7 11
C8ØØ:ØB CC D6 D6 D6 D6 D6 D6 29
C8Ø8:D6 19 64 64 64 64 64 64 FF
C81Ø:64 64 64 64 64 64 64 64 A1
C818:64 ØØ ØØ ØØ ØØ ØØ ØØ ØØ DB
C82Ø:ØØ 6Ø ØØ ØØ ØØ ØØ ØØ ØØ C9
C828:ØØ 3C ØØ ØØ FF ØØ ØØ FF C8
C83Ø:ØØ ØØ FF ØØ ØØ 3C ØØ Ø2 B4
C838:FF 8Ø Ø2 FF 8Ø Ø2 FF 8Ø B6
C84Ø:ØØ FF 8Ø ØØ 55 8Ø ØØ FF 8E
C848:ØØ ØØ FF ØØ ØØ FF ØØ ØØ D9
C85Ø:FF ØØ ØØ FF ØØ ØØ 64 ØØ AA
C858:ØØ AA ØØ ØØ AA ØØ ØØ AA 94
C86Ø:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ F1
C868:ØØ 3C ØØ ØØ FF ØØ ØØ FF Ø9
C87Ø:ØØ ØØ FF ØØ ØØ 3C ØØ Ø2 F4
C878:FF 8Ø Ø2 FF 8Ø Ø2 FF 8Ø F6
C88Ø:ØØ FF 8Ø ØØ 55 8Ø ØØ FF CE
C888:ØØ ØØ FF ØØ ØØ FF ØØ ØØ 1A
C89Ø:AF ØØ ØØ AF ØØ ØØ ØA ØØ Ø9
C898:ØØ ØA ØØ ØØ ØA ØØ ØØ ØA Ø7
C8AØ:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 32
C8A8:ØØ 3C ØØ ØØ FF ØØ ØØ FF 49
C8BØ:ØØ ØØ FF ØØ ØØ 3C ØØ Ø2 35
C8B8:FF 8Ø Ø2 FF 8Ø Ø2 FF 8Ø 37
C8CØ:Ø2 FF ØØ Ø2 55 ØØ ØØ FF 1E
```

```
C8C8:ØØ ØØ FF ØØ ØØ FF ØØ ØØ 5A
C8DØ:FA ØØ ØØ FA ØØ ØØ AØ ØØ DØ
C8D8:ØØ AØ ØØ ØØ AØ ØØ ØØ AØ 38
C8EØ:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 72
C8E8:ØØ 3C ØØ ØØ FF ØØ ØØ EB 75
C8FØ:ØØ ØØ FF ØØ ØØ 3C ØØ Ø2 75
C8F8:FF 8Ø Ø2 FF 8Ø Ø1 FF 8Ø 73
C9ØØ:Ø1 FF 8Ø ØØ 55 8Ø ØØ FF DØ
C9Ø8:8Ø ØØ FF ØØ ØØ FF ØØ ØØ DB
C91Ø:FF ØØ ØØ FF ØØ ØØ AA ØØ F8
C918:ØØ AA ØØ ØØ AA ØØ ØØ AA 56
C92Ø:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ B3
C928:ØØ 3C ØØ ØØ FF ØØ ØØ EB B6
C93Ø:ØØ ØØ FF ØØ ØØ 3C ØØ Ø2 B6
C938:FF 8Ø Ø2 FF 8Ø Ø1 FF 8Ø B4
C94Ø:Ø1 FF 8Ø ØØ 55 8Ø ØØ FF 11
C948:ØØ ØØ FF ØØ ØØ FF ØØ ØØ DB
C95Ø:FF ØØ ØØ AF ØØ ØØ ØA ØØ F2
C958:ØØ ØA ØØ ØØ ØA ØØ ØØ ØA C8
C96Ø:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ F3
C968:ØØ 3C ØØ ØØ FF ØØ ØØ EB F6
C97Ø:ØØ ØØ FF ØØ ØØ 3C ØØ Ø2 F6
C978:FF 8Ø Ø2 FF 8Ø Ø2 FF 8Ø F8
C98Ø:Ø2 FF 8Ø Ø1 55 ØØ Ø1 FF E1
C988:ØØ ØØ FF ØØ ØØ FF ØØ ØØ 1C
C99Ø:FF ØØ ØØ FA ØØ ØØ AØ ØØ 15
C998:ØØ AØ ØØ ØØ AØ ØØ ØØ AØ F9
C9AØ:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 34
C9A8:ØØ FØ ØØ ØØ FC ØØ ØØ BC 1D
C9BØ:ØØ ØØ FC ØØ ØØ 3C ØØ ØØ D4
C9B8:FB ØØ 14 FA ØØ Ø6 FE 8Ø 13
C9CØ:ØØ FF 8Ø ØØ 56 ØØ ØØ F8 1Ø
C9C8:ØØ ØØ FF ØØ Ø3 CF ØØ ØF C2
C9DØ:ØF ØØ ØC ØF ØØ Ø8 Ø3 A8 2D
C9D8:Ø8 ØØ Ø8 Ø8 ØØ Ø8 28 ØØ 62
C9EØ:ØØ FF ØØ ØØ ØØ ØØ ØØ ØØ 74
C9E8:ØØ FØ ØØ ØØ FC ØØ ØØ BC 5D
C9FØ:ØØ ØØ FC ØØ ØØ 3C ØØ ØØ 15
C9F8:FB ØØ ØØ FB ØØ Ø2 AB ØØ A9
CAØØ:ØØ FF ØØ ØØ 54 ØØ ØØ FC 35
CAØ8:ØØ ØØ FC ØØ ØØ 3C ØØ ØØ 2E
CA1Ø:3C ØØ ØØ 3C ØØ ØØ 28 ØØ D7
CA18:ØØ 28 ØØ ØØ 28 ØØ ØØ A8 A1
CA2Ø:ØØ FF ØØ ØØ ØØ ØØ ØØ ØØ B5
CA28:ØØ ØF ØØ ØØ 3F ØØ ØØ 3E B9
CA3Ø:ØØ ØØ 3F ØØ ØØ 3C ØØ ØØ 9E
CA38:EF ØØ ØØ AF 2Ø Ø2 BF 8Ø C9
CA4Ø:Ø2 FF ØØ ØØ 95 ØØ ØØ 27 AA
CA48:ØØ ØØ F5 ØØ ØØ F3 CØ ØØ ED
CA5Ø:FØ FØ ØØ FØ 3Ø 2A CØ 2Ø 75
```

```
CA58:2Ø ØØ 2Ø 2Ø ØØ 2Ø ØØ ØØ 84
CA6Ø:28 FF ØØ ØØ ØØ ØØ ØØ ØØ ØA
CA68:ØØ ØF ØØ ØØ 3F ØØ ØØ 3E F9
CA7Ø:ØØ ØØ 3F ØØ ØØ 3C ØØ ØØ DE
CA78:EF ØØ ØØ EF 14 ØØ EA 9Ø ØC
CA8Ø:ØØ FF ØØ ØØ 15 ØØ ØØ 3F FD
CA88:ØØ ØØ 3F ØØ ØØ 3C ØØ ØØ F6
CA9Ø:3C ØØ ØØ 3C ØØ ØØ 28 ØØ 58
CA98:ØØ 28 ØØ ØØ 28 ØØ ØØ 2A A3
CAAØ:ØØ FF ØØ ØØ ØØ ØØ ØØ ØØ 36
CAA8:ØØ ØØ ØØ ØØ AA ØØ Ø2 AA 42
CABØ:8Ø ØA 5A AØ Ø9 5A AØ 29 7A
CAB8:6A A8 29 AA A8 2A AA A8 69
CACØ:2A AA A8 2A AA A8 2A AA C4
CAC8:A8 2A AA A8 ØA AA AØ ØA 63
CADØ:AA AØ Ø2 AA 8Ø ØØ AA ØØ 28
CAD8:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 6E
CAEØ:ØØ ØØ 78 A9 33 85 Ø1 A9 7B
CAE8:ØØ 85 Ø2 A9 38 85 Ø3 A9 42
CAFØ:ØØ 85 Ø4 A9 DØ 85 Ø5 A2 4C
CAF8:Ø8 AØ ØØ B1 Ø4 91 Ø2 88 C8
CBØØ:DØ F9 E6 Ø5 E6 Ø3 CA DØ 55
CBØ8:F2 A9 37 85 Ø1 A9 1E 8D 3B
CB1Ø:18 DØ AØ 3F B9 1F CB 99 6B
CB18:Ø8 39 88 1Ø F7 58 6Ø 8Ø 76
CB2Ø:CØ D8 E7 E7 D8 CØ 8Ø FF 94
CB28:7E 18 24 24 18 18 18 Ø1 1E
CB3Ø:Ø3 1B E7 E7 1B Ø3 Ø1 18 8A
CB38:18 18 24 24 18 7E FF 7F E2
CB4Ø:3F 2A 1F 15 ØF ØA Ø7 FF E5
CB48:FF A5 FF 52 FF A5 FF F8 FD
CB5Ø:FØ 5Ø EØ AØ CØ 4Ø 8Ø 1C BE
CB58:14 1D 3F 5C 3C 14 36 AD 3B
CB6Ø:1E DØ 29 Ø1 FØ 65 A2 Ø7 D9
CB68:A9 FF 85 1Ø A9 Ø9 85 A4 A7
CB7Ø:BD 96 Ø3 DØ 39 AD 5Ø Ø3 1E
CB78:DD 5Ø Ø3 DØ 31 AD 3C Ø3 3C
CB8Ø:38 FD 3C Ø3 1Ø Ø5 49 FF 92
CB88:18 69 Ø1 C9 ØF BØ 1F 85 42
CB9Ø:A5 AD 46 Ø3 38 FD 46 Ø3 A8
CB98:1Ø Ø5 49 FF 18 69 Ø1 C9 D4
CBAØ:ØF BØ ØB 18 65 A5 C5 1Ø 2C
CBA8:BØ Ø4 85 1Ø 86 A4 CA DØ 78
CBBØ:BF A6 A4 EØ Ø9 FØ 14 A9 52
CBB8:Ø2 9D 96 Ø3 86 A6 E4 A7 FB
CBCØ:DØ ØA BD 27 DØ 29 ØF C9 8Ø
CBC8:Ø1 DØ Ø1 6Ø A9 Ø5 85 1Ø B7
CBDØ:A9 14 8D ØB D4 A9 ØØ 8D 7F
CBD8:Ø8 D4 A9 ØA 8D ØC D4 A9 6F
CBEØ:15 8D ØB D4 A2 ØØ AC 1B 9E
```

```
CBE8:D4 A9 ØØ 99 ØØ 33 99 ØØ EE
CBFØ:34 8E Ø8 D4 99 8Ø 34 2Ø EB
CBF8:31 CC CA DØ E9 C6 1Ø DØ 1E
CCØØ:E5 68 68 2Ø CA CC CE B3 9Ø
CCØ8:Ø2 FØ Ø3 4C B8 CØ A9 ØØ 2Ø
CC1Ø:8D 15 DØ A2 ØB AØ ØF 18 ØB
CC18:2Ø FØ FF A9 2Ø 8D EØ Ø7 98
CC2Ø:A9 ØF AØ CD 2Ø 1E AB AD C1
CC28:ØØ DC 29 1Ø DØ F9 4C Ø5 2B
CC3Ø:CØ AØ 64 88 DØ FD 6Ø A5 4C
CC38:A9 ØA ØA ØA 18 69 17 CD 6D
CC4Ø:46 Ø3 BØ ØF AD 5Ø Ø3 DØ 4A
CC48:ØA AD 3C Ø3 C9 2A BØ Ø3 65
CC5Ø:4C CC CB 6Ø A2 ØØ 8A 9D 8A
CC58:ØØ D4 E8 EØ 1A DØ F8 A9 Ø2
CC6Ø:14 8D Ø1 D4 A9 42 8D Ø5 4B
CC68:D4 A9 Ø8 8D Ø3 D4 A9 Ø5 74
CC7Ø:8D ØC D4 A9 Ø5 8D ØA D4 5Ø
CC78:A9 FF 8D ØF D4 A9 ØF 8D 82
CC8Ø:18 D4 6Ø ØØ ØØ CE AA Ø2 F9
CC88:DØ 1A A9 32 8D AA Ø2 A5 2A
CC9Ø:A9 C9 1A FØ ØF C6 A9 1Ø BA
CC98:Ø6 A9 ØØ 85 A9 FØ Ø5 AØ B3
CCAØ:Ø7 2Ø A5 CC 6Ø 84 1Ø A8 25
CCA8:A9 D8 85 Ø2 A9 D7 85 Ø3 D8
CCBØ:A5 Ø2 18 69 28 85 Ø2 A5 38
CCB8:Ø3 69 ØØ 85 Ø3 88 DØ FØ 53
CCCØ:A5 1Ø AØ Ø2 91 Ø2 88 1Ø 1B
CCC8:FB 6Ø A9 Ø1 8D B4 Ø2 2Ø 21
CCDØ:48 C1 A5 A2 C5 A2 FØ FC 75
CCD8:A5 ØA DØ EE 6Ø 9E 92 4C BF
CCEØ:2Ø 41 2Ø 53 2Ø 45 2Ø 52 BC
CCE8:ØØ 42 2Ø 45 2Ø 41 2Ø 4D FE
CCFØ:96 11 11 11 9D 9D 9D 9D 89
CCF8:9D 9D 9D 4C 45 56 45 4C 9B
CDØØ:2Ø 2Ø 11 11 11 9D 9D 9D BE
CDØ8:9D 9D 9D 9D 9D 9D 9D 12 18
CD1Ø:9F 48 49 54 2Ø 46 49 52 FA
CD18:45 42 55 54 54 4F 4E ØØ 53
CD2Ø:9E 4C 45 56 45 4C 2Ø 43 ØB
CD28:4F 4D 5Ø 4C 45 54 45 44 D7
CD3Ø:11 11 9D 9D 9D 9D 9D 9D 62
CD38:9D 9D 9D 9D 9D 9D 42 4F CE
CD4Ø:4E 55 53 2Ø ØØ AD B7 Ø2 EC
CD48:FØ 2D CE B6 Ø2 DØ 28 CE 5F
CD5Ø:B8 Ø2 DØ 23 A2 Ø3 8E B8 ØC
CD58:Ø2 CE B7 Ø2 AD B7 Ø2 DØ EØ
CD6Ø:1Ø A9 18 85 A9 A9 Ø1 A2 62
CD68:Ø6 9D 28 DØ CA 1Ø FA A9 B6
CD7Ø:ØØ AA A9 2Ø 9D CØ Ø7 6Ø 4C
```

```
CD78:A2 18 AØ 1A 18 2Ø FØ FF 44
CD8Ø:AD BA Ø2 C9 FF FØ Ø7 AE FE
CD88:B9 Ø2 2Ø CD BD 6Ø A9 98 BD
CD9Ø:AØ CD 2Ø 1E AB 68 68 6Ø Ø6
CD98:93 11 11 11 2Ø 43 4F 4E 7Ø
CDAØ:47 52 41 54 55 4C 41 54 94
CDA8:49 4F 4E 53 2C 2Ø 59 4F 9F
CDBØ:55 2Ø 41 52 45 2Ø 4F 4E E3
CDB8:45 2Ø 4F 46 2Ø 54 48 45 75
CDCØ:2Ø 46 45 57 2Ø 54 4F 2Ø 2D
CDC8:57 49 4E 2Ø 41 54 2Ø 4C 16
CDDØ:41 53 45 52 2Ø 42 45 41 85
CDD8:4D 2E ØØ ØØ ØØ ØØ ØØ ØØ A6
```

Program 6
Whirlybird

Load and run MLX. When the prompts appear, answer them with

Starting address: 0801
Ending address: 1458

Save a copy of the game to tape or disk once you're finished typing in all the data.

Whirlybird

For mistake-proof program entry, use "MLX: Machine Language Entry Program" to type in this program.

```
Ø8Ø1:1B Ø8 ØØ ØØ 9E 32 3Ø 38 F6
Ø8Ø9:3Ø 3A 8F 22 14 14 14 14 Ø1
Ø811:14 2Ø 57 48 49 52 4C 59 28
Ø819:42 49 52 44 ØØ ØØ ØØ A9 D4
Ø821:93 2Ø D2 FF A2 ØØ 8E 21 BØ
Ø829:DØ 8E 2Ø DØ E8 8E 1B DØ DE
Ø831:A9 Ø3 8D 86 Ø2 A2 1Ø A9 55
Ø839:ØØ 9D ØØ DØ CA 1Ø FA A2 EC
Ø841:18 9D ØØ D4 CA 1Ø FA A2 41
Ø849:18 BD 3B 14 9D ØØ D4 CA DE
Ø851:1Ø F7 A9 7F 8D ØD DC A9 98
Ø859:33 85 Ø1 A9 ØØ 85 FB A9 D6
Ø861:DØ 85 FC A9 ØØ 85 FD A9 31
Ø869:38 85 FE AØ ØØ B1 FB 91 31
Ø871:FD E6 FB DØ Ø2 E6 FC E6 53
Ø879:FD DØ Ø2 E6 FE 38 A5 FC 8C
Ø881:C9 D8 9Ø E9 A9 37 85 Ø1 93
Ø889:A9 81 8D ØD DC A2 Ø8 BD 9Ø
Ø891:DØ 13 9D DØ 3A BD D8 13 1D
Ø899:9D 5Ø 3B BD EØ 13 9D 5Ø AE
Ø8A1:39 CA 1Ø EB A9 1F 8D 18 BE
Ø8A9:DØ A9 11 85 FB A9 11 85 35
Ø8B1:FC A9 ØØ 85 FD A9 3C 85 97
Ø8B9:FE AØ ØØ B1 FB 91 FD E6 95
Ø8C1:FB DØ Ø2 E6 FC E6 FD DØ Ø3
Ø8C9:Ø2 E6 FE 38 A5 FB E9 DØ B9
Ø8D1:A5 FC E9 13 9Ø E5 78 A9 19
Ø8D9:DB 8D 14 Ø3 A9 ØF 8D 15 A7
Ø8E1:Ø3 58 A9 ØØ 8D 69 Ø3 A9 8Ø
Ø8E9:FØ 8D 3C Ø3 A9 ØØ 8D 57 4D
Ø8F1:Ø3 8D 58 Ø3 2Ø 3C ØA A9 D1
```

```
Ø8F9:14 8D ØB D4 A9 15 8D ØB ED
Ø9Ø1:D4 2Ø C6 ØE AD 62 Ø3 FØ 2D
Ø9Ø9:F8 CE 62 Ø3 A9 Ø1 8D 51 85
Ø911:Ø3 A9 ØØ 8D 5F Ø3 AD 57 A1
Ø919:Ø3 ØD 58 Ø3 E9 ØA 9Ø 14 D7
Ø921:38 AD 57 Ø3 E9 ØA 8D 57 BF
Ø929:Ø3 AD 58 Ø3 E9 ØØ 8D 58 26
Ø931:Ø3 2Ø 2E ØE 2Ø CA 1Ø AD 6D
Ø939:ØØ DC 29 1Ø DØ F6 2Ø F8 44
Ø941:ØB 2Ø E7 ØC 2Ø 9F FF 2Ø 3E
Ø949:E4 FF C9 85 FØ Ø6 C9 88 1B
Ø951:FØ ØA DØ ØD A9 Ø1 8D 69 1F
Ø959:Ø3 4C 45 Ø9 A9 ØØ 8D 69 ØB
Ø961:Ø3 AD 69 Ø3 DØ DE AD 46 61
Ø969:Ø3 DØ Ø8 A9 ØØ 8D 68 Ø3 D6
Ø971:4C 7E Ø9 AD 68 Ø3 DØ Ø5 3B
Ø979:A9 Ø1 8D 68 Ø3 2Ø 97 ØE AE
Ø981:AD 5B Ø3 DØ 12 2Ø Ø2 ØE D1
Ø989:2Ø Ø2 ØE 2Ø Ø2 ØE 2Ø Ø2 7A
Ø991:ØE 2Ø Ø2 ØE 4C Ø2 Ø9 AD FD
Ø999:55 Ø3 FØ 32 2Ø 39 ØD 2Ø 78
Ø9A1:ED ØB AD ØØ DC 29 ØF C9 96
Ø9A9:ØB FØ Ø7 C9 Ø7 FØ ØB 4C 59
Ø9B1:CØ Ø9 A9 ØØ 8D 53 Ø3 4C A7
Ø9B9:CØ Ø9 A9 Ø1 8D 53 Ø3 2Ø 93
Ø9C1:Ø2 ØE 2Ø 51 ØA A2 Ø1 8E DC
Ø9C9:51 Ø3 8A 4C D1 Ø9 A9 ØØ 61
Ø9D1:8D 56 Ø3 2Ø 51 ØA 2Ø 38 CD
Ø9D9:ØC AD 63 Ø3 FØ Ø3 4C Ø5 2B
Ø9E1:Ø9 AD ØØ DC 29 ØF C9 ØB D5
Ø9E9:C9 ØB FØ 24 C9 Ø7 FØ 18 68
Ø9F1:AD 46 Ø3 DØ Ø3 4C 45 Ø9 B6
Ø9F9:CE 64 Ø3 AD 64 Ø3 DØ 2D C5
ØAØ1:A9 Ø3 8D 64 Ø3 4C 2B ØA 4C
ØAØ9:AD 41 Ø3 DØ Ø8 4C 26 ØA 79
ØA11:AD 41 Ø3 DØ 1Ø AD 46 Ø3 8Ø
ØA19:C9 ØC 9Ø Ø3 4C 45 Ø9 EE CF
ØA21:46 Ø3 4C 45 Ø9 AD 46 Ø3 85
ØA29:FØ Ø6 CE 46 Ø3 4C 45 Ø9 52
ØA31:AD 41 Ø3 49 Ø1 8D 41 Ø3 25
ØA39:4C 45 Ø9 A9 9B 8D Ø7 D4 76
ØA41:A9 Ø3 8D Ø8 D4 A9 ØB 8D ØE
ØA49:ØØ D4 A9 Ø7 8D Ø1 D4 6Ø B2
ØA51:EE 47 Ø3 AD 47 Ø3 DØ F8 CA
ØA59:EE 48 Ø3 AD 48 Ø3 C9 ØF 23
ØA61:9Ø EE A9 ØØ 8D 48 Ø3 6Ø A2
ØA69:A9 CØ 8D 49 Ø3 A9 Ø7 8D 23
ØA71:4A Ø3 A9 5Ø 85 Ø3 A9 Ø4 35
ØA79:85 Ø4 A9 6A 8D 4B Ø3 A9 76
ØA81:ØD 8D 4C Ø3 2Ø 9Ø ØB A9 3C
```

```
ØA89:E7 8D 49 Ø3 A9 Ø7 8D 4A 1D
ØA91:Ø3 A9 77 85 Ø3 A9 Ø4 85 25
ØA99:Ø4 A9 6A 8D 4B Ø3 2Ø 9Ø 77
ØAA1:ØB A9 F9 85 Ø3 A9 Ø6 85 8D
ØAA9:Ø4 A9 1E 8D 49 Ø3 A9 Ø7 77
ØAB1:8D 4A Ø3 A9 5A 8D 4B Ø3 BC
ØAB9:A9 Ø7 8D 4C Ø3 2Ø C2 ØB Ø4
ØAC1:A9 21 85 Ø3 A9 Ø7 85 Ø4 4C
ØAC9:A9 46 8D 49 Ø3 A9 Ø7 8D E4
ØAD1:4A Ø3 A9 Ø6 8D 4C Ø3 2Ø 25
ØAD9:C2 ØB A9 49 85 Ø3 A9 Ø7 6E
ØAE1:85 Ø4 A9 6E 8D 49 Ø3 A9 17
ØAE9:Ø7 8D 4A Ø3 A9 Ø5 8D 4C 27
ØAF1:Ø3 2Ø C2 ØB A9 71 85 Ø3 B9
ØAF9:A9 Ø7 85 Ø4 A9 96 8D 49 A1
ØBØ1:Ø3 A9 Ø7 8D 4A Ø3 A9 Ø7 75
ØBØ9:8D 4C Ø3 2Ø C2 ØB A9 99 8A
ØB11:85 Ø3 A9 Ø7 85 Ø4 A9 BE 9E
ØB19:8D 49 Ø3 A9 Ø7 8D 4A Ø3 49
ØB21:A9 Ø6 8D 4C Ø3 2Ø C2 ØB 2D
ØB29:A9 C1 85 Ø3 A9 Ø7 85 Ø4 DD
ØB31:A9 E6 8D 49 Ø3 A9 Ø7 8D 76
ØB39:4A Ø3 A9 Ø3 8D 4C Ø3 2Ø 5E
ØB41:C2 ØB A9 57 8D Ø2 Ø7 8D 36
ØB49:15 Ø7 8D 77 Ø7 8D 9Ø Ø7 6B
ØB51:8D C4 Ø7 8D CA Ø7 8D DØ 77
ØB59:Ø7 8D 58 Ø7 8D 6Ø Ø7 8D 5B
ØB61:D7 Ø7 8D DD Ø7 8D E3 Ø7 F1
ØB69:A9 Ø1 8D Ø2 DB 8D 15 DB 81
ØB71:8D 77 DB 8D 9Ø DB 8D C4 54
ØB79:DB 8D CA DB 8D DØ DB 8D EC
ØB81:D7 DB 8D 58 DB 8D 6Ø DB 63
ØB89:8D DD DB 8D E3 DB 6Ø AØ 22
ØB91:ØØ AD 4B Ø3 91 Ø3 A5 Ø3 93
ØB99:85 Ø5 18 A5 Ø4 69 D4 85 Ø6
ØBA1:Ø6 AD 4C Ø3 91 Ø5 18 A5 56
ØBA9:Ø3 69 28 85 Ø3 A5 Ø4 69 19
ØBB1:ØØ 85 Ø4 38 AD 49 Ø3 E5 AB
ØBB9:Ø3 AD 4A Ø3 E5 Ø4 BØ D1 A8
ØBC1:6Ø AØ ØØ AD 4B Ø3 91 Ø3 97
ØBC9:A5 Ø3 85 Ø5 18 A5 Ø4 69 3D
ØBD1:D4 85 Ø6 AD 4C Ø3 91 Ø5 E5
ØBD9:E6 Ø3 DØ Ø2 E6 Ø4 38 AD C3
ØBE1:49 Ø3 E5 Ø3 AD 4A Ø3 E5 CC
ØBE9:Ø4 BØ D8 6Ø A9 2Ø AØ ØØ 5E
ØBF1:91 FB A9 Ø1 91 FD 6Ø 2Ø 7A
ØBF9:34 ØC C9 23 BØ F9 8D 4F 7Ø
ØCØ1:Ø3 18 A9 CA 6D 4F Ø3 85 B6
ØCØ9:FB A9 Ø4 69 ØØ 85 FC 18 C8
ØC11:A5 FB 69 ØØ 85 FD A5 FC 94
```

```
ØC19:69 D4 85 FE A9 Ø1 8D 51 79
ØC21:Ø3 2Ø 34 ØC 3Ø Ø7 A9 ØØ FA
ØC29:8D 53 Ø3 FØ Ø5 A9 Ø1 8D AA
ØC31:53 Ø3 6Ø AD 1B D4 6Ø A9 31
ØC39:ØØ 8D 63 Ø3 2Ø ED ØB AD CD
ØC41:53 Ø3 DØ 16 2Ø 71 ØC AØ BE
ØC49:ØØ B1 FB C9 6A FØ Ø2 DØ D5
ØC51:3B 2Ø 7F ØC 2Ø 2Ø ØD 4C A7
ØC59:38 ØC 2Ø 7F ØC AØ ØØ B1 21
ØC61:FB C9 6A FØ Ø2 DØ 25 2Ø Ø4
ØC69:71 ØC 2Ø 2Ø ØD 4C 38 ØC 59
ØC71:38 A5 FB E9 Ø1 85 FB A5 E8
ØC79:FC E9 ØØ 85 FC 6Ø 18 A5 22
ØC81:FB 69 Ø1 85 FB A5 FC 69 44
ØC89:ØØ 85 FC 6Ø A5 FB 85 AA 7B
ØC91:A5 FC 85 AB AD 51 Ø3 DØ BØ
ØC99:1B 38 A5 FB E9 51 A5 FC 9E
ØCA1:E9 Ø4 9Ø 5A 38 A5 FB E9 A1
ØCA9:28 85 FB A5 FC E9 ØØ 85 26
ØCB1:FC 4C CD ØC 38 A5 FB E9 1Ø
ØCB9:CØ A5 FC E9 Ø7 BØ 4D 18 87
ØCC1:A5 FB 69 28 85 FB A5 FC BF
ØCC9:69 ØØ 85 FC B1 FB C9 57 7F
ØCD1:DØ Ø3 4C 54 ØD C9 5A FØ 17
ØCD9:Ø7 C9 6A FØ 42 4C E7 ØC 63
ØCE1:2Ø 15 ØD 2Ø 83 ØE AØ ØØ 88
ØCE9:A9 2A 91 FB 18 A5 FB 69 ØC
ØCF1:ØØ 85 FD A5 FC 69 D4 85 42
ØCF9:FE A9 Ø1 91 FD 6Ø A9 Ø1 FA
ØDØ1:8D 51 Ø3 8D 63 Ø3 A9 ØØ E9
ØDØ9:8D 46 Ø3 6Ø A9 ØØ 8D 51 9B
ØD11:Ø3 4C 39 ØD AD 51 Ø3 49 B9
ØD19:Ø1 8D 51 Ø3 4C 39 ØD AD 8Ø
ØD21:53 Ø3 49 Ø1 8D 53 Ø3 4C EA
ØD29:39 ØD 38 A5 FB E9 28 85 E1
ØD31:FB A5 FC E9 ØØ 85 FC 6Ø 61
ØD39:A9 4Ø 8D Ø4 D4 2Ø CA 1Ø F6
ØD41:A9 B4 8D 5E Ø3 A9 Ø1 8D 43
ØD49:5D Ø3 8D 5F Ø3 A9 41 8D 49
ØD51:Ø4 D4 6Ø 2Ø 3C ØA A9 Ø3 11
ØD59:8D 46 Ø3 A9 Ø2 8D 2Ø DØ 1E
ØD61:8D 21 DØ 2Ø ED ØB 2Ø CA 4D
ØD69:1Ø EE Ø1 DØ EE Ø1 DØ AD 3F
ØD71:Ø1 DØ 8D Ø1 D4 C9 E9 9Ø 34
ØD79:Ø2 BØ Ø5 AD 55 Ø3 FØ E6 BB
ØD81:2Ø 39 ØD 2Ø 34 ØC C9 2Ø 23
ØD89:BØ F9 8D 4F Ø3 A9 AC 85 BE
ØD91:BØ A9 Ø6 85 B1 AØ ØØ 18 AF
ØD99:A5 BØ 6D 4F Ø3 85 BØ A5 8A
ØDA1:B1 69 ØØ 85 B1 B1 BØ C9 C6
```

```
ØDA9:6A FØ D8 C8 CØ Ø5 DØ F5 8E
ØDB1:AØ Ø2 A9 6A 91 BØ 18 A5 9D
ØDB9:BØ 69 ØØ 85 BØ A5 B1 69 C7
ØDC1:D4 85 B1 A9 Ø3 91 BØ AØ D8
ØDC9:ØØ A9 Ø3 91 BØ AØ ØØ AD 7D
ØDD1:Ø1 DØ C9 3C 9Ø 12 2Ø CA 75
ØDD9:1Ø CE Ø1 DØ CE Ø1 DØ AD A6
ØDE1:Ø1 DØ 8D Ø1 D4 4C DØ ØD F8
ØDE9:AD 1F DØ A9 ØØ 8D 2Ø DØ 9E
ØDF1:8D 21 DØ 8D 55 Ø3 2Ø 39 3E
ØDF9:ØD 68 68 2Ø 3C ØA 4C 35 9B
ØEØ1:Ø9 18 AD 57 Ø3 69 ØA 8D 32
ØEØ9:57 Ø3 AD 58 Ø3 69 ØØ 8D 18
ØE11:58 Ø3 38 AD 57 Ø3 ED 54 F2
ØE19:14 AD 58 Ø3 ED 55 14 9Ø 63
ØE21:ØC AD 57 Ø3 8D 54 14 AD 5D
ØE29:58 Ø3 8D 55 14 A9 13 2Ø C6
ØE31:D2 FF A2 ØB A9 2Ø 9D ØØ C4
ØE39:Ø4 CA 1Ø FA A9 1D 2Ø D2 9Ø
ØE41:FF A9 1D 2Ø D2 FF AE 57 B8
ØE49:Ø3 AD 58 Ø3 2Ø CD BD A9 EA
ØE51:2Ø 2Ø D2 FF A9 E8 AØ 13 25
ØE59:2Ø 1E AB AE 54 14 AD 55 11
ØE61:14 2Ø CD BD A9 2Ø 2Ø D2 Ø6
ØE69:FF A9 18 AØ 14 2Ø 1E AB Ø6
ØE71:AE 62 Ø3 A9 ØØ 2Ø CD BD 52
ØE79:A9 2Ø 2Ø D2 FF A9 2Ø 4C D6
ØE81:D2 FF 18 AD 57 Ø3 69 Ø1 7F
ØE89:8D 57 Ø3 AD 58 Ø3 69 ØØ 1F
ØE91:8D 58 Ø3 4C 13 ØE A9 ØØ D3
ØE99:8D 5B Ø3 85 AA A9 Ø4 85 95
ØEA1:AB AØ ØØ B1 AA C9 5A FØ F8
ØEA9:16 C9 57 FØ 12 E6 AA DØ 8F
ØEB1:Ø2 E6 AB 38 A5 AA E9 E6 14
ØEB9:A5 AB E9 Ø7 9Ø E5 6Ø A9 C7
ØEC1:Ø1 8D 5B Ø3 6Ø 2Ø 3C ØA 63
ØEC9:AD 56 14 8D 62 Ø3 A9 Ø3 23
ØED1:8D 64 Ø3 A9 93 2Ø D2 FF 8B
ØED9:A9 ØØ 8D 15 DØ 2Ø 69 ØA B1
ØEE1:2Ø 2E ØE A9 Ø1 8D 1D DØ 3F
ØEE9:A9 3D 8D Ø1 DØ A9 F3 8D 8E
ØEF1:F8 Ø7 8D 3C Ø3 A9 Ø6 8D 1A
ØEF9:3E Ø3 A9 Ø2 8D 46 Ø3 A9 8Ø
ØFØ1:Ø1 8D 51 Ø3 8D 53 Ø3 A9 C6
ØFØ9:5Ø 8D 3F Ø3 8D ØØ DØ A9 82
ØF11:ØØ 8D 4Ø Ø3 8D 1Ø DØ 8D A6
ØF19:41 Ø3 A2 Ø8 BD FD 13 9D 17
ØF21:4D Ø5 A9 Ø3 9D 4D D9 BD 2Ø
ØF29:Ø6 14 9D 75 Ø5 A9 Ø3 9D CC
ØF31:75 D9 BD ØF 14 9D 9D Ø5 8Ø
```

```
ØF39:A9 Ø3 9D 9D D9 CA 1Ø DC 71
ØF41:A9 22 8D 44 Ø3 A9 ØØ 8D FE
ØF49:45 Ø3 A9 2Ø 8D 42 Ø3 A9 27
ØF51:Ø1 8D 43 Ø3 8D 15 DØ 2Ø 6E
ØF59:CA 1Ø AD ØØ DC 29 ØF C9 ØA
ØF61:ØE FØ Ø7 C9 ØD FØ ØC 4C DØ
ØF69:7D ØF EE 62 Ø3 2Ø 6A ØE 89
ØF71:4C 77 ØF CE 62 Ø3 2Ø 6A 2C
ØF79:ØE 2Ø 51 ØA A5 C5 C9 Ø4 4D
ØF81:FØ ØB C9 Ø5 FØ 14 C9 Ø3 D2
ØF89:FØ 49 4C 58 ØF A9 ØØ 8D 2E
ØF91:5C Ø3 A9 Ø1 8D 1D DØ 4C B2
ØF99:B9 ØF A9 Ø1 8D 5C Ø3 A9 2B
ØFA1:ØØ 8D 1D DØ A9 2Ø 8D 44 Ø1
ØFA9:Ø3 A9 ØØ 8D 45 Ø3 A9 37 4D
ØFB1:8D 42 Ø3 A9 Ø1 8D 43 Ø3 E9
ØFB9:A2 Ø8 A9 2Ø 9D 4D Ø5 9D 2C
ØFC1:75 Ø5 9D 9D Ø5 CA 1Ø F4 D1
ØFC9:A9 ØØ 8D 57 Ø3 8D 58 Ø3 E5
ØFD1:4C 2E ØE A9 ØØ 85 C6 4C ED
ØFD9:C6 ØØ AD 69 Ø3 FØ Ø3 4C D5
ØFE1:31 EA EE 5D Ø3 AD 5D Ø3 93
ØFE9:C9 Ø1 9Ø 33 A9 ØØ 8D 5D 38
ØFF1:Ø3 AD 5F Ø3 DØ 17 EE 5E 38
ØFF9:Ø3 EE 5E Ø3 AD 5E Ø3 C9 Ø8
1ØØ1:FA BØ Ø2 9Ø 1A A9 Ø1 8D 1B
1ØØ9:5F Ø3 4C 2Ø 1Ø CE 5E Ø3 AØ
1Ø11:CE 5E Ø3 AD 5E Ø3 C9 32 3Ø
1Ø19:BØ Ø5 A9 ØØ 8D 5F Ø3 AD A5
1Ø21:5E Ø3 8D 16 D4 AD 1F DØ BØ
1Ø29:8D 55 Ø3 AD 46 Ø3 4A 8D Ø1
1Ø31:3E Ø3 AD 3D Ø3 CD 3E Ø3 89
1Ø39:BØ Ø6 EE 3D Ø3 4C 6Ø 1Ø FE
1Ø41:A9 ØØ 8D 3D Ø3 AD 3C Ø3 Ø6
1Ø49:C9 F9 BØ ØC EE 3C Ø3 AD BF
1Ø51:3C Ø3 8D F8 Ø7 4C 6Ø 1Ø CB
1Ø59:A9 FØ 8D 3C Ø3 DØ ED AD E4
1Ø61:41 Ø3 FØ 2B 38 AD 3F Ø3 AD
1Ø69:ED 42 Ø3 AD 4Ø Ø3 ED 43 79
1Ø71:Ø3 BØ 15 18 AD 3F Ø3 6D 41
1Ø79:46 Ø3 8D 3F Ø3 AD 4Ø Ø3 75
1Ø81:69 ØØ 8D 4Ø Ø3 4C BB 1Ø DC
1Ø89:A9 ØØ 8D 41 Ø3 FØ 2B 38 AE
1Ø91:AD 44 Ø3 ED 3F Ø3 AD 45 7F
1Ø99:Ø3 ED 4Ø Ø3 FØ 17 BØ 15 49
1ØA1:38 AD 3F Ø3 ED 46 Ø3 8D 7D
1ØA9:3F Ø3 AD 4Ø Ø3 E9 ØØ 8D 31
1ØB1:4Ø Ø3 4C BB 1Ø A9 Ø1 8D AE
1ØB9:41 Ø3 AD 3F Ø3 8D ØØ DØ Ø4
1ØC1:AD 4Ø Ø3 8D 1Ø DØ 4C 31 8F
```

```
1ØC9:EA A2 17 AØ ØØ A9 5Ø 85 C1
1ØD1:BØ A9 D8 85 B1 A9 77 85 DØ
1ØD9:B4 A9 D8 85 B5 AD 1B D4 A1
1ØE1:29 ØF C9 ØØ FØ F7 C9 Ø2 9Ø
1ØE9:FØ F3 C9 ØB FØ EF 91 BØ 84
1ØF1:91 B4 18 A5 BØ 69 28 85 66
1ØF9:BØ A5 B1 69 ØØ 85 B1 18 3A
11Ø1:A5 B4 69 28 85 B4 A5 B5 D2
11Ø9:69 ØØ 85 B5 CA 1Ø CE 6Ø 8Ø
1111:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 33
1119:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 3B
1121:ØØ ØØ ØØ 18 ØØ Ø8 18 1Ø 25
1129:1E 18 78 3F 99 FC 67 FF F2
1131:E6 4Ø FF Ø2 8Ø 3C Ø1 ØØ ED
1139:3C ØØ ØØ 18 ØØ ØØ 18 ØØ 2B
1141:Ø1 FF 8Ø ØØ ØØ ØØ ØØ ØØ F3
1149:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 6B
1151:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 73
1159:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 7B
1161:ØØ ØØ ØØ 1C ØØ Ø4 18 2Ø A5
1169:ØE 18 7Ø 1F 99 F8 37 FF B7
1171:EC 6Ø FF Ø6 4Ø 3C Ø2 ØØ 79
1179:3C ØØ ØØ 18 ØØ ØØ 18 ØØ 6B
1181:ØØ FF ØØ ØØ ØØ ØØ ØØ ØØ A3
1189:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ AB
1191:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ B3
1199:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ BB
11A1:ØØ ØØ ØØ 1C ØØ Ø2 18 ØØ BD
11A9:Ø7 18 EØ ØF D9 FØ 1F FF 33
11B1:F8 38 FF 1C 3Ø 3C ØC ØØ AA
11B9:3C ØØ ØØ 18 ØØ ØØ 18 ØØ AB
11C1:ØØ 7E ØØ ØØ ØØ ØØ ØØ ØØ 83
11C9:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ EB
11D1:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ F3
11D9:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ FB
11E1:ØØ ØØ ØØ 1E ØØ Ø3 18 ØØ 22
11E9:Ø7 98 8Ø ØF DB EØ ØF FF 37
11F1:FØ 1C 7E 38 18 3C 18 ØØ C8
11F9:3C ØØ ØØ 18 ØØ ØØ 18 ØØ EB
12Ø1:ØØ 3C ØØ ØØ ØØ ØØ ØØ ØØ 34
12Ø9:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 2D
1211:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 35
1219:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 3D
1221:ØØ ØØ ØØ 1E ØØ ØØ 5C ØØ DF
1229:ØØ F9 ØØ Ø1 FF 8Ø Ø3 FF E3
1231:CØ Ø7 7E EØ Ø6 3C EØ ØØ 38
1239:3C ØØ ØØ 18 ØØ ØØ 18 ØØ 2D
1241:ØØ Ø8 ØØ ØØ ØØ ØØ ØØ ØØ 67
1249:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 6D
1251:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 75
```

```
1259:00 00 00 00 00 00 00 00 7D
1261:00 00 00 18 00 00 18 00 37
1269:00 3C 00 00 3C 00 00 7E FC
1271:00 00 7E 00 00 7E 00 00 5F
1279:3C 00 00 18 00 00 18 00 6D
1281:00 18 00 00 00 00 00 00 AB
1289:00 00 00 00 00 00 00 00 AD
1291:00 00 00 00 00 00 00 00 B5
1299:00 00 00 00 00 00 00 00 BD
12A1:00 00 00 78 00 00 3A 00 C1
12A9:00 9F 00 01 FF 80 03 FF CD
12B1:C0 07 7E E0 07 3C 60 00 BF
12B9:3C 00 00 18 00 00 18 00 AD
12C1:00 10 00 00 00 00 00 00 E9
12C9:00 00 00 00 00 00 00 00 ED
12D1:00 00 00 00 00 00 00 00 F5
12D9:00 00 00 00 00 00 00 00 FD
12E1:00 00 00 78 00 00 18 C0 7E
12E9:01 19 E0 07 DB F0 0F FF 22
12F1:F0 1C 7E 38 18 3C 18 00 CA
12F9:3C 00 00 18 00 00 18 00 ED
1301:00 3C 00 00 00 00 00 00 36
1309:00 00 00 00 00 00 00 00 2F
1311:00 00 00 00 00 00 00 00 37
1319:00 00 00 00 00 00 00 00 3F
1321:00 00 00 38 00 00 18 40 3B
1329:07 18 E0 0F 9B F0 1F FF C4
1331:F8 38 FF 1C 30 3C 0C 00 2E
1339:3C 00 00 18 00 00 18 00 2F
1341:00 7E 00 00 00 00 00 00 07
1349:00 00 00 00 00 00 00 00 6F
1351:00 00 00 00 00 00 00 00 77
1359:00 00 00 00 00 00 00 00 7F
1361:00 00 00 38 00 04 18 20 6B
1369:0E 18 70 1F 99 F8 37 FF BB
1371:EC 60 FF 06 40 3C 02 00 7D
1379:3C 00 00 18 00 00 18 00 6F
1381:00 FF 00 00 00 00 00 00 A7
1389:00 00 00 00 00 00 00 00 AF
1391:00 00 00 00 00 00 00 00 B7
1399:00 00 00 00 00 00 00 00 BF
13A1:00 00 00 18 00 08 18 10 A9
13A9:1E 18 78 3F 99 FC 67 FF 77
13B1:E6 40 FF 02 80 3C 01 00 72
13B9:3C 00 00 18 00 00 18 00 AF
13C1:01 FF 80 00 00 00 00 00 78
13C9:00 00 00 00 00 00 00 FF EF
13D1:FF C3 C3 C3 C3 FF FF FF BB
13D9:3C 18 18 18 18 3C FF 00 5A
```

```
13E1:3C 7E FF FF 7E 3C ØØ 13 BD
13E9:1D 1D 1D 1D 1D 1D 1D 1D 1Ø
13F1:1D 1D 1D 1D 1D 48 49 47 47
13F9:48 3A 2Ø ØØ Ø6 31 2Ø ØE 1A
14Ø1:ØF 12 ØD Ø1 ØC Ø6 33 2Ø E5
14Ø9:Ø5 18 1Ø Ø5 12 14 Ø6 37 3Ø
1411:2Ø 11 15 Ø9 14 2Ø 2Ø 13 35
1419:1D 1D 1D 1D 1D 1D 1D 1D 41
1421:1D 1D 1D 1D 1D 1D 1D 1D 49
1429:1D 1D 1D 1D 1D 1D 1D 1D 51
1431:1D 1D 1D 45 47 47 53 3A 5F
1439:2Ø ØØ ØB Ø7 Ø5 C8 4Ø ØA 19
1441:ØØ 9B Ø3 Ø5 C8 14 ØF FØ A6
1449:FF FF ØØ ØØ 8Ø ØØ ØØ ØØ 75
1451:64 F3 3F ØØ ØØ ØA ØØ ØØ B8
```

Program 7
Miami Ice

Load and run your copy of MLX from tape or disk. When you first run MLX, you'll be asked to supply a starting address and an ending address for Miami Ice. The addresses you'll need are

Starting address: 0801
Ending address: 1320

After entering all the data from the listing below, be sure to save at least one copy before you exit MLX.

Miami Ice

For mistake-proof program entry, use "MLX: Machine Language Entry Program" to type in this program.

```
Ø8Ø1:ØC Ø8 ØA ØØ 9E 2Ø 32 3Ø 64
Ø8Ø9:36 32 ØØ ØØ ØØ 2Ø EØ ØE 11
Ø811:2Ø BC ØD A9 ØØ AØ 18 B9 Ø9
Ø819:A2 Ø8 99 ØØ D4 88 1Ø F7 9Ø
Ø821:2Ø C4 Ø8 2Ø 75 ØA AD Ø1 A5
Ø829:DC 29 1Ø FØ F9 A2 ØØ AØ FD
Ø831:1Ø 18 2Ø FØ FF A9 E2 AØ 6F
Ø839:ØB 2Ø 1E AB A6 B4 E8 A9 D8
Ø841:ØØ 2Ø CD BD A9 C8 85 F8 63
Ø849:A9 ØØ 85 F9 2Ø D6 ØD 2Ø 15
Ø851:52 ØA 2Ø 95 ØE A2 64 88 B6
Ø859:DØ FD CA DØ FA AD 1F DØ 55
Ø861:AD 1E DØ AD Ø1 DC 29 1Ø A2
Ø869:DØ E8 A9 4Ø 8D Ø4 D4 A9 25
Ø871:41 8D Ø4 D4 2Ø 1D Ø9 2Ø FA
Ø879:48 Ø9 CE 21 13 DØ Ø9 AD 77
Ø881:22 13 8D 21 13 2Ø 95 ØE 7D
Ø889:CE 25 13 DØ ØC AD 26 13 3Ø
Ø891:8D 25 13 2Ø 16 ØE 2Ø 4F 8E
Ø899:ØD AØ ØØ 88 DØ FD 4C 75 6D
Ø8A1:Ø8 ØØ Ø5 ØØ Ø1 ØØ 19 FØ 81
Ø8A9:ØØ 1E ØØ ØØ ØØ 89 ØØ ØØ 67
Ø8B1:ØA ØØ ØØ ØØ 2B ØØ ØØ Ø5 25
Ø8B9:F1 4F 2Ø CD BD A9 2Ø 2Ø 6C
Ø8C1:D2 FF 6Ø A9 ØØ 85 C3 85 Ø5
Ø8C9:C4 A9 ØØ 85 B4 A9 Ø3 85 D6
Ø8D1:BD A9 93 2Ø D2 FF A2 Ø3 7E
Ø8D9:8E 21 DØ E8 8E 2Ø DØ A2 5B
Ø8E1:Ø7 AØ ØB 18 2Ø FØ FF A9 EE
Ø8E9:EC AØ ØB 2Ø 1E AB A5 B4 9B
```

```
Ø8F1:18 69 31 8D A7 Ø5 A9 ØA 16
Ø8F9:2Ø 45 ØA AD Ø1 DC 4A BØ 48
Ø9Ø1:ØA A5 B4 C9 Ø6 FØ E7 E6 5F
Ø9Ø9:B4 1Ø E3 4A BØ Ø8 A5 B4 4Ø
Ø911:FØ DC C6 B4 1Ø D8 4A 4A B9
Ø919:4A BØ D3 6Ø C6 F7 DØ 26 DA
Ø921:A9 C8 85 F7 A5 F8 Ø5 F9 7F
Ø929:FØ 1C A5 F8 38 E9 Ø1 85 EF
Ø931:F8 A5 F9 E9 ØØ 85 F9 A2 B3
Ø939:18 AØ Ø7 18 2Ø FØ FF A6 4D
Ø941:F8 A5 F9 2Ø BB Ø8 6Ø AD E6
Ø949:23 13 1Ø Ø5 49 FF 18 69 E7
Ø951:Ø1 85 Ø2 AD 24 13 1Ø Ø5 F2
Ø959:49 FF 18 69 Ø1 18 65 Ø2 DE
Ø961:8D ØØ D4 AD 1F DØ 29 Ø8 46
Ø969:DØ 14 AD 1E DØ 85 Ø2 29 4A
Ø971:ØA C9 ØA FØ Ø9 A5 Ø2 29 57
Ø979:ØC C9 ØC FØ 3A 6Ø A9 4Ø 7B
Ø981:8D Ø4 D4 A9 8Ø 8D 12 D4 C3
Ø989:A9 81 8D 12 D4 A9 Ø8 8D 8E
Ø991:2Ø 13 A9 Ø5 2Ø 45 ØA AD D5
Ø999:2Ø 13 C9 ØD FØ Ø6 EE 2Ø 28
Ø9A1:13 4C 93 Ø9 A9 Ø7 8D 15 EC
Ø9A9:DØ A9 64 2Ø 45 ØA C6 BD BA
Ø9B1:FØ 5A 68 68 4C 4D Ø8 A9 B7
Ø9B9:4Ø 8D Ø4 D4 A9 ØØ 85 Ø2 77
Ø9C1:A5 F8 Ø5 F9 FØ 32 A5 F8 B9
Ø9C9:38 E9 Ø1 85 F8 A5 F9 E9 27
Ø9D1:ØØ 85 F9 A5 B4 85 Ø3 E6 87
Ø9D9:C3 DØ Ø2 E6 C4 2Ø 52 ØA Ø6
Ø9E1:E6 Ø2 A5 Ø2 8D Ø1 D4 A9 8Ø
Ø9E9:4Ø 8D Ø4 D4 A9 41 8D Ø4 BE
Ø9F1:D4 C6 Ø3 1Ø E2 4C C1 Ø9 56
Ø9F9:A9 4Ø 8D Ø4 D4 E6 B4 A9 38
ØAØ1:64 2Ø 45 ØA A9 Ø5 8D Ø1 16
ØAØ9:D4 4C 24 Ø8 2Ø 52 ØA A9 A7
ØA11:ØØ 8D 15 DØ A2 ØC AØ Ø5 C3
ØA19:18 2Ø FØ FF A9 9E AØ ØB 73
ØA21:2Ø 1E AB A2 ØB 2Ø 38 ØA BF
ØA29:A2 ØD 2Ø 38 ØA AD Ø1 DC 3F
ØA31:29 1Ø DØ F9 4C 21 Ø8 AØ 2F
ØA39:Ø5 18 2Ø FØ FF A9 C1 AØ B3
ØA41:ØB 4C 1E AB 85 Ø2 A9 ØØ F3
ØA49:85 A2 A5 A2 C5 Ø2 DØ FA 7A
ØA51:6Ø A9 9F 2Ø D2 FF A2 18 E9
ØA59:AØ 17 18 2Ø FØ FF A6 C3 21
ØA61:A5 C4 2Ø BB Ø8 A2 18 AØ D4
ØA69:26 18 2Ø FØ FF A6 BD A9 69
ØA71:ØØ 4C CD BD A9 93 2Ø D2 DC
ØA79:FF A9 Ø1 8D 21 DØ A9 Ø3 93
```

```
ØA81:8D 2Ø DØ A2 18 AØ ØØ 18 Ø4
ØA89:2Ø FØ FF A9 71 AØ ØB 2Ø C8
ØA91:1E AB AØ 27 A9 AØ 99 ØØ 29
ØA99:Ø4 99 98 Ø7 A9 Ø4 99 ØØ 2A
ØAA1:D8 99 98 DB 88 1Ø ED A9 63
ØAA9:ØØ 85 FB 85 FD A9 Ø4 85 1B
ØAB1:FC A9 D8 85 FE A2 18 AØ 75
ØAB9:ØØ A9 AØ 91 FB AØ 27 91 A7
ØAC1:FB A9 Ø4 91 FD AØ ØØ 91 DB
ØAC9:FD A5 FB 18 69 28 85 FB 3A
ØAD1:A5 FC 69 ØØ 85 FC A5 FD 8E
ØAD9:18 69 28 85 FD A5 FE 69 9F
ØAE1:ØØ 85 FE CA DØ D1 A9 ØØ Ø5
ØAE9:85 FB A9 Ø4 85 FC A6 B4 57
ØAF1:EØ Ø7 9Ø Ø2 A2 Ø6 BD Ø9 1C
ØAF9:ØD ØA AA BD FB ØC 85 Ø3 66
ØBØ1:BD FC ØC 85 Ø4 AØ FF C8 7A
ØBØ9:B1 Ø3 DØ 19 C8 B1 Ø3 8D Ø5
ØB11:Ø2 DØ 8D Ø4 DØ 8D ØØ DØ DB
ØB19:C8 B1 Ø3 8D Ø3 DØ 8D Ø5 B4
ØB21:DØ 8D Ø1 DØ 6Ø 84 Ø2 48 91
ØB29:48 29 1F 8D 28 13 68 4A 13
ØB31:4A 4A 4A 29 Ø6 AA AØ ØØ F6
ØB39:68 1Ø Ø2 AØ AØ 8C 29 13 6E
ØB41:A5 FB 18 7D 37 ØC 85 FB F4
ØB49:A5 FC 7D 38 ØC 85 FC AØ B5
ØB51:ØØ AD 29 13 FØ 11 91 FB 14
ØB59:A5 FB 85 FD A5 FC 18 69 8C
ØB61:D4 85 FE A9 Ø4 91 FD A4 C4
ØB69:Ø2 CE 28 13 DØ D2 FØ 97 B5
ØB71:12 9F 54 49 4D 45 52 3A F5
ØB79:2Ø 2Ø 2Ø 2Ø 2Ø 2Ø 2Ø 2Ø 8F
ØB81:2Ø 2Ø 53 43 4F 52 45 3A D6
ØB89:2Ø 2Ø 2Ø 2Ø 2Ø 2Ø 2Ø 2Ø 9F
ØB91:2Ø 2Ø 43 41 52 53 3A 2Ø BØ
ØB99:2Ø 9D 94 2Ø ØØ 12 9F 2Ø 63
ØBA1:92 47 41 4D 45 2Ø 4F 56 6F
ØBA9:45 52 2Ø 2D 2Ø 5Ø 52 45 F9
ØBB1:53 53 2Ø 46 49 52 45 42 ØF
ØBB9:55 54 54 4F 4E 12 2Ø ØØ ØA
ØBC1:12 9F 2Ø 2Ø 2Ø 2Ø 2Ø 2Ø BØ
ØBC9:2Ø 2Ø 2Ø 2Ø 2Ø 2Ø 2Ø 2Ø DF
ØBD1:2Ø 2Ø 2Ø 2Ø 2Ø 2Ø 2Ø 2Ø E7
ØBD9:2Ø 2Ø 2Ø 2Ø 2Ø 2Ø 2Ø 2Ø EF
ØBE1:ØØ 9C 12 53 43 52 45 45 C9
ØBE9:4E 2Ø ØØ 92 9C 4D 2Ø 49 FB
ØBF1:2Ø 41 2Ø 4D 2Ø 49 2Ø 2Ø C7
ØBF9:2Ø 49 2Ø 43 2Ø 45 11 11 F3
ØCØ1:11 9D 9D 9D 9D 9D 9D 9D D2
ØCØ9:9D 9D 9D 9D 9D 1F 53 43 38
```

```
ØC11:52 45 45 4E 2Ø 31 11 11 2A
ØC19:11 9D 9D 9D 9D 9D 9D 9D EA
ØC21:9D 9D 9D 9D 9D 5Ø 52 45 15
ØC29:53 53 2Ø 46 49 52 45 42 88
ØC31:55 54 54 4F 4E ØØ D8 FF AC
ØC39:FF FF 28 ØØ Ø1 ØØ 4B FF F4
ØC41:86 CD ØØ 37 B4 6A D1 A3 1A
ØC49:E6 66 C6 9Ø A3 E6 65 86 5A
ØC51:C5 E3 CC A3 E6 ØØ FB 42 8A
ØC59:45 EC 47 25 EE 8B 6A 43 48
ØC61:CD AA C2 Ø2 B4 45 65 ØØ Ø9
ØC69:2D C8 68 C3 45 A7 EB C3 F7
ØC71:83 E4 83 C3 E6 C3 83 E2 61
ØC79:A1 Ø4 83 47 E6 83 C3 E3 F8
ØC81:C3 45 E5 AB 83 C6 83 AA 2D
ØC89:83 C6 83 AA 83 C6 45 65 57
ØC91:83 ØØ FA BC 66 C6 A1 E3 ØC
ØC99:A2 C8 E1 27 E2 A1 65 C4 11
ØCA1:45 24 E1 63 Ø5 EA C2 82 B3
ØCA9:EA 81 C1 E5 46 81 Ø5 E3 54
ØCB1:Ø6 E4 A1 C2 82 AD 81 C1 F5
ØCB9:A6 C2 82 A4 84 E1 A1 82 E1
ØCC1:A1 E1 81 68 Ø5 C3 A1 E1 36
ØCC9:C3 EA C1 81 E2 Ø6 C1 ØØ 81
ØCD1:DC 46 65 D3 E9 64 E2 81 FB
ØCD9:C1 EE 8E A2 24 AA 24 A4 43
ØCE1:C9 E8 64 EC ØØ 73 89 46 9B
ØCE9:F2 82 E1 C2 EC C5 E2 C1 8A
ØCF1:A2 C4 AC C2 A1 82 B1 ØØ C8
ØCF9:2D C8 3F ØC 46 ØC 59 ØC A4
ØDØ1:6B ØC 95 ØC D3 ØC E8 ØC F3
ØDØ9:ØØ Ø6 Ø1 Ø5 Ø2 Ø4 Ø3 CE ØA
ØD11:27 13 DØ 39 A9 Ø7 8D 27 DC
ØD19:13 CE 2A 13 DØ ØF A9 4B 48
ØD21:8D 2A 13 AD 26 13 C9 Ø7 E1
ØD29:FØ Ø3 EE 26 13 AD 23 13 65
ØD31:FØ ØB 1Ø Ø6 EE 23 13 4C 5F
ØD39:3E ØD CE 23 13 AD 24 13 6C
ØD41:FØ ØB 1Ø Ø6 EE 24 13 4C 73
ØD49:4E ØD CE 24 13 6Ø AE 2Ø 81
ØD51:13 AD 23 13 18 7D 77 ØD A8
ØD59:C9 46 9Ø Ø4 C9 B9 9Ø Ø3 95
ØD61:8D 23 13 AD 24 13 18 7D 63
ØD69:7F ØD C9 46 9Ø Ø4 C9 B9 Ø6
ØD71:9Ø Ø3 8D 24 13 6Ø Ø1 ØØ A4
ØD79:FF FF FF ØØ Ø1 Ø1 FF FF 9F
ØD81:FF ØØ Ø1 Ø1 Ø1 ØØ A9 Ø1 28
ØD89:8D 19 DØ AD 1B 13 8D Ø6 EB
ØD91:DØ AD 1E 13 8D Ø7 DØ AD 4C
ØD99:1C 13 ØA ØA ØA 8D 1Ø DØ DF
```

```
ØDA1:AD 2Ø 13 18 69 D4 8D FB 34
ØDA9:Ø7 A9 FA 8D 12 DØ AD ØD 26
ØDB1:DC 29 Ø1 FØ Ø3 4C 31 EA 4A
ØDB9:4C BC FE A9 1B 8D 11 DØ A5
ØDC1:A9 7F 8D ØD DC A9 87 8D 3D
ØDC9:14 Ø3 A9 ØD 8D 15 Ø3 A9 25
ØDD1:81 8D 1A DØ 6Ø A9 ØØ 8D 97
ØDD9:1C 13 8D 23 13 8D 24 13 D4
ØDE1:A9 26 8D 1B 13 A9 3C 8D Ø3
ØDE9:1E 13 A9 Ø7 8D 2Ø 13 A5 36
ØDF1:A2 C5 A2 FØ FC A9 ØF 8D 6C
ØDF9:15 DØ A9 64 8D 21 13 8D F2
ØEØ1:22 13 A9 Ø7 8D 25 13 8D 4D
ØEØ9:26 13 A9 Ø7 8D 27 13 A9 7B
ØE11:37 8D 2A 13 6Ø 2Ø 1Ø ØD 53
ØE19:AD 23 13 3Ø 1A 18 6D 1A 6Ø
ØE21:13 8D 1A 13 AD 1B 13 69 Ø8
ØE29:ØØ 8D 1B 13 AD 1C 13 69 AA
ØE31:ØØ 8D 1C 13 4C 57 ØE 49 8A
ØE39:FF 69 Ø1 85 Ø2 AD 1A 13 36
ØE41:38 E5 Ø2 8D 1A 13 AD 1B 9F
ØE49:13 E9 ØØ 8D 1B 13 AD 1C DE
ØE51:13 E9 ØØ 8D 1C 13 AD 24 F6
ØE59:13 3Ø 18 18 6D 1D 13 8D 23
ØE61:1D 13 AD 1E 13 69 ØØ 8D 34
ØE69:1E 13 AD 1F 13 69 ØØ 8D CC
ØE71:1F 13 6Ø 49 FF 18 69 Ø1 B6
ØE79:85 Ø2 AD 1D 13 38 E5 Ø2 A7
ØE81:8D 1D 13 AD 1E 13 E9 ØØ F9
ØE89:8D 1E 13 AD 1F 13 E9 ØØ 4A
ØE91:8D 1F 13 6Ø AD Ø1 DC 4A 1A
ØE99:4A 4A BØ 12 2Ø D3 ØE EE FF
ØEA1:2Ø 13 AE 2Ø 13 EØ Ø8 DØ 67
ØEA9:Ø5 A2 ØØ 8E 2Ø 13 4A BØ 6C
ØEB1:ØD 2Ø D3 ØE CE 2Ø 13 1Ø E4
ØEB9:Ø5 A2 Ø7 8E 2Ø 13 4A BØ 5D
ØEC1:1Ø AD 26 13 C9 Ø2 FØ Ø9 88
ØEC9:CE 26 13 AD 26 13 8D 25 D1
ØED1:13 6Ø 48 A9 8Ø 8D ØB D4 58
ØED9:A9 81 8D ØB D4 68 6Ø AØ 37
ØEE1:ØØ B9 5A ØF 99 ØØ 35 B9 99
ØEE9:5A 1Ø 99 ØØ 36 B9 5A 11 C8
ØEF1:99 ØØ 37 B9 5A 12 99 ØØ AB
ØEF9:38 A9 ØØ 99 ØØ 39 88 DØ FC
ØFØ1:EØ AØ 3F B9 DA 12 99 8Ø ØE
ØFØ9:39 88 1Ø F7 AØ Ø2 A9 FF C7
ØF11:99 ØØ 39 99 3C 39 88 1Ø A4
ØF19:F7 AØ 36 A9 8Ø 99 Ø3 39 66
ØF21:88 88 88 1Ø F8 A9 ØC 8D CB
ØF29:5C 39 8D 62 39 A2 E4 8E 48
```

```
ØF31:F9 Ø7 E8 8E FA Ø7 E8 8E 68
ØF39:F8 Ø7 AØ Ø3 B9 56 ØF 99 B8
ØF41:27 DØ 88 1Ø F7 A9 Ø8 8D 3D
ØF49:1C DØ A9 ØØ 8D 25 DØ A9 2B
ØF51:Ø7 8D 26 DØ 6Ø Ø2 ØØ Ø2 35
ØF59:Ø2 ØØ ØØ ØØ ØØ ØØ ØØ ØØ 78
ØF61:ØØ ØØ ØØ ØØ 1Ø ØØ ØØ 6Ø 6Ø
ØF69:ØØ ØØ 28 ØØ ØØ A9 Ø4 Ø2 3D
ØF71:84 14 8A ØØ 52 A8 ØØ 4B A8
ØF79:EØ ØØ 2F AØ ØØ 2A 84 ØØ A9
ØF81:ØA 14 ØØ Ø2 5Ø ØØ ØØ 4Ø 8C
ØF89:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ A7
ØF91:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ AF
ØF99:ØØ ØØ ØØ ØØ Ø1 21 ØØ Ø1 45
ØFA1:A9 ØØ Ø1 21 ØØ ØØ 2Ø ØØ Ø7
ØFA9:ØØ 2Ø ØØ ØØ 2Ø ØØ ØØ 2Ø FØ
ØFB1:ØØ ØØ 2Ø ØØ ØØ A8 ØØ ØØ 76
ØFB9:B8 ØØ ØØ B8 ØØ Ø4 B8 4Ø 81
ØFC1:Ø5 A9 4Ø Ø5 A9 4Ø Ø4 A8 24
ØFC9:4Ø ØØ ØØ ØØ ØØ ØØ ØØ ØØ Ø8
ØFD1:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ EF
ØFD9:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ F7
ØFE1:ØØ ØØ Ø4 ØØ ØØ Ø9 ØØ ØØ A4
ØFE9:28 ØØ ØØ 6A ØØ ØØ 12 8Ø 67
ØFF1:1Ø ØØ A2 14 ØØ 2B 85 ØØ 65
ØFF9:ØB E1 ØØ ØA E8 ØØ 12 A8 CA
1ØØ1:ØØ 14 AØ ØØ Ø5 8Ø ØØ Ø1 65
1ØØ9:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 29
1Ø11:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 31
1Ø19:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 39
1Ø21:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 41
1Ø29:ØØ ØØ ØØ ØØ ØØ 54 ØØ ØØ 9A
1Ø31:54 4Ø ØØ 1Ø 4Ø ØA A8 AA B2
1Ø39:AB E8 AA AB E8 4Ø ØA A8 7E
1Ø41:4Ø ØØ 1Ø ØØ ØØ 54 ØØ ØØ D4
1Ø49:54 ØØ ØØ ØØ ØØ ØØ ØØ ØØ 93
1Ø51:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 71
1Ø59:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 79
1Ø61:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 81
1Ø69:ØØ ØØ ØØ ØØ Ø1 ØØ ØØ Ø5 96
1Ø71:8Ø ØØ 14 AØ ØØ 12 E8 ØØ 78
1Ø79:ØB E8 ØØ ØB A1 ØØ 2A 85 FØ
1Ø81:ØØ A2 14 12 8Ø 1Ø 6A ØØ Ø7
1Ø89:ØØ 28 ØØ ØØ Ø9 ØØ ØØ Ø4 FF
1Ø91:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ B1
1Ø99:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ B9
1ØA1:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ C1
1ØA9:ØØ ØØ ØØ Ø4 A8 4Ø Ø5 A9 Ø4
1ØB1:4Ø Ø5 A9 4Ø Ø4 B8 4Ø ØØ EF
1ØB9:B8 ØØ ØØ B8 ØØ ØØ A8 ØØ 13
```

```
1ØC1:ØØ A8 ØØ ØØ 2Ø ØØ ØØ 2Ø 2D
1ØC9:ØØ ØØ 2Ø ØØ ØØ 2Ø ØØ Ø1 6F
1ØD1:21 ØØ Ø1 A9 ØØ Ø1 21 ØØ 83
1ØD9:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ F9
1ØE1:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ Ø2
1ØE9:ØØ ØØ ØØ ØØ 4Ø ØØ Ø2 5Ø 6Ø
1ØF1:ØØ ØA 14 ØØ 2B 84 ØØ 2B AD
1ØF9:EØ ØØ 4A EØ ØØ 52 A8 ØØ 7C
11Ø1:14 8A ØØ Ø4 Ø2 84 ØØ ØØ 32
11Ø9:A9 ØØ ØØ 28 ØØ ØØ 6Ø ØØ 43
1111:ØØ 1Ø ØØ ØØ ØØ ØØ ØØ ØØ 37
1119:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 3B
1121:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 43
1129:ØØ ØØ ØØ 15 ØØ ØØ 15 ØØ C6
1131:ØØ Ø4 ØØ Ø1 2A AØ Ø1 2B 65
1139:EA AA 2B EA AA 2A AØ Ø1 CF
1141:Ø4 ØØ Ø1 15 ØØ ØØ 15 ØØ Ø1
1149:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 6B
1151:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 73
1159:ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ 7B
1161:Ø1 ØØ ØØ Ø7 4Ø 1Ø Ø7 4Ø Ø5
1169:1Ø Ø1 ØØ Ø1 3Ø 28 Ø8 8Ø 96
1171:38 Ø2 8Ø CØ Ø4 A2 Ø3 ØØ FC
1179:98 Ø1 ØC AB Ø1 Ø2 A8 8Ø 46
1181:Ø4 22 BØ Ø3 ØØ 6Ø ØC 1Ø 1E
1189:ØØ 28 CØ 4Ø 2Ø Ø1 DØ CØ 39
1191:Ø1 DØ ØØ ØØ 4Ø ØØ ØØ ØØ 6A
1199:ØØ ØØ ØØ ØØ ØØ Ø1 ØØ 4Ø FF
11A1:Ø7 4Ø 4Ø Ø7 4Ø ØØ 31 ØØ 34
11A9:Ø4 ØØ ØA 2Ø ØØ ØE ØØ 8Ø C9
11B1:ØØ Ø2 88 CØ 42 2A 83 ØØ 33
11B9:A8 ØØ 3Ø 2B 2Ø Ø2 A8 ØC 4F
11C1:1Ø 2Ø 8Ø ØØ ØØ 28 Ø3 ØØ AA
11C9:4Ø 3Ø ØØ ØØ ØØ 1Ø 1Ø 83 FB
11D1:ØØ 74 AØ ØØ 74 ØØ ØØ 1Ø D8
11D9:ØØ ØØ Ø7 4Ø 4Ø Ø1 ØØ ØØ E6
11E1:ØØ ØØ Ø4 3Ø ØØ ØØ ØØ ØØ 87
11E9:8Ø ØØ ØØ Ø2 8Ø ØØ ØØ 8A FA
11F1:8Ø ØØ 2A ØC 42 Ø8 Ø3 Ø2 94
11F9:8B ØØ CØ 2Ø EØ CA 28 Ø3 81
12Ø1:DØ A8 ØØ Ø2 ØØ 28 3Ø 2Ø F8
12Ø9:Ø2 ØC ØØ 4Ø CØ ØØ ØØ ØØ 3B
1211:1Ø Ø1 Ø3 ØØ Ø7 8Ø ØØ Ø7 1F
1219:ØØ ØØ Ø1 ØØ 1Ø 3Ø ØØ ØØ 9E
1221:ØØ ØØ ØØ ØØ ØØ Ø2 88 ØØ 5E
1229:Ø2 2Ø ØØ ØØ 32 AØ ØØ ØA 74
1231:Ø3 ØØ ØØ 8Ø ØØ 82 ØØ Ø2 EA
1239:A8 ØØ 2Ø Ø8 EØ 3A ØØ ØØ 26
1241:28 Ø2 Ø8 4Ø Ø2 Ø2 ØØ 2Ø 37
1249:ØØ ØØ ØØ ØØ 3Ø ØØ 1Ø ØØ ØF
```

```
1251:00 00 00 10 00 03 00 01 83
1259:00 40 30 00 00 00 00 08 9B
1261:00 00 0A 00 00 00 08 00 D6
1269:00 30 2A 00 00 A0 00 00 61
1271:00 0A 00 20 02 00 80 00 2B
1279:08 A0 20 00 B8 CC 00 00 C6
1281:28 00 00 20 00 02 00 02 C5
1289:00 00 22 00 00 00 00 C0 B2
1291:00 00 00 00 01 00 10 00 DD
1299:00 08 30 00 28 00 00 00 07
12A1:00 00 00 02 00 00 0C 00 FD
12A9:00 00 0A A0 00 00 80 00 1A
12B1:00 00 00 02 00 00 08 00 06
12B9:00 0A 80 00 0B 30 00 20 A9
12C1:00 00 00 00 00 00 00 00 E5
12C9:00 00 00 00 00 00 00 00 ED
12D1:00 00 08 02 01 00 12 00 43
12D9:00 00 00 00 3F FF FC 5F 51
12E1:FF FA 5F FF FA 6F FF F6 3D
12E9:6F FF F6 77 FF EE 77 FF C6
12F1:EE 7B FF DE 7B F3 DE 7C 40
12F9:00 3E 7B F3 DE 7B FF DE 20
1301:77 FF EE 77 FF EE 6F FF D2
1309:F6 6F FF F6 5F FF FA 5F 46
1311:FF FA 3F FF FC 00 00 00 C5
1319:FF 00 00 00 00 00 00 00 3F
```

Chapter 3
The Source Code

An Introduction to Assemblers

Todd Heimarck

If you plan to get into machine language, you need an assembler. Here's an overview of features to consider before purchasing one—and an up-to-date list of commercially available assemblers.

A machine language assembler turns *source code* written by a programmer into *object code,* the program which can be executed by the computer. (Source code is sometimes referred to as assembly language because it's a language for communicating with assemblers.)

If you're in the market for a machine language (ML) assembler, here's a brief introduction to the features you should expect to find.

Two Parts

Many assemblers have at least two parts—the assembler itself and an editor program. A standard editor allows you to enter source code, insert lines, delete lines, print out the program, and save a file to disk. It may be a separate program, or it may be built into the assembler.

An editor program is similar to a word processor, and, in some cases, you can substitute your favorite word processor if its files—usually sequential ASCII—are compatible with the files used by the assembler. The advantage of this is that you don't have to learn a new set of commands for using the assembler's text editor.

But there's a disadvantage as well. If the editor and assembler are separate programs, writing an ML program takes several steps. You have to load the editor before typing in the program. Once the program's finished, you have to save it to disk, then load the assembler, and finally assemble the file into object code. To make changes to the program—if there are bugs, for example—you have to go back to the editor, load the file, and edit it.

An integrated editor/assembler gives you both programs in memory at the same time. It's not necessary to go through the process of loading one, then loading the other, switching back and forth as you develop a program. Having both available saves time, especially for programmers who like to write a few lines, assemble them, test the program, make a few changes, and continue from there. One disadvantage of an integrated package is that two programs take up more memory than a single program; you may have less memory available for storing the source code.

In addition to separate editors and integrated editor/assemblers, there's a third type. Some assemblers let you create source code with the built-in BASIC editor. You write and edit the source code with line numbers, as if it were a BASIC program. To look at the program, you LIST it. To save to disk, just SAVE. Relying on the BASIC editor means it's not necessary to load a separate program that takes up valuable memory. With this kind of assembler, you can usually have your favorite programming utilities (like COMPUTE! Publications' "BASIC Aid" and "MetaBASIC," or Commodore's "DOS Wedge") in memory at the same time as the assembler and thus have such additional conveniences as automatic line numbering.

When you're writing ML programs, you'll be spending a lot of time in the editor, so it's important to find one you feel comfortable with. Some programmers prefer an editor that's like a word processor, while others lean to assemblers that let you write with the built-in BASIC editor.

Remember the Mnemonics

The source code for an ML program is made up of *mnemonics*, English-like abbreviations for ML instructions. On eight-bit Commodore machines, including the 64 and 128, all mnemonics are three letters (an exception is CP/M mode on the 128).

The three-letter abbreviation for "LoaD the Accumulator with a number" is *LDA*. In source code, for example, the line **LDA #0** means load the number 0 into the accumulator, a register inside the main chip. When the file is assembled, the source code instruction LDA #0 is turned into the object code (numbers) 169 and 0. The primary task of an assembler is this translation from mnemonics to numbers.

A disk-based assembler requires the source code to be stored on disk. When it comes time to assemble, you save the file and tell the assembler to start creating the object code from the source code on disk. RAM-based assemblers (which almost always have a disk option as well) read through the source code in memory. RAM-based assemblers are much faster than disk-based assemblers, which are hampered by Commodore's traditionally slow serial communications.

Depending on your instructions to the assembler, the object code (the executable ML program) can be sent to a disk file or to memory. When a program is assembled to disk, you have to load the program before testing it. If the assembler places the object code directly in memory, you can SYS to it immediately after assembly.

In BASIC, GOTO and GOSUB require a line number reference—GOTO 500, for instance. The corresponding ML instructions JMP (jump) and JSR (jump to subroutine) are followed by memory addresses, which serve the same function as line numbers in a BASIC program. But in a source code file, the current memory location (also known as the program counter) is not usually evident (the source code listings in this book are an exception—the memory location of each instruction is at the far left of the column). You may have a general subroutine for reading the joystick, but if you don't know the exact starting address, you can't JSR to it in the program.

Labeling

Assemblers solve this problem by allowing *labels* within your source code. You place the label at the beginning of a subroutine, and elsewhere in the program you can call it by name. You might label the joystick routine something like READJOY and then JSR READJOY when you need to use it. Labels are also used extensively within loops and branches.

Equates are another sort of label. At the start of the program, you can give names to important memory locations, so you can use a name instead of a number. The screen background color on the 64 is stored in 53281, which could be called BACKGROUND. To modify the screen color, you would store a value into BACKGROUND. When the assembler got to that line, it would know that BACKGROUND stands for location 53281.

Assemblers that allow labels and equates are usually advertised as *two-pass, three-pass,* or *label-based.* In the first pass through the source file, the assembler sets up a list of labels and the equivalent memory locations or values. During the final pass, the file is assembled and the numbers are substituted for the labels.

The Commodore 128 includes the MONITOR command, which enables the built-in machine language monitor. Although the 64 doesn't come with a monitor, programs like *Micromon* and *Supermon* are widely available.

Monitors generally have a *simple assembler,* which is suitable for writing short routines, but lacking many features found in a full-fledged assembler. Simple assemblers don't support labels, for one thing, leaving it up to the programmer to calculate the addresses where subroutines start and end. Another shortcoming is that it's often difficult to insert new lines or make corrections; if an assembler/editor is like a word processor, then a monitor is like a typewriter—to add a sentence may require retyping the whole page.

The commands for disassembly and memory display make ML monitors useful for debugging, though, so some assemblers for the 64 include a monitor to complement the other programs in the package.

What to Look For

At the very least, a stripped-down assembler should include the following features:

- A way to type, save, and load source files, whether it's a separate editor or the built-in BASIC editor.
- Assembly from disk or memory, with object files written to disk or memory.
- Labels and equates.
- Error checking, to flag misspelled labels, duplicate labels, and syntax mistakes.

In addition, many assemblers offer additional features like macro-instructions—usually called *macros* for short. A macro is essentially a fill-in-the-blanks miniprogram. A macro that opens a sequential file for reading, for example, might call several ROM routines and leave a blank for the filename. Within your source code, you would provide the name of the OPEN macro and tell it the name of the file to be opened for reading. When the assembler assembled that line, it would find the macro in the library and insert as many instructions as necessary in the object code; it would also fill in the name of the sequential file you wanted to open for reading. Macros, however, are often rather cumbersome to use—the complexities of passing parameters to them can make them more trouble than they're worth.

Most assemblers allow you to insert comments in the source file. These notes to yourself are like REM statements in BASIC. Some assemblers offer conditional assembly, which is helpful if you're writing for several different systems, a 64 and a 128, for example. Like macros, however, conditional assembly is an additional level of complexity (and thus an additional potential source of bugs). Writing dual-purpose source code is, many programmers think, of dubious value.

Some also include utility programs. An *unassembler,* for example, reads an object file and turns it into source code which can be loaded, edited, and reassembled. You may also find a disk *disassembler,* which reads a machine language file directly from disk and lists it out as mnemonics. Occasionally, an assembler package will include the source code for the assembler itself. This is valuable because you can study the program (if you're just learning ML) or change the program, adding new features (if you're beyond the beginner's stage). That's why we've included the complete source code for the seven games in this book.

One final note: In many cases, source files are not compatible between different assemblers. If you plan to trade source code with a friend, it would be a good idea to use the same assembler. This is also true if you're studying ML in a class sponsored by a college or a user group.

What follows is a list of commercial assemblers for the Commodore 64 and 128. For more information, contact the manufacturer.

Abacus Software
P.O. Box 7211
Grand Rapids, MI 49510
(616) 241-5510
Assembler Monitor 64 (64)
$39.95

Aquarian Software
P.O. Box 22184
Portland, OR 97222
(503) 654-2641
Total Software Development System
(64 and 128 versions on same disk)
$49.95
Instant Editor Assembler (64)
$17.95

Commodore Business Machines
1200 Wilson Drive
West Chester, PA 19380
(215) 431-9100
Commodore 64 Macro Assembler
Development System (64)
$49.95

COMPUTE! Publications
P.O. Box 5038
F.D.R. Station
New York, NY 10150
(800) 346-6767
(212) 887-8575 (in NY)
LADS (64/128)
$12.95–$16.95, depending on
version (plus shipping)

Eastern House Software
3239 Linda Drive
Winston-Salem, NC 27106
(919) 748-8446
MAE Macro Assembler/Text Editor
(64)
$19.95 (plus shipping)

FS! Software
P.O. Box 635
Faribault, MN 55021
(507) 332-8122
The Machine Shop (64)
$39.95

Hughes Associates
45341 Harmony Lane
Belleville, MI 48111
(313) 699-1931
Freedom Assembler
$39.95 (64)
$49.95 (128)

Micol Systems
9 Lynch Road
Toronto, Ontario
Canada M2J 2V6
(416) 495-6864
M-2000 Machine Language Development System (64)
$49.95

Nu-Age Software
2311-28th Street N.
St. Petersburg, FL 33713
(813) 323-8389
Rebel Assembler Editor (64)
$29.95

Pro-Line Software, Ltd.
755 The Queensway East
Unit 8
Mississauga, Ontario
Canada L4Y 4C5
(416) 273-6350
PAL (64)
$49.95
Buddy 128 Assembler (128)
$69.95

Robin's Software
10349 Zinran Circle
Bloomington, MN 55438
(612) 944-8654
Instant Editor Assembler (64)
$5.00 plus $2.00 shipping and
handling

Roger Wagner Publishing
P.O. Box 582
Santee, CA 92071
(619) 562-3221
Merlin 64 Macro Assembler (64)
$49.95

The Source Code

The source code for each of the seven games in *COMPUTE!'s Machine Language Games for the Commodore 64* was originally developed at COMPUTE! Publications using the *PAL* assembler from Pro-Line Software.

PAL uses several pseudo-ops, such as .ASC and .WORD, that aren't found in most other assemblers. Most pseudo-ops, however, are in standard MOS 6502 notation: *= updates the program counter (some assemblers use .ORG instead); **.BYT** or **.BYTE** assembles a list of numbers (some assemblers use .DB or .DW); **.WOR** or **.WORD** assembles a list of addresses into low-byte/high-byte format; **.ASC** is used to assemble an ASCII character string (many assemblers use .BYTE for this also); the pseudo-op < extracts the low byte of a 16-bit expression, while > extracts the high byte of the low byte of a 16-bit expression (some assemblers reverse the use of < and >, and others use $255 and /256 for the same effect); and = is used to assign an expression to a label (some assemblers use .EQU).

You'll find comments in almost every game's source code, though some are more heavily commented than others. You needn't type them in (if that's what you're using the source code for), but if you do, remember to precede each comment with a semicolon (;).

Since all source code was written with *PAL*, it's not surprising that the listings which follow are also in *PAL* format. The listings aren't source code in the strictest sense, but rather source code that has been assembled by *PAL*. Your assembler may display things differently.

Here's a typical line from the listing for "CUT-OFF!" (the boldface headings indicate each column's content). Note that pseudo-ops like .ASC and .BYT don't have a final operand, but instead list the ASCII string or a series of numbers.

Address	**Opcode**	**Operand**	**Label**	**Mnemonic**	**Operand**
C006	AD	42 03	MAINLOOP	LDA	STEPS

Notes on the Code

As mentioned above, the listings that follow show the *assembled* source code, not the raw source code you would actually enter for your assembler. Thus, you can't type the listings in the format shown. To enter the code so that you can assemble it yourself, you'll need to know the source code format your assembler expects. For example, for Commodore's *Macro Assembler Development System* you would type in just the labels, mnemonics, and operands (the three rightmost columns of the listings). *PAL,* on the other hand, requires a line number (just like BASIC) before every line, but allows you to use the colon to combine more than one instruction per line.

You'll come across some odd-looking characters in several of the games' listings. These characters, limited to the .ASC pseudo-op, are found only within quotation marks ("). They're not meant to be typed in as shown, but instead represent key(s) you should press.

Words within braces are the most common such characters: {DOWN} means to press the cursor-down key; {5 LEFT} means to press the cursor-left key five times.

To indicate that a key should be *shifted* (hold down the SHIFT key while pressing another key), the character is underlined. For example, A̲ means hold down the SHIFT key and press A. You may see strange characters on your screen, but that's to be expected. If you find a pair of braces enclosing a number followed by an underlined key, for instance, {8 A̲}, type the key as many times as indicated (in the example, enter eight SHIFTed A's).

If a key is enclosed in special brackets, [<>], hold down the Commodore key (at the lower-left corner of the keyboard) and press the indicated character.

Refer to the following table:

When You Read:	Press:		See:
{CLR}	SHIFT	CLR/HOME	♥
{HOME}		CLR/HOME	S
{UP}	SHIFT	↑ CRSR ↓	●
{DOWN}		↑ CRSR ↓	Q
{LEFT}	SHIFT	← CRSR →	▏
{RIGHT}		← CRSR →	]
{RVS}	CTRL	9	R
{OFF}	CTRL	0	▬
{BLK}	CTRL	1	■
{WHT}	CTRL	2	E
{RED}	CTRL	3	£
{CYN}	CTRL	4	◥
{PUR}	CTRL	5	▒
{GRN}	CTRL	6	↑
{BLU}	CTRL	7	←
{YEL}	CTRL	8	π

When You Read:	Press:		See:
Ɛ 1 Ǝ	COMMODORE	1	♣
Ɛ 2 Ǝ	COMMODORE	2	◤
Ɛ 3 Ǝ	COMMODORE	3	▣
Ɛ 4 Ǝ	COMMODORE	4	◘
Ɛ 5 Ǝ	COMMODORE	5	┴
Ɛ 6 Ǝ	COMMODORE	6	▌
Ɛ 7 Ǝ	COMMODORE	7	✚
Ɛ 8 Ǝ	COMMODORE	8	▚
{ F1 }		f1	▬
{ F2 }	SHIFT	f1	◣
{ F3 }		f3	▬
{ F4 }	SHIFT	f3	◣
{ F5 }		f5	▌
{ F6 }	SHIFT	f5	◢
{ F7 }		f7	▌
{ F8 }	SHIFT	f7	■
←	←		←
↑	SHIFT	↑	π

CUT-OFF! Source Code

Source code by Tom R. Halfhill

```
C000                     *=   $C000
C000          NEXTPOS    =    $F7
C000          SRAM       =    $FD
C000          CRAM       =    $FB
C000          BLACK      =    0
C000          RED        =    2
C000          GREEN      =    5
C000          BLUE       =    6
C000          GRAY3      =    15
C000          LEVEL      =    828
C000          SCORE1     =    829
C000          SCORE2     =    831
C000          PLAYER     =    833
C000          STEPS      =    834
C000          LIVES1     =    836
C000          LIVES2     =    837
C000          POS        =    838
C000          LATCH2     =    842
C000          LATCH1     =    843
C000          SHAPE2     =    844
C000          SHAPE1     =    845
C000          JOY1       =    56321
C000          JOY2       =    56320
C000          CHROUT     =    $FFD2
C000          PLOT       =    $FFF0
C000          PRNUM      =    $BDCD
C000          VOLUME     =    54296
C000          WAVEFORM   =    54276
C000          ATTACK     =    54277
C000          RELEASE    =    54278
C000          V1LO       =    54272
C000          V1HI       =    54273
C000          BORDER     =    53280
C000 20 C1 C1            JSR  NEWGAME
C003 4C 06 C0            JMP  MAINLOOP
;
; main loop
;
C006 AD 42 03 MAINLOOP   LDA  STEPS
C009 18                  CLC
C00A 69 01               ADC  #1
C00C 8D 42 03            STA  STEPS
C00F AD 43 03            LDA  STEPS+1
C012 69 00               ADC  #0
C014 8D 43 03            STA  STEPS+1
C017 AE 3C 03            LDX  LEVEL
C01A 20 8A C3            JSR  DELAY
C01D A9 01               LDA  #1
C01F 8D 41 03            STA  PLAYER
C022 20 C4 C3            JSR  READJOY
C025 A2 02               LDX  #2
C027 A1 F7               LDA  (NEXTPOS,X)
C029 C9 20               CMP  #32
C02B F0 20               BEQ  NOHIT1
C02D 20 93 C3            JSR  MOVE
C030 20 66 C5            JSR  CRASH
C033 20 D0 C2            JSR  SCORE
C036 AD 44 03            LDA  LIVES1
C039 C9 00               CMP  #0
C03B F0 53               BEQ  OUTLOOP
C03D AD 45 03            LDA  LIVES2
```

```
C040 C9 00               CMP  #0
C042 F0 4C               BEQ  OUTLOOP
C044 20 BE C0            JSR  NEWSCREEN
C047 20 43 C4            JSR  SETUP
C04A 4C 06 C0            JMP  MAINLOOP
C04D 20 93 C3 NOHIT1     JSR  MOVE
C050 A9 00               LDA  #0
C052 8D 41 03            STA  PLAYER
C055 20 C4 C3            JSR  READJOY
C058 A2 00               LDX  #0
C05A A1 F7               LDA  (NEXTPOS,X)
C05C C9 20               CMP  #32
C05E F0 20               BEQ  NOHIT2
C060 20 93 C3            JSR  MOVE
C063 20 66 C5            JSR  CRASH
C066 20 D0 C2            JSR  SCORE
C069 AD 44 03            LDA  LIVES1
C06C C9 00               CMP  #0
C06E F0 20               BEQ  OUTLOOP
C070 AD 45 03            LDA  LIVES2
C073 C9 00               CMP  #0
C075 F0 19               BEQ  OUTLOOP
C077 20 BE C0            JSR  NEWSCREEN
C07A 20 43 C4            JSR  SETUP
C07D 4C 06 C0            JMP  MAINLOOP
C080 20 93 C3 NOHIT2     JSR  MOVE
C083 AD 01 DC            LDA  JOY1
C086 2D 00 DC            AND  JOY2
C089 29 10               AND  #16
C08B F0 06               BEQ  FREEZE
C08D 4C 06 C0            JMP  MAINLOOP
C090 4C F4 C4 OUTLOOP    JMP  ENDGAME
C093 AD 01 DC FREEZE     LDA  JOY1
C096 2D 00 DC            AND  JOY2
C099 29 10               AND  #16
C09B F0 F6               BEQ  FREEZE
C09D A2 FA               LDX  #250
C09F 20 8A C3            JSR  DELAY
C0A2 AD 01 DC PAUSE      LDA  JOY1
C0A5 2D 00 DC            AND  JOY2
C0A8 29 10               AND  #16
C0AA D0 F6               BNE  PAUSE
C0AC AD 01 DC PAUSE2     LDA  JOY1
C0AF 2D 00 DC            AND  JOY2
C0B2 29 10               AND  #16
C0B4 F0 F6               BEQ  PAUSE2
C0B6 A2 FA               LDX  #250
C0B8 20 8A C3            JSR  DELAY
C0BB 4C 06 C0            JMP  MAINLOOP
;
;newscreen routine
;
C0BE A9 93    NEWSCREEN  LDA  #147
C0C0 20 D2 FF            JSR  CHROUT
C0C3 A9 0F               LDA  #GRAY3
C0C5 8D 20 D0            STA  53280
C0C8 8D 21 D0            STA  53281
C0CB A2 00               LDX  #0
C0CD A9 A0    LOOP1      LDA  #160
C0CF 9D 00 04            STA  1024,X
C0D2 A9 00               LDA  #BLACK
```

```
C0D4 9D 00 D8              STA  55296,X
C0D7 E8                    INX
C0D8 E0 28                 CPX  #40
C0DA D0 F1                 BNE  LOOP1
C0DC A2 00                 LDX  #0
C0DE A9 A0       LOOP2     LDA  #160
C0E0 9D C0 07              STA  1984,X
C0E3 A9 00                 LDA  #BLACK
C0E5 9D C0 DB              STA  56256,X
C0E8 E8                    INX
C0E9 E0 28                 CPX  #40
C0EB D0 F1                 BNE  LOOP2
C0ED A9 00                 LDA  #0
C0EF 85 FD                 STA  SRAM
C0F1 A9 04                 LDA  #4
C0F3 85 FE                 STA  SRAM+1
C0F5 A9 00                 LDA  #0
C0F7 85 FB                 STA  CRAM
C0F9 A9 D8                 LDA  #216
C0FB 85 FC                 STA  CRAM+1
C0FD A2 00                 LDX  #0
C0FF A9 A0       LOOP3     LDA  #160
C101 A0 00                 LDY  #0
C103 91 FD                 STA  (SRAM),Y
C105 A9 00                 LDA  #BLACK
C107 91 FB                 STA  (CRAM),Y
C109 A0 27                 LDY  #39
C10B A9 A0                 LDA  #160
C10D 91 FD                 STA  (SRAM),Y
C10F A9 00                 LDA  #BLACK
C111 91 FB                 STA  (CRAM),Y
C113 18                    CLC
C114 A5 FD                 LDA  SRAM
C116 69 28                 ADC  #40
C118 85 FD                 STA  SRAM
C11A A5 FE                 LDA  SRAM+1
C11C 69 00                 ADC  #0
C11E 85 FE                 STA  SRAM+1
C120 18                    CLC
C121 A5 FB                 LDA  CRAM
C123 69 28                 ADC  #40
C125 85 FB                 STA  CRAM
C127 A5 FC                 LDA  CRAM+1
C129 69 00                 ADC  #0
C12B 85 FC                 STA  CRAM+1
C12D E8                    INX
C12E E0 19                 CPX  #25
C130 D0 CD                 BNE  LOOP3
C132 A9 05                 LDA  #GREEN
C134 A2 10                 LDX  #16
C136 9D 00 D8    LOOP4     STA  55296,X
C139 E8                    INX
C13A E0 18                 CPX  #24
C13C D0 F8                 BNE  LOOP4
C13E A9 83                 LDA  #131
C140 8D 10 04              STA  1040
C143 A9 95                 LDA  #149
C145 8D 11 04              STA  1041
C148 A9 94                 LDA  #148
C14A 8D 12 04              STA  1042
C14D A9 AD                 LDA  #173
C14F 8D 13 04              STA  1043
C152 A9 8F                 LDA  #143
C154 8D 14 04              STA  1044
C157 A9 86                 LDA  #134
C159 8D 15 04              STA  1045
C15C 8D 16 04              STA  1046
C15F A9 A1                 LDA  #161
C161 8D 17 04              STA  1047
C164 A9 06                 LDA  #BLUE
C166 8D C7 D9              STA  55751
```

```
C169 A9 02                 LDA  #RED
C16B 8D D1 D9              STA  55761
C16E A9 51                 LDA  #81
C170 8D C7 05              STA  1479
C173 A9 57                 LDA  #87
C175 8D D1 05              STA  1489
C178 A9 00                 LDA  #0
C17A 8D 42 03              STA  STEPS
C17D 8D 43 03              STA  STEPS+1
C180 A9 07                 LDA  #7
C182 8D 4B 03              STA  LATCH1
C185 A9 0B                 LDA  #11
C187 8D 4A 03              STA  LATCH2
C18A A9 D1                 LDA  #$D1
C18C 8D 46 03              STA  POS
C18F A9 05                 LDA  #$5
C191 8D 47 03              STA  POS+1
C194 A9 C7                 LDA  #$C7
C196 8D 48 03              STA  POS+2
C199 A9 05                 LDA  #$5
C19B 8D 49 03              STA  POS+3
C19E 18                    CLC
C19F A2 00                 LDX  #0
C1A1 A0 07                 LDY  #7
C1A3 20 F0 FF              JSR  PLOT
C1A6 AE 3D 03              LDX  SCORE1
C1A9 AD 3E 03              LDA  SCORE1+1
C1AC 20 CD BD              JSR  PRNUM
C1AF 18                    CLC
C1B0 A2 00                 LDX  #0
C1B2 A0 1D                 LDY  #29
C1B4 20 F0 FF              JSR  PLOT
C1B7 AE 3F 03              LDX  SCORE2
C1BA AD 40 03              LDA  SCORE2+1
C1BD 20 CD BD              JSR  PRNUM
C1C0 60                    RTS
;
;newgame routine
;
C1C1 A9 00       NEWGAME   LDA  #0
C1C3 8D 3D 03              STA  SCORE1
C1C6 8D 3E 03              STA  SCORE1+1
C1C9 8D 3F 03              STA  SCORE2
C1CC 8D 40 03              STA  SCORE2+1
C1CF A9 51                 LDA  #81
C1D1 8D 4D 03              STA  SHAPE1
C1D4 A9 57                 LDA  #87
C1D6 8D 4C 03              STA  SHAPE2
C1D9 20 BE C0              JSR  NEWSCREEN
C1DC A9 05                 LDA  #GREEN
C1DE A2 09                 LDX  #9
C1E0 9D F0 D8    LOOP5     STA  55536,X
C1E3 E8                    INX
C1E4 E0 1E                 CPX  #30
C1E6 D0 F8                 BNE  LOOP5
C1E8 A2 09                 LDX  #9
C1EA 9D 40 D9    LOOP6     STA  55616,X
C1ED E8                    INX
C1EE E0 1E                 CPX  #30
C1F0 D0 F8                 BNE  LOOP6
C1F2 8D A3 D9              STA  55715
C1F5 A2 00                 LDX  #0
C1F7 BD A4 C2    LOOP7     LDA  TABLE1,X
C1FA F0 06                 BEQ  NEXT1
C1FC 9D F9 04              STA  1273,X
C1FF E8                    INX
C200 D0 F5                 BNE  LOOP7
C202 A2 00       NEXT1     LDX  #0
C204 BD BA C2    LOOP8     LDA  TABLE2,X
C207 F0 06                 BEQ  NEXT2
C209 9D 49 05              STA  1353,X
```

```
C20C E8                  INX
C20D D0 F5               BNE   LOOP8
C20F A9 30     NEXT2     LDA   #48
C211 8D 3C 03            STA   LEVEL
C214 8D A3 05            STA   1443
C217 A2 64     LOOP9     LDX   #100
C219 20 8A C3            JSR   DELAY
C21C AD 01 DC            LDA   JOY1
C21F 29 0F               AND   #15
C221 C9 0E               CMP   #14
C223 F0 21               BEQ   LEVUP
C225 C9 0D               CMP   #13
C227 F0 0A               BEQ   LEVDOWN
C229 AD 01 DC            LDA   JOY1
C22C 29 10               AND   #16
C22E F0 3F               BEQ   SETLEV
C230 4C 17 C2            JMP   LOOP9
C233 AD 3C 03  LEVDOWN   LDA   LEVEL
C236 38                  SEC
C237 E9 01               SBC   #1
C239 C9 2F               CMP   #47
C23B F0 1C               BEQ   LOW
C23D 8D 3C 03            STA   LEVEL
C240 8D A3 05            STA   1443
C243 4C 17 C2            JMP   LOOP9
C246 AD 3C 03  LEVUP     LDA   LEVEL
C249 18                  CLC
C24A 69 01               ADC   #1
C24C C9 3A               CMP   #58
C24E F0 14               BEQ   HIGH
C250 8D 3C 03            STA   LEVEL
C253 8D A3 05            STA   1443
C256 4C 17 C2            JMP   LOOP9
C259 A9 39     LOW       LDA   #57
C25B 8D 3C 03            STA   LEVEL
C25E 8D A3 05            STA   1443
C261 4C 17 C2            JMP   LOOP9
C264 A9 30     HIGH      LDA   #48
C266 8D 3C 03            STA   LEVEL
C269 8D A3 05            STA   1443
C26C 4C 17 C2            JMP   LOOP9
C26F AD 01 DC  SETLEV    LDA   JOY1
C272 29 10               AND   #16
C274 F0 F9               BEQ   SETLEV
C276 A2 FA               LDX   #250
C278 20 8A C3            JSR   DELAY
C27B AD 3C 03            LDA   LEVEL
C27E 38                  SEC
C27F E9 30               SBC   #48
C281 AA                  TAX
C282 A9 32               LDA   #50
C284 8D 3C 03            STA   LEVEL
C287 E0 00     LEVLOOP   CPX   #0
C289 F0 0D               BEQ   START
C28B AD 3C 03            LDA   LEVEL
C28E 38                  SEC
C28F E9 05               SBC   #5
C291 8D 3C 03            STA   LEVEL
C294 CA                  DEX
C295 4C 87 C2            JMP   LEVLOOP
C298 20 BE C0  START     JSR   NEWSCREEN
C29B A9 0A               LDA   #10
C29D 8D 44 03            STA   LIVES1
C2A0 8D 45 03            STA   LIVES2
C2A3 60                  RTS
C2A4 0D 0F 16  TABLE1    .BYT  13,15,22,5,32,10,15,
                               25,19,20
C2AE 09 03 0B            .BYT  9,3,11,32,21,16,47,4
                               ,15,23,14,0
C2BA 14 0F 20  TABLE2    .BYT  20,15,32,3,8,15,15,1
                               9,5,32
```

```
C2C4 13 0B 09            .BYT  19,11,9,12,12,32,12,
                               5,22,5,12,0
;
; score routine
;
C2D0 AD 46 03  SCORE     LDA   POS
C2D3 CD 48 03            CMP   POS+2
C2D6 F0 03               BEQ   CHEKHI
C2D8 4C 5C C3            JMP   NOHEADON
C2DB AD 47 03  CHEKHI    LDA   POS+1
C2DE CD 49 03            CMP   POS+3
C2E1 D0 79               BNE   NOHEADON
C2E3 AD 4B 03            LDA   LATCH1
C2E6 AE 4A 03            LDX   LATCH2
C2E9 C9 0E               CMP   #14
C2EB F0 0C               BEQ   NCOLL
C2ED C9 07               CMP   #7
C2EF F0 0F               BEQ   ECOLL
C2F1 C9 0D               CMP   #13
C2F3 F0 12               BEQ   SCOLL
C2F5 C9 0B               CMP   #11
C2F7 F0 15               BEQ   WCOLL
C2F9 E0 0D     NCOLL     CPX   #13
C2FB F0 18               BEQ   HEADON
C2FD 4C 5C C3            JMP   NOHEADON
C300 E0 0B     ECOLL     CPX   #11
C302 F0 11               BEQ   HEADON
C304 4C 5C C3            JMP   NOHEADON
C307 E0 0E     SCOLL     CPX   #14
C309 F0 0A               BEQ   HEADON
C30B 4C 5C C3            JMP   NOHEADON
C30E E0 07     WCOLL     CPX   #7
C310 F0 03               BEQ   HEADON
C312 4C 5C C3            JMP   NOHEADON
C315 AD 3D 03  HEADON    LDA   SCORE1
C318 18                  CLC
C319 6D 42 03            ADC   STEPS
C31C 8D 3D 03            STA   SCORE1
C31F AD 3E 03            LDA   SCORE1+1
C322 69 00               ADC   #0
C324 8D 3E 03            STA   SCORE1+1
C327 AD 43 03            LDA   STEPS+1
C32A 18                  CLC
C32B 6D 3E 03            ADC   SCORE1+1
C32E 8D 3E 03            STA   SCORE1+1
C331 AE 44 03            LDX   LIVES1
C334 CA                  DEX
C335 8E 44 03            STX   LIVES1
C338 AD 3F 03            LDA   SCORE2
C33B 18                  CLC
C33C 6D 42 03            ADC   STEPS
C33F 8D 3F 03            STA   SCORE2
C342 AD 40 03            LDA   SCORE2+1
C345 69 00               ADC   #0
C347 8D 40 03            STA   SCORE2+1
C34A AD 43 03            LDA   STEPS+1
C34D 18                  CLC
C34E 6D 40 03            ADC   SCORE2+1
C351 8D 40 03            STA   SCORE2+1
C354 AE 45 03            LDX   LIVES2
C357 CA                  DEX
C358 8E 45 03            STX   LIVES2
C35B 60                  RTS
C35C AD 41 03  NOHEADON  LDA   PLAYER
C35F 0A                  ASL
C360 AA                  TAX
C361 BD 3D 03            LDA   SCORE1,X
C364 18                  CLC
C365 6D 42 03            ADC   STEPS
C368 9D 3D 03            STA   SCORE1,X
C36B BD 3E 03            LDA   SCORE1+1,X
```

```
C36E 69 00                ADC   #0
C370 9D 3E 03             STA   SCORE1+1,X
C373 AD 43 03             LDA   STEPS+1
C376 18                   CLC
C377 7D 3E 03             ADC   SCORE1+1,X
C37A 9D 3E 03             STA   SCORE1+1,X
C37D AE 41 03             LDX   PLAYER
C380 BD 44 03             LDA   LIVES1,X
C383 38                   SEC
C384 E9 01                SBC   #1
C386 9D 44 03             STA   LIVES1,X
C389 60                   RTS
;
; delay routine
;
C38A A0 00      DELAY     LDY   #0
C38C C8         LOOP10    INY
C38D D0 FD                BNE   LOOP10
C38F CA                   DEX
C390 D0 F8                BNE   DELAY
C392 60                   RTS
;
;move routine
;
C393 AE 41 03   MOVE      LDX   PLAYER
C396 BC 4C 03             LDY   SHAPE2,X
C399 8A                   TXA
C39A 0A                   ASL
C39B AA                   TAX
C39C B5 F7                LDA   NEXTPOS,X
C39E 9D 46 03             STA   POS,X
C3A1 B5 F8                LDA   NEXTPOS+1,X
C3A3 9D 47 03             STA   POS+1,X
C3A6 98                   TYA
C3A7 81 F7                STA   (NEXTPOS,X)
C3A9 B5 F8                LDA   NEXTPOS+1,X
C3AB 18                   CLC
C3AC 69 D4                ADC   #212
C3AE 95 F8                STA   NEXTPOS+1,X
C3B0 E0 02                CPX   #2
C3B2 D0 08                BNE   MOVERED
C3B4 A9 06                LDA   #BLUE
C3B6 81 F7                STA   (NEXTPOS,X)
C3B8 20 99 C5             JSR   BEEP
C3BB 60                   RTS
C3BC A9 02      MOVERED   LDA   #RED
C3BE 81 F7                STA   (NEXTPOS,X)
C3C0 20 99 C5             JSR   BEEP
C3C3 60                   RTS
;
; readjoy routine
;
C3C4 AE 41 03   READJOY   LDX   PLAYER
C3C7 BD 00 DC             LDA   JOY2,X
C3CA 29 0F                AND   #15
C3CC C9 0E      READIT    CMP   #14
C3CE F0 12                BEQ   NORTH
C3D0 C9 07                CMP   #7
C3D2 F0 26                BEQ   EAST
C3D4 C9 0D                CMP   #13
C3D6 F0 3A                BEQ   SOUTH
C3D8 C9 0B                CMP   #11
C3DA F0 4E                BEQ   WEST
C3DC BD 4A 03             LDA   LATCH2,X
C3DF 4C CC C3             JMP   READIT
C3E2 9D 4A 03   NORTH     STA   LATCH2,X
C3E5 8A                   TXA
C3E6 0A                   ASL
C3E7 AA                   TAX
C3E8 38                   SEC
C3E9 BD 46 03             LDA   POS,X
```

```
C3EC E9 28                SBC   #40
C3EE 95 F7                STA   NEXTPOS,X
C3F0 BD 47 03             LDA   POS+1,X
C3F3 E9 00                SBC   #0
C3F5 95 F8                STA   NEXTPOS+1,X
C3F7 4C 42 C4             JMP   JOYEXIT
C3FA 9D 4A 03   EAST      STA   LATCH2,X
C3FD 8A                   TXA
C3FE 0A                   ASL
C3FF AA                   TAX
C400 18                   CLC
C401 BD 46 03             LDA   POS,X
C404 69 01                ADC   #1
C406 95 F7                STA   NEXTPOS,X
C408 BD 47 03             LDA   POS+1,X
C40B 69 00                ADC   #0
C40D 95 F8                STA   NEXTPOS+1,X
C40F 4C 42 C4             JMP   JOYEXIT
C412 9D 4A 03   SOUTH     STA   LATCH2,X
C415 8A                   TXA
C416 0A                   ASL
C417 AA                   TAX
C418 18                   CLC
C419 BD 46 03             LDA   POS,X
C41C 69 28                ADC   #40
C41E 95 F7                STA   NEXTPOS,X
C420 BD 47 03             LDA   POS+1,X
C423 69 00                ADC   #0
C425 95 F8                STA   NEXTPOS+1,X
C427 4C 42 C4             JMP   JOYEXIT
C42A 9D 4A 03   WEST      STA   LATCH2,X
C42D 8A                   TXA
C42E 0A                   ASL
C42F AA                   TAX
C430 38                   SEC
C431 BD 46 03             LDA   POS,X
C434 E9 01                SBC   #1
C436 95 F7                STA   NEXTPOS,X
C438 BD 47 03             LDA   POS+1,X
C43B E9 00                SBC   #0
C43D 95 F8                STA   NEXTPOS+1,X
C43F 4C 42 C4             JMP   JOYEXIT
C442 60         JOYEXIT   RTS
;
; setup routine
;
C443 A2 00      SETUP     LDX   #0
C445 BD D4 C4   LOOP20    LDA   TABLE3,X
C448 F0 06                BEQ   NEXT21
C44A 9D FF 04             STA   1279,X
C44D E8                   INX
C44E D0 F5                BNE   LOOP20
C450 A2 00      NEXT21    LDX   #0
C452 BD E1 C4   LOOP21    LDA   TABLE4,X
C455 F0 06                BEQ   NEXT22
C457 9D 4F 05             STA   1359,X
C45A E8                   INX
C45B D0 F5                BNE   LOOP21
C45D A2 00      NEXT22    LDX   #0
C45F BD ED C4   LOOP22    LDA   TABLE5,X
C462 F0 06                BEQ   NEXT23
C464 9D 11 06             STA   1553,X
C467 E8                   INX
C468 D0 F5                BNE   LOOP22
C46A A2 00      NEXT23    LDX   #0
C46C BD ED C4   LOOP23    LDA   TABLE5,X
C46F F0 06                BEQ   NEXT24
C471 9D 21 06             STA   1569,X
C474 E8                   INX
C475 D0 F5                BNE   LOOP23
C477 A9 05      NEXT24    LDA   #GREEN
```

```
C479 A2 09              LDX   #9
C47B 9D F0 D8  LOOP24   STA   55536,X
C47E E8                 INX
C47F E0 1E              CPX   #30
C481 D0 F8              BNE   LOOP24
C483 A2 09              LDX   #9
C485 9D 40 D9  LOOP25   STA   55616,X
C488 E8                 INX
C489 E0 1E              CPX   #30
C48B D0 F8              BNE   LOOP25
C48D A2 01              LDX   #1
C48F 9D 08 DA  LOOP26   STA   55816,X
C492 E8                 INX
C493 E0 25              CPX   #37
C495 D0 F8              BNE   LOOP26
C497 18                 CLC
C498 A2 0D              LDX   #13
C49A A0 0F              LDY   #15
C49C 20 F0 FF           JSR   PLOT
C49F AE 45 03           LDX   LIVES2
C4A2 A9 00              LDA   #0
C4A4 20 CD BD           JSR   PRNUM
C4A7 18                 CLC
C4A8 A2 0D              LDX   #13
C4AA A0 1F              LDY   #31
C4AC 20 F0 FF           JSR   PLOT
C4AF AE 44 03           LDX   LIVES1
C4B2 A9 00              LDA   #0
C4B4 20 CD BD           JSR   PRNUM
C4B7 AD 01 DC  WAIT     LDA   JOY1
C4BA 2D 00 DC           AND   JOY2
C4BD 29 10              AND   #16
C4BF D0 F6              BNE   WAIT
C4C1 AD 01 DC  WAIT1    LDA   JOY1
C4C4 2D 00 DC           AND   JOY2
C4C7 29 10              AND   #16
C4C9 F0 F6              BEQ   WAIT1
C4CB A2 00              LDX   #0
C4CD 20 8A C3           JSR   DELAY
C4D0 20 BE C0           JSR   NEWSCREEN
C4D3 60                 RTS
C4D4 10 12 05  TABLE3   .BYT  16,18,5,19,19,32,2,2
                              1,20,20,15,14,0
C4E1 14 0F 20  TABLE4   .BYT  20,15,32,3,15,14,20,
                              9,14,21,5,0
C4ED 0C 09 16  TABLE5   .BYT  12,9,22,5,19,61,0
;
; endgame routine
;
C4F4 20 BE C0  ENDGAME  JSR   NEWSCREEN
C4F7 A2 00              LDX   #0
C4F9 BD 41 C5  LOOP30   LDA   TABLE6,X
C4FC F0 06              BEQ   NEXT31
C4FE 9D 00 05           STA   1280,X
C501 E8                 INX
C502 D0 F5              BNE   LOOP30
C504 A2 00     NEXT31   LDX   #0
C506 BD 4B C5  LOOP31   LDA   TABLE7,X
C509 F0 06              BEQ   NEXT32
C50B 9D 47 05           STA   1351,X
C50E E8                 INX
C50F D0 F5              BNE   LOOP31
C511 A9 05     NEXT32   LDA   #GREEN
C513 A2 01              LDX   #1
C515 9D F0 D8  LOOP32   STA   55536,X
C518 E8                 INX
C519 E0 1E              CPX   #30
C51B D0 F8              BNE   LOOP32
C51D A2 01              LDX   #1
C51F 9D 40 D9  LOOP33   STA   55616,X
C522 E8                 INX
```

```
C523 E0 25              CPX   #37
C525 D0 F8              BNE   LOOP33
C527 AD 01 DC  LOOP34   LDA   JOY1
C52A 2D 00 DC           AND   JOY2
C52D 29 10              AND   #16
C52F D0 F6              BNE   LOOP34
C531 AD 01 DC  LOOP35   LDA   JOY1
C534 2D 00 DC           AND   JOY2
C537 29 10              AND   #16
C539 F0 F6              BEQ   LOOP35
C53B 20 C1 C1           JSR   NEWGAME
C53E 4C 06 C0           JMP   MAINLOOP
C541 07 01 0D  TABLE6   .BYT  7,1,13,5,32,15,22,5,
                              18,0
C54B 10 12 05  TABLE7   .BYT  16,18,5,19,19,32,2,2
                              1,20,20,15,14,32,20
C559 0F 20 10           .BYT  15,32,16,12,1,25,32,
                              1,7,1,9,14,0
;
; crash routine
;
C566 A9 0F     CRASH    LDA   #15
C568 8D 18 D4           STA   VOLUME
C56B A9 81              LDA   #129
C56D 8D 04 D4           STA   WAVEFORM
C570 A9 09              LDA   #9
C572 8D 05 D4           STA   ATTACK
C575 A9 64              LDA   #100
C577 8D 00 D4           STA   V1LO
C57A A9 0C              LDA   #12
C57C 8D 01 D4           STA   V1HI
C57F A9 0F              LDA   #15
C581 8D 20 D0  BOOM     STA   BORDER
C584 A2 28              LDX   #40
C586 20 8A C3           JSR   DELAY
C589 38                 SEC
C58A E9 01              SBC   #1
C58C C9 00              CMP   #0
C58E D0 F1              BNE   BOOM
C590 A9 00              LDA   #0
C592 8D 04 D4           STA   WAVEFORM
C595 8D 05 D4           STA   ATTACK
C598 60                 RTS
;
; beep routine
;
C599 A9 08     BEEP     LDA   #8
C59B 8D 18 D4           STA   VOLUME
C59E A9 10              LDA    #16
C5A0 8D 05 D4           STA   ATTACK
C5A3 A9 80              LDA   #128
C5A5 8D 06 D4           STA   RELEASE
C5A8 A9 0A              LDA   #10
C5AA A2 00              LDX   #0
C5AC 18        BEEPSET  CLC
C5AD 6D 41 03           ADC   PLAYER
C5B0 E8                 INX
C5B1 E0 0A              CPX   #10
C5B3 D0 F7              BNE   BEEPSET
C5B5 8D 01 D4           STA   V1HI
C5B8 A9 25              LDA   #37
C5BA 8D 00 D4           STA   V1LO
C5BD A9 21              LDA   #33
C5BF 8D 04 D4           STA   WAVEFORM
C5C2 AE 3C 03           LDX   LEVEL
C5C5 20 8A C3           JSR   DELAY
C5C8 A9 00              LDA   #0
C5CA 8D 04 D4           STA   WAVEFORM
C5CD 8D 05 D4           STA   ATTACK
C5D0 8D 06 D4           STA   RELEASE
C5D3 60                 RTS
```

Nessie Source Code

Source code by Charles Brannon

```
C000                         *=     $C000
;
C000               BITS      =      7
C000               V         =      53248
C000               SPRITES   =      $3800
C000               INPTR     =      $FB
C000               OUTPTR    =      $FD
C000               SID       =      $D400
C000               RANDOM    =      $D41B
C000               SPREX     =      V
C000               SPRY      =      V+1
C000               SXMSB     =      V+16
C000               ENABLE    =      V+$15
C000               SEXY      =      V+$17
C000               MSPRITE   =      V+$1C
C000               SEXX      =      V+$1D
C000               SBCOLZ    =      V+$1F
C000               SSCOLZ    =      V+$1E
C000               VBORDER   =      V+$20
C000               VBACKGR   =      V+$21
C000               SPMC0     =      V+$25
C000               SPMC1     =      V+$26
C000               SPR0COL   =      V+$27
C000               STROUT    =      $AB1E
C000 A9 47         INIT      LDA    #<SHAPES
C002 85 FB                   STA    INPTR
C004 A9 C7                   LDA    #>SHAPES
C006 85 FC                   STA    INPTR+1
C008 A9 00                   LDA    #<SPRITES
C00A 85 FD                   STA    OUTPTR
C00C A9 38                   LDA    #>SPRITES
C00E 85 FE                   STA    OUTPTR+1
C010 A2 04                   LDX    #4
C012 A0 FF                   LDY    #255
C014 B1 FB         MOVSHAP   LDA    (INPTR),Y
C016 91 FD                   STA    (OUTPTR),Y
C018 88                      DEY
C019 C0 FF                   CPY    #255
C01B D0 F7                   BNE    MOVSHAP
C01D E6 FC                   INC    INPTR+1
C01F E6 FE                   INC    OUTPTR+1
C021 CA                      DEX
C022 30 07                   BMI    OUT
C024 D0 EE                   BNE    MOVSHAP
C026 A0 7F                   LDY    #127
C028 4C 14 C0                JMP    MOVSHAP
C02B C6 FE         OUT       DEC    OUTPTR+1
C02D A0 80                   LDY    #128
C02F A9 00                   LDA    #0
C031 91 FD         CLOOP     STA    (OUTPTR),Y
C033 C8                      INY
C034 D0 FB                   BNE    CLOOP
C036 A9 FF                   LDA    #255
C038 8D 0E D4                STA    $D40E
C03B 8D 0F D4                STA    $D40F
C03E A9 80                   LDA    #128
C040 8D 12 D4                STA    $D412
C043 A9 00                   LDA    #0
C045 8D 18 D4                STA    $D418
C048 A9 FC                   LDA    #252
```

```
C04A 8D 1B D0                STA    53275
C04D A2 05                   LDX    #5
C04F 20 BD C1      MAKEBEAST JSR    INITDIR
C052 A9 80                   LDA    #128
C054 9D CD CB                STA    XPOSL,X
C057 9D D3 CB                STA    YPOS,X
C05A A9 00                   LDA    #0
C05C 9D C7 CB                STA    XPOSH,X
C05F BD 18 C7                LDA    COLTAB+2,X
C062 9D 29 D0                STA    SPR0COL+2,X
C065 BD 20 C7                LDA    PTRS+2,X
C068 9D FA 07                STA    2042,X
C06B 8A                      TXA
C06C 0A                      ASL
C06D A8                      TAY
C06E A9 00                   LDA    #0
C070 99 05 D0                STA    SPRY+4,Y
C073 CA                      DEX
C074 10 D9                   BPL    MAKEBEAST
C076 AD 16 C7                LDA    COLTAB
C079 8D 27 D0                STA    SPR0COL
C07C AD 17 C7                LDA    COLTAB+1
C07F 8D 28 D0                STA    SPR0COL+1
C082 AD 1E C7                LDA    PTRS
C085 8D F8 07                STA    2040
C088 AD 1F C7                LDA    PTRS+1
C08B 8D F9 07                STA    2041
C08E A9 00                   LDA    #0
C090 8D EC CB                STA    CRXMSB
C093 A9 A0                   LDA    #160
C095 8D EB CB                STA    CROSSX
C098 8D 00 D0                STA    SPREX
C09B 8D 02 D0                STA    SPREX+2
C09E A9 80                   LDA    #128
C0A0 8D ED CB                STA    CROSSY
C0A3 8D 01 D0                STA    SPRY
C0A6 8D 03 D0                STA    SPRY+2
C0A9 A9 FF                   LDA    #255
C0AB 8D 15 D0                STA    ENABLE
C0AE 8D 1C D0                STA    MSPRITE
C0B1 A9 03                   LDA    #3
C0B3 8D 1D D0                STA    SEXX
C0B6 8D 17 D0                STA    SEXY
C0B9 A9 06                   LDA    #6
C0BB 8D 20 D0                STA    VBORDER
C0BE 8D 21 D0                STA    VBACKGR
C0C1 A9 05                   LDA    #5
C0C3 8D 25 D0                STA    SPMC0
C0C6 A9 04                   LDA    #4
C0C8 8D 26 D0                STA    SPMC1
C0CB A9 01                   LDA    #1
C0CD 8D F6 CB                STA    DUR
C0D0 20 E7 C1                JSR    INITGAME
C0D3 78                      SEI
C0D4 A9 76                   LDA    #<IRQ
C0D6 8D 14 03                STA    $314
C0D9 A9 C4                   LDA    #>IRQ
C0DB 8D 15 03                STA    $315
C0DE 58                      CLI
;
```

```
C0DF A0 00     MAIN      LDY  #0
C0E1 84 07               STY  BITS
C0E3 98        CREATURE  TYA
C0E4 0A                  ASL
C0E5 AA                  TAX
C0E6 B9 C7 CB            LDA  XPOSH,Y
C0E9 4A                  LSR
C0EA 66 07               ROR  BITS
C0EC B9 CD CB            LDA  XPOSL,Y
C0EF 9D 04 D0            STA  SPREX+4,X
C0F2 B9 D3 CB            LDA  YPOS,Y
C0F5 9D 05 D0            STA  SPRY+4,X
C0F8 C9 32               CMP  #50
C0FA 90 07               BCC  NOK1
C0FC C9 E5               CMP  #229
C0FE B0 03               BCS  NOK1
C100 4C 0E C1            JMP  CHECK2
C103 B9 E5 CB  NOK1      LDA  YDIR,Y
C106 49 FF               EOR  #255
C108 18                  CLC
C109 69 01               ADC  #1
C10B 99 E5 CB            STA  YDIR,Y
C10E B9 C7 CB  CHECK2    LDA  XPOSH,Y
C111 F0 0A               BEQ  CHECK3
C113 B9 CD CB            LDA  XPOSL,Y
C116 C9 40               CMP  #$40
C118 B0 0D               BCS  NOK2
C11A 4C 3E C1            JMP  OVERNOK
C11D B9 CD CB  CHECK3    LDA  XPOSL,Y
C120 C9 18               CMP  #$18
C122 90 03               BCC  NOK2
C124 4C 3E C1            JMP  OVERNOK
C127 A9 00     NOK2      LDA  #0
C129 99 DF CB            STA  XDIRH,Y
C12C B9 D9 CB            LDA  XDIR,Y
C12F 49 FF               EOR  #255
C131 18                  CLC
C132 69 01               ADC  #1
C134 99 D9 CB            STA  XDIR,Y
C137 C9 FF               CMP  #255
C139 D0 03               BNE  OVERNOK
C13B 99 DF CB            STA  XDIRH,Y
C13E B9 FA 07  OVERNOK   LDA  2042,Y
C141 49 01               EOR  #1
C143 99 FA 07            STA  2042,Y
C146 B9 D9 CB            LDA  XDIR,Y
C149 AA                  TAX
C14A B9 FA 07            LDA  2042,Y
C14D 09 02               ORA  #2
C14F 99 FA 07            STA  2042,Y
C152 E0 01               CPX  #1
C154 F0 05               BEQ  NOTURN
C156 49 02               EOR  #2
C158 99 FA 07            STA  2042,Y
C15B 18        NOTURN    CLC
C15C B9 CD CB            LDA  XPOSL,Y
C15F 79 D9 CB            ADC  XDIR,Y
C162 99 CD CB            STA  XPOSL,Y
C165 B9 C7 CB            LDA  XPOSH,Y
C168 79 DF CB            ADC  XDIRH,Y
C16B 99 C7 CB            STA  XPOSH,Y
C16E 18                  CLC
C16F B9 D3 CB            LDA  YPOS,Y
C172 79 E5 CB            ADC  YDIR,Y
C175 99 D3 CB            STA  YPOS,Y
C178 C8                  INY
C179 C0 06               CPY  #6
C17B F0 03               BEQ  DONEX
C17D 4C E3 C0            JMP  CREATURE
C180 AD 10 D0  DONEX     LDA  SXMSB
```

```
C183 29 03               AND  #3
C185 05 07               ORA  BITS
C187 8D 10 D0            STA  SXMSB
C18A AE F5 CB            LDX  SPEED
C18D F0 08               BEQ  NODELAY
C18F A0 00               LDY  #0
C191 88        DELAY     DEY
C192 D0 FD               BNE  DELAY
C194 CA                  DEX
C195 D0 FA               BNE  DELAY
C197 AD 1B D4  NODELAY   LDA  RANDOM
C19A C9 80               CMP  #128
C19C 90 0B               BCC  NOCHANGE
C19E AD 1B D4  ANOTHER   LDA  RANDOM
C1A1 C9 06               CMP  #6
C1A3 B0 F9               BCS  ANOTHER
C1A5 AA                  TAX
C1A6 20 BD C1            JSR  INITDIR
C1A9 20 E4 FF  NOCHANGE  JSR  $FFE4
C1AC F0 0C               BEQ  NOFREEZE
C1AE C9 88               CMP  #"{F7}"
C1B0 D0 03               BNE  FREEZE
C1B2 4C D0 C5            JMP  REGAME
C1B5 20 E4 FF  FREEZE    JSR  $FFE4
C1B8 F0 FB               BEQ  FREEZE
C1BA 4C DF C0  NOFREEZE  JMP  MAIN
C1BD A9 00     INITDIR   LDA  #0
C1BF 9D DF CB            STA  XDIRH,X
C1C2 AD 1B D4            LDA  RANDOM
C1C5 29 03               AND  #3
C1C7 A8                  TAY
C1C8 B9 3C C7            LDA  MOVY,Y
C1CB 9D D9 CB            STA  XDIR,X
C1CE C9 FF               CMP  #255
C1D0 D0 03               BNE  NOTNEG
C1D2 9D DF CB            STA  XDIRH,X
C1D5 AD 1B D4  NOTNEG    LDA  RANDOM
C1D8 29 03               AND  #3
C1DA A8                  TAY
C1DB B9 3C C7            LDA  MOVY,Y
C1DE 9D E5 CB            STA  YDIR,X
C1E1 1D D9 CB            ORA  XDIR,X
C1E4 F0 D7               BEQ  INITDIR
C1E6 60                  RTS
C1E7 A9 00     INITGAME  LDA  #0
C1E9 8D EF CB            STA  SCORE
C1EC 8D F0 CB            STA  SCORE+1
C1EF 8D F1 CB            STA  TIMER
C1F2 8D F2 CB            STA  TIMER+1
C1F5 8D F3 CB            STA  LENS
C1F8 8D 4C CC            STA  GAME
C1FB A9 14               LDA  #20
C1FD 8D F4 CB            STA  FILM
C200 A9 05               LDA  #5
C202 8D F5 CB            STA  SPEED
C205 A9 BF               LDA  #<TOPLIN
C207 A0 C2               LDY  #>TOPLIN
C209 20 1E AB            JSR  STROUT
C20C A9 F0               LDA  #<OPTMSG
C20E A0 C2               LDY  #>OPTMSG
C210 20 1E AB            JSR  STROUT
C213 20 E4 FF  GETAKEY   JSR  $FFE4
C216 F0 FB               BEQ  GETAKEY
C218 C9 85               CMP  #"{F1}"
C21A D0 12               BNE  NOTF1
C21C A9 E0               LDA  #$E0
C21E 8D F8 07            STA  2040
C221 A9 E1               LDA  #$E1
C223 8D F9 07            STA  2041
C226 A9 00               LDA  #0
```

```
C228 8D F3 CB            STA   LENS
C22B 4C 13 C2            JMP   GETAKEY
C22E C9 86     NOTF1     CMP   #"{F3}"
C230 D0 12               BNE   NOTF3
C232 A9 E2               LDA   #$E2
C234 8D F8 07            STA   2040
C237 A9 E3               LDA   #$E3
C239 8D F9 07            STA   2041
C23C A9 01               LDA   #1
C23E 8D F3 CB            STA   LENS
C241 4C 13 C2            JMP   GETAKEY
C244 C9 87     NOTF3     CMP   #"{F5}"
C246 D0 1E               BNE   NOTF5
C248 EE F5 CB            INC   SPEED
C24B AD F5 CB            LDA   SPEED
C24E 10 05               BPL   NOTNINE
C250 A9 09               LDA   #9
C252 8D F5 CB            STA   SPEED
C255 C9 0A     NOTNINE   CMP   #10
C257 90 05               BCC   OVERIT
C259 A9 00               LDA   #0
C25B 8D F5 CB            STA   SPEED
C25E 09 30     OVERIT    ORA   #48
C260 8D 06 04            STA   1030
C263 4C 13 C2            JMP   GETAKEY
C266 C9 88     NOTF5     CMP   #"{F7}"
C268 F0 33               BEQ   GOGAME
C26A C9 8C               CMP   #"{F8}"
C26C D0 A5               BNE   GETAKEY
C26E EE 4C CC            INC   GAME
C271 20 9D C2            JSR   GOGAME
C274 A2 32               LDX   #50
C276 A0 28               LDY   #40
C278 AD 1B D4  NOGOOD    LDA   RANDOM
C27B C9 A7               CMP   #167
C27D B0 F9               BCS   NOGOOD
C27F 85 FD               STA   OUTPTR
C281 AD 1B D4            LDA   RANDOM
C284 29 03               AND   #3
C286 18                  CLC
C287 69 04               ADC   #4
C289 85 FE               STA   OUTPTR+1
C28B A9 A0               LDA   #160
C28D 91 FD               STA   (OUTPTR),Y
C28F A5 FE               LDA   OUTPTR+1
C291 69 D4               ADC   #$D4
C293 85 FE               STA   OUTPTR+1
C295 A9 00               LDA   #0
C297 91 FD               STA   (OUTPTR),Y
C299 CA                  DEX
C29A D0 DC               BNE   NOGOOD
C29C 60                  RTS
C29D A9 BF     GOGAME    LDA   #<TOPLIN
C29F A0 C2               LDY   #>TOPLIN
C2A1 20 1E AB            JSR   STROUT
C2A4 AD F5 CB            LDA   SPEED
C2A7 09 30               ORA   #48
C2A9 8D 06 04            STA   1030
C2AC 38                  SEC
C2AD A9 09               LDA   #9
C2AF ED F5 CB            SBC   SPEED
C2B2 8D F5 CB            STA   SPEED
C2B5 0E F5 CB            ASL   SPEED
C2B8 0E F5 CB            ASL   SPEED
C2BB 0E F5 CB            ASL   SPEED
C2BE 60                  RTS
C2BF 93 92 0E  TOPLIN    .ASC  "{CLR}{OFF}{N}{
                               CYN}Speed:{YEL}
                               5 [<7>]Nessie [<8
                               >]Film Left:{YEL}20
                               "
```

```
C2EE 0D 00               .BYT  13,0
C2F0 99 12 46  OPTMSG    .ASC  " [<6>]{RVS}F1{O
                               FF}:TELEPHOTO {R
                               VS}F3{OFF}:Wid
                               e Angle"
C30F 20 12 46            .ASC  " {RVS}F5{OFF}:S
                               PEED"
C31A 0D                  .BYT  13
C31B 12 46 37            .ASC  "{RVS}F7{OFF}:EA
                               SY GAME {RVS}F8
                               {OFF}:Hard Game"
C338 00                  .BYT  0
C339 AD 00 DC  JOYMOVE   LDA   56320
C33C 48                  PHA
C33D 29 10               AND   #16
C33F 8D EE CB            STA   FIRE
C342 68                  PLA
C343 29 0F               AND   #15
C345 49 0F               EOR   #15
C347 AA                  TAX
C348 18                  CLC
C349 AD EB CB            LDA   CROSSX
C34C 7D 31 C7            ADC   LMOVX,X
C34F 8D EB CB            STA   CROSSX
C352 8D 00 D0            STA   SPREX
C355 8D 02 D0            STA   SPREX+2
C358 AD EC CB            LDA   CRXMSB
C35B 7D 26 C7            ADC   HMOVX,X
C35E 8D EC CB            STA   CRXMSB
C361 4A                  LSR
C362 08                  PHP
C363 AD 10 D0            LDA   SXMSB
C366 4A                  LSR
C367 4A                  LSR
C368 28                  PLP
C369 08                  PHP
C36A 2A                  ROL
C36B 28                  PLP
C36C 2A                  ROL
C36D 8D 10 D0            STA   SXMSB
C370 18                  CLC
C371 AD ED CB            LDA   CROSSY
C374 7D 3C C7            ADC   MOVY,X
C377 8D ED CB            STA   CROSSY
C37A 8D 01 D0            STA   SPRY
C37D 8D 03 D0            STA   SPRY+2
C380 AD EE CB            LDA   FIRE
C383 F0 0D               BEQ   FRAME
C385 A9 00               LDA   #0
C387 8D F7 CB            STA   YESFIRE
C38A A9 01               LDA   #1
C38C 8D F6 CB            STA   DUR
C38F 4C A0 C3            JMP   EXPIRED
C392 AD F7 CB  FRAME     LDA   YESFIRE
C395 F0 17               BEQ   DOFR
C397 AD F6 CB            LDA   DUR
C39A F0 04               BEQ   EXPIRED
C39C CE F6 CB            DEC   DUR
C39F 60                  RTS
C3A0 A9 06     EXPIRED   LDA   #6
C3A2 8D 20 D0            STA   VBORDER
C3A5 8D 21 D0            STA   VBACKGR
C3A8 A9 80               LDA   #128
C3AA 8D 04 D4            STA   SID+4
C3AD 60                  RTS
C3AE CE 26 04  DOFR      DEC   1062
C3B1 AD 26 04            LDA   1062
C3B4 C9 30               CMP   #"0"
C3B6 B0 14               BCS   OK
```

```
C3B8 A9 39                LDA   #"9"
C3BA 8D 26 04             STA   1062
C3BD CE 25 04             DEC   1061
C3C0 AD 25 04             LDA   1061
C3C3 C9 31                CMP   #"1"
C3C5 B0 05                BCS   OK
C3C7 A9 20                LDA   #32
C3C9 8D 25 04             STA   1061
C3CC A9 1F      OK        LDA   #$1F
C3CE 8D 05 D4             STA   SID+5
C3D1 8D 18 D4             STA   SID+24
C3D4 A9 03                LDA   #3
C3D6 8D 06 D4             STA   SID+6
C3D9 A9 FF                LDA   #255
C3DB 8D 00 D4             STA   SID
C3DE 8D 01 D4             STA   SID+1
C3E1 8D 04 D4             STA   SID+4
C3E4 A9 04                LDA   #4
C3E6 8D 4A CC             STA   MASK
C3E9 A2 06                LDX   #6
C3EB AC F4 CB             LDY   FILM
C3EE 88                   DEY
C3EF A9 00                LDA   #0
C3F1 99 20 CC             STA   BORDER,Y
C3F4 AD 4A CC   BCOLZ     LDA   MASK
C3F7 09 02                ORA   #2
C3F9 8D 15 D0             STA   ENABLE
C3FC AD 1E D0             LDA   SSCOLZ
C3FF 20 7F C4             JSR   WAIT60
C402 AD 1E D0             LDA   SSCOLZ
C405 29 FD                AND   #253
C407 19 20 CC             ORA   BORDER,Y
C40A 99 20 CC             STA   BORDER,Y
C40D 0E 4A CC             ASL   MASK
C410 CA                   DEX
C411 D0 E1                BNE   BCOLZ
C413 A9 01                LDA   #1
C415 8D 20 D0             STA   VBORDER
C418 8D 21 D0             STA   VBACKGR
C41B 8D F7 CB             STA   YESFIRE
C41E A9 04                LDA   #4
C420 8D 4A CC             STA   MASK
C423 A2 06                LDX   #6
C425 A9 00                LDA   #0
C427 99 34 CC             STA   HAIR,Y
C42A AD 4A CC   HCOLZ     LDA   MASK
C42D 09 01                ORA   #1
C42F 8D 15 D0             STA   ENABLE
C432 AD 1E D0             LDA   SSCOLZ
C435 20 7F C4             JSR   WAIT60
C438 AD 1E D0             LDA   SSCOLZ
C43B 29 FE                AND   #254
C43D 19 34 CC             ORA   HAIR,Y
C440 99 34 CC             STA   HAIR,Y
C443 0E 4A CC             ASL   MASK
C446 CA                   DEX
C447 D0 E1                BNE   HCOLZ
C449 AD 1F D0             LDA   SBCOLZ
C44C 20 7F C4             JSR   WAIT60
C44F AD 1F D0             LDA   SBCOLZ
C452 99 F8 CB             STA   GARBAGE,Y
C455 AD 00 DC             LDA   56320
C458 29 0F                AND   #15
C45A 49 0F                EOR   #15
C45C 99 0C CC             STA   BLUR,Y
C45F A9 FF                LDA   #255
C461 8D 15 D0             STA   ENABLE
C464 CE F4 CB             DEC   FILM
C467 AD F4 CB             LDA   FILM
C46A F0 26                BEQ   GAMEOVER
```

```
C46C C9 05                CMP   #5
C46E D0 05                BNE   NOYEL
C470 A9 07                LDA   #7
C472 8D 28 D0             STA   SPROCOL+1
C475 60         NOYEL     RTS
C476 20 89 C4   IRQ       JSR   INCTIM
C479 20 39 C3             JSR   JOYMOVE
C47C 4C 31 EA             JMP   $EA31
C47F 20 82 C4   WAIT60    JSR   WAIT
C482 AD 12 D0   WAIT      LDA   $D012
C485 C9 FF                CMP   #255
C487 D0 F9                BNE   WAIT
C489 EE F1 CB   INCTIM    INC   TIMER
C48C D0 03                BNE   NOINC
C48E EE F2 CB             INC   TIMER+1
C491 60         NOINC     RTS
C492 A9 93      GAMEOVER  LDA   #"{CLR}"
C494 20 D2 FF             JSR   $FFD2
C497 A9 00                LDA   #0
C499 8D 20 D0             STA   VBORDER
C49C 8D 21 D0             STA   VBACKGR
C49F 78                   SEI
C4A0 A9 0F                LDA   #<RASTER
C4A2 8D 14 03             STA   $314
C4A5 A9 C6                LDA   #>RASTER
C4A7 8D 15 03             STA   $315
C4AA A9 01                LDA   #1
C4AC 8D 1A D0             STA   $D01A
C4AF A9 1B                LDA   #27
C4B1 8D 11 D0             STA   $D011
C4B4 A9 32                LDA   #50
C4B6 8D 12 D0             STA   $D012
C4B9 A9 7F                LDA   #127
C4BB 8D 0D DC             STA   $DC0D
C4BE A9 FF                LDA   #255
C4C0 8D 1B D0             STA   53275
C4C3 A9 00                LDA   #0
C4C5 8D 4B CC             STA   WHICH
C4C8 A9 1F                LDA   #31
C4CA 8D 15 D0             STA   ENABLE
C4CD A9 FF                LDA   #255
C4CF 8D 1D D0             STA   SEXX
C4D2 8D 17 D0             STA   SEXY
C4D5 A9 10                LDA   #16
C4D7 8D 10 D0             STA   SXMSB
C4DA A2 00                LDX   #0
C4DC 8A         INITIX    TXA
C4DD 0A                   ASL
C4DE A8                   TAY
C4DF 0A                   ASL
C4E0 0A                   ASL
C4E1 0A                   ASL
C4E2 0A                   ASL
C4E3 0A                   ASL
C4E4 18                   CLC
C4E5 69 1D                ADC   #29
C4E7 99 00 D0             STA   SPREX,Y
C4EA A9 09                LDA   #9
C4EC 9D 27 D0             STA   SPROCOL,X
C4EF E8                   INX
C4F0 E0 05                CPX   #5
C4F2 D0 E8                BNE   INITIX
C4F4 58                   CLI
C4F5 A2 13                LDX   #19
C4F7 A0 00                LDY   #0
C4F9 BD F8 CB   TALLY     LDA   GARBAGE,X
C4FC 29 03                AND   #3
C4FE 1D 0C CC             ORA   BLUR,X
C501 F0 0A                BEQ   NOTBLUR
C503 A9 F1                LDA   #$F1
```

```
C505 99 65 C6             STA   WHAT,Y
C508 A9 00                LDA   #0
C50A 4C 56 C5             JMP   XXX
C50D BD 34 CC  NOTBLUR    LDA   HAIR,X
C510 29 40                AND   #64
C512 D0 2C                BNE   NESSIE
C514 BD 34 CC             LDA   HAIR,X
C517 D0 0A                BNE   NOTBLANK
C519 A9 F2                LDA   #$F2
C51B 99 65 C6             STA   WHAT,Y
C51E A9 00                LDA   #0
C520 4C 56 C5             JMP   XXX
C523 8E 48 CC  NOTBLANK   STX   TEMP
C526 BD 34 CC             LDA   HAIR,X
C529 A2 00                LDX   #0
C52B 4A        BLOOP      LSR
C52C B0 04                BCS   FBIT
C52E E8                   INX
C52F 4C 2B C5             JMP   BLOOP
C532 BD 1E C7  FBIT       LDA   PTRS,X
C535 99 65 C6             STA   WHAT,Y
C538 AE 48 CC             LDX   TEMP
C53B A9 05                LDA   #5
C53D 4C 56 C5             JMP   XXX
C540 BD 20 CC  NESSIE     LDA   BORDER,X
C543 F0 0A                BEQ   TRUENESS
C545 A9 F0                LDA   #$F0
C547 99 65 C6             STA   WHAT,Y
C54A A9 0A                LDA   #10
C54C 4C 56 C5             JMP   XXX
C54F A9 EC     TRUENESS   LDA   #$EC
C551 99 65 C6             STA   WHAT,Y
C554 A9 C8                LDA   #200
C556 18        XXX        CLC
C557 6D EF CB             ADC   SCORE
C55A 8D EF CB             STA   SCORE
C55D AD F0 CB             LDA   SCORE+1
C560 69 00                ADC   #0
C562 8D F0 CB             STA   SCORE+1
C565 CA                   DEX
C566 C8                   INY
C567 C0 14                CPY   #20
C569 D0 8E                BNE   TALLY
C56B 20 79 C6             JSR   BOXES
C56E AD EF CB             LDA   SCORE
C571 0D F0 CB             ORA   SCORE+1
C574 F0 1D                BEQ   PRSCORE
C576 38                   SEC
C577 AD EF CB             LDA   SCORE
C57A ED F2 CB             SBC   TIMER+1
C57D 8D EF CB             STA   SCORE
C580 AD F0 CB             LDA   SCORE+1
C583 E9 00                SBC   #0
C585 8D F0 CB             STA   SCORE+1
C588 AD 4C CC             LDA   GAME
C58B F0 06                BEQ   PRSCORE
C58D 0E EF CB             ASL   SCORE
C590 2E F0 CB             ROL   SCORE+1
C593 A2 18     PRSCORE    LDX   #24
C595 A0 00                LDY   #0
C597 18                   CLC
C598 20 F0 FF             JSR   $FFF0
C59B A9 EC                LDA   #<SCORWORD
C59D A0 C5                LDY   #>SCORWORD
C59F 20 1E AB             JSR   STROUT
C5A2 AE EF CB             LDX   SCORE
C5A5 AD F0 CB             LDA   SCORE+1
C5A8 20 CD BD             JSR   $BDCD
C5AB A9 30                LDA   #"0"
C5AD 20 D2 FF             JSR   $FFD2
```

```
C5B0 AD F3 CB             LDA   LENS
C5B3 D0 05                BNE   FB0
C5B5 A9 30                LDA   #"0"
C5B7 20 D2 FF             JSR   $FFD2
C5BA A2 78     FB0        LDX   #120
C5BC 20 7F C4  DELAY2     JSR   WAIT60
C5BF CA                   DEX
C5C0 D0 FA                BNE   DELAY2
C5C2 AD 00 DC  FB1        LDA   56320
C5C5 29 10                AND   #16
C5C7 F0 F9                BEQ   FB1
C5C9 AD 00 DC  FB2        LDA   56320
C5CC 29 10                AND   #16
C5CE D0 F9                BNE   FB2
C5D0 78        REGAME     SEI
C5D1 A9 00                LDA   #0
C5D3 8D 1A D0             STA   $D01A
C5D6 A9 FF                LDA   #255
C5D8 8D 0D DC             STA   $DC0D
C5DB A9 31                LDA   #$31
C5DD 8D 14 03             STA   $314
C5E0 A9 EA                LDA   #$EA
C5E2 8D 15 03             STA   $315
C5E5 58                   CLI
C5E6 A2 FF                LDX   #255
C5E8 9A                   TXS
C5E9 4C 00 C0             JMP   INIT
C5EC 9F D0 4C  SCORWORD   .ASC  "{CYN}PLAY AGAIN
                                :SNAP SHUTTER [<
                                7>]SCORE:{YEL}{R
                                VS}"
C60E 00                   .BYT  0
C60F AD 12 D0  RASTER     LDA   $D012
C612 18                   CLC
C613 69 02                ADC   #2
C615 8D 01 D0             STA   SPRY
C618 8D 03 D0             STA   SPRY+2
C61B 8D 05 D0             STA   SPRY+4
C61E 8D 07 D0             STA   SPRY+6
C621 8D 09 D0             STA   SPRY+8
C624 8D 0B D0             STA   SPRY+10
C627 AE 4B CC             LDX   WHICH
C62A BD 61 C6             LDA   LAST,X
C62D A8                   TAY
C62E A2 00                LDX   #0
C630 B9 65 C6  WLOOP      LDA   WHAT,Y
C633 9D F8 07             STA   2040,X
C636 C8                   INY
C637 E8                   INX
C638 E0 05                CPX   #5
C63A D0 F4                BNE   WLOOP
C63C EE 4B CC             INC   WHICH
C63F AD 4B CC             LDA   WHICH
C642 C9 04                CMP   #4
C644 D0 05                BNE   RX
C646 A9 00                LDA   #0
C648 8D 4B CC             STA   WHICH
C64B AA        RX         TAX
C64C BD 5D C6             LDA   VPOS,X
C64F 8D 12 D0             STA   $D012
C652 A9 01     REXIT      LDA   #1
C654 8D 19 D0             STA   $D019
C657 68                   PLA
C658 A8                   TAY
C659 68                   PLA
C65A AA                   TAX
C65B 68                   PLA
C65C 40                   RTI
C65D 31 61 92  VPOS       .BYT  49,97,146,194
C661 00 05 0A  LAST       .BYT  0,5,10,15
```

```
C665 E0 E1 E2  WHAT      .BYT   $E0,$E1,$E2,$E3,$E
                                4
C66A E4 E3 E1            .BYT   $E4,$E3,$E1,$E2,$E
                                4
C66F E1 E3 E2            .BYT   $E1,$E3,$E2,$E4,$E
                                0
C674 E1 E3 E1            .BYT   $E1,$E3,$E1,$E0,$E
                                5
;
C679 A2 00     BOXES     LDX    #0
C67B A9 07               LDA    #7
C67D 9D 00 D8  CFILL     STA    $D800,X
C680 9D 00 D9            STA    $D900,X
C683 9D 00 DA            STA    $DA00,X
C686 9D 00 DB            STA    $DB00,X
C689 E8                  INX
C68A D0 F1               BNE    CFILL
C68C A9 00               LDA    #0
C68E 85 FD               STA    OUTPTR
C690 A9 04               LDA    #4
C692 8D 49 CC            STA    TEMP+1
C695 85 FE               STA    OUTPTR+1
C697 A9 05     BOXROW    LDA    #5
C699 8D 48 CC            STA    TEMP
C69C A0 00               LDY    #0
C69E A9 70     TOPBOX    LDA    #112
C6A0 91 FD               STA    (OUTPTR),Y
C6A2 C8                  INY
C6A3 A2 06               LDX    #6
C6A5 A9 40               LDA    #64
C6A7 91 FD     BAR1      STA    (OUTPTR),Y
C6A9 C8                  INY
C6AA CA                  DEX
C6AB D0 FA               BNE    BAR1
C6AD A9 6E               LDA    #110
C6AF 91 FD               STA    (OUTPTR),Y
C6B1 C8                  INY
C6B2 CE 48 CC            DEC    TEMP
C6B5 AD 48 CC            LDA    TEMP
C6B8 D0 E4               BNE    TOPBOX
C6BA A2 04               LDX    #4
C6BC 20 08 C7  SIDES     JSR    L40
C6BF A0 00               LDY    #0
C6C1 18                  CLC
C6C2 A9 5D     SIDE      LDA    #93
C6C4 91 FD               STA    (OUTPTR),Y
C6C6 98                  TYA
C6C7 69 07               ADC    #7
C6C9 A8                  TAY
C6CA A9 5D               LDA    #93
C6CC 91 FD               STA    (OUTPTR),Y
C6CE C8                  INY
C6CF C0 28               CPY    #40
C6D1 90 EF               BCC    SIDE
C6D3 CA                  DEX
C6D4 D0 E6               BNE    SIDES
C6D6 20 08 C7            JSR    L40
C6D9 A9 05               LDA    #5
C6DB 8D 48 CC            STA    TEMP
C6DE A0 00               LDY    #0
C6E0 A9 6D     BOTBOX    LDA    #109
C6E2 91 FD               STA    (OUTPTR),Y
C6E4 C8                  INY
C6E5 A2 06               LDX    #6
C6E7 A9 40               LDA    #64
C6E9 91 FD     BAR2      STA    (OUTPTR),Y
C6EB C8                  INY
C6EC CA                  DEX
C6ED D0 FA               BNE    BAR2
C6EF A9 7D               LDA    #125
```

```
C6F1 91 FD               STA    (OUTPTR),Y
C6F3 C8                  INY
C6F4 CE 48 CC            DEC    TEMP
C6F7 AD 48 CC            LDA    TEMP
C6FA D0 E4               BNE    BOTBOX
C6FC 20 08 C7            JSR    L40
C6FF CE 49 CC            DEC    TEMP+1
C702 AD 49 CC            LDA    TEMP+1
C705 D0 90               BNE    BOXROW
C707 60                  RTS
C708 A5 FD     L40       LDA    OUTPTR
C70A 18                  CLC
C70B 69 28               ADC    #40
C70D 85 FD               STA    OUTPTR
C70F A5 FE               LDA    OUTPTR+1
C711 69 00               ADC    #0
C713 85 FE               STA    OUTPTR+1
C715 60                  RTS
C716 0A 0E 09  COLTAB    .BYT   10,14,9,9,1,1,7,9
C71E E0 E1 E4  PTRS      .BYT   $E0,$E1,$E4,$E4,$E
                                8,$E8,$EC,$E4
C726 00 00 00  HMOVX     .BYT   0,0,0,0,255,255,255,
                                0,0,0,0
C731 00 00 00  LMOVX     .BYT   0,0,0,0,255,255,255,
                                0,1,1,1
C73C 00 FF 01  MOVY      .BYT   0,255,1,0,0,255,1,0,
                                0,255,1
;
C747 00 00 00  SHAPES    .BYT   0,0,0,0,0,0,0,0
C74F 00 00 80            .BYT   0,0,128,0,0,128,0,2
C757 A0 00 00            .BYT   160,0,0,128,0,0,128,
                                0
C75F 00 00 00            .BYT   0,0,0,0,0,0,0,0
C767 00 00 00            .BYT   0,0,0,0,0,0,0,0
C76F 00 00 00            .BYT   0,0,0,0,0,0,0,0
C777 00 00 00            .BYT   0,0,0,0,0,0,0,0
C77F 00 00 00            .BYT   0,0,0,0,0,0,0,147
C787 AA AA 80            .BYT   170,170,128,128,0,1
                                28,128,0
C78F 80 80 00            .BYT   128,128,0,128,128,0
                                ,128,128
C797 00 80 80            .BYT   0,128,128,0,128,128
                                ,0,128
C79F 80 00 80            .BYT   128,0,128,128,0,128
                                ,128,0
C7A7 80 AA AA            .BYT   128,170,170,128,0,0
                                ,0,0
C7AF 00 00 00            .BYT   0,0,0,0,0,0,0,0
C7B7 00 00 00            .BYT   0,0,0,0,0,0,0,0
C7BF 00 00 00            .BYT   0,0,0,0,0,0,0,147
C7C7 00 00 00            .BYT   0,0,0,0,0,0,0,0
C7CF 00 00 00            .BYT   0,0,0,0,0,0,0,0
C7D7 08 00 00            .BYT   008,0,0,8,0,0,8,0
C7DF 00 AA 80            .BYT   0,170,128,0,8,0,0,8
C7E7 00 00 08            .BYT   0,0,8,0,0,0,0,0
C7EF 00 00 00            .BYT   0,0,0,0,0,0,0,0
C7F7 00 00 00            .BYT   0,0,0,0,0,0,0,0
C7FF 00 00 00            .BYT   0,0,0,0,0,0,0,48
C807 2A AA AA            .BYT   042,170,170,32,0,2,
                                32,0
C80F 02 20 00            .BYT   002,32,0,2,32,0,2,32
C817 00 02 20            .BYT   0,2,32,0,2,32,0,2
C81F 20 00 02            .BYT   032,0,2,32,0,2,32,0
C827 02 20 00            .BYT   002,32,0,2,32,0,2,32
C82F 00 02 20            .BYT   0,2,32,0,2,32,0,2
C837 2A AA AA            .BYT   042,170,170,0,0,0,0,
                                0
C83F 00 00 00            .BYT   0,0,0,0,0,0,0,48
C847 00 00 00            .BYT   0,0,0,0,0,0,0,0
C84F 00 00 00            .BYT   0,0,0,0,0,0,0,0
```

```
C857 00 00 00           .BYT    0,0,0,0,0,0,0,0
C85F 00 00 00           .BYT    0,0,0,80,170,0,118,1
                                70
C867 A0 56 AE           .BYT    160,86,174,168,22,2
                                34,186,6
C86F AE AA 0B           .BYT    174,170,11,170,233,
                                10,170,149
C877 05 40 14           .BYT    005,64,20,1,0,80,0,6
                                4
C87F 40 00 00           .BYT    064,0,0,0,0,0,0,233
C887 00 00 00           .BYT    0,0,0,0,0,0,0,0
C88F 00 00 00           .BYT    0,0,0,0,0,0,0,0
C897 00 00 00           .BYT    0,0,0,0,0,0,0,0
C89F 00 00 00           .BYT    0,0,0,0,170,0,82,170
C8A7 A0 76 AE           .BYT    160,118,174,168,86,
                                234,186,22
C8AF AE AA 0B           .BYT    174,170,11,170,233,
                                10,170,148
C8B7 01 40 14           .BYT    001,64,20,0,80,5,0,1
                                6
C8BF 00 00 00           .BYT    0,0,0,0,0,0,0,233
C8C7 00 00 00           .BYT    0,0,0,0,0,0,0,0
C8CF 00 00 00           .BYT    0,0,0,0,0,0,0,0
C8D7 00 00 00           .BYT    0,0,0,0,0,0,0,0
C8DF 00 00 00           .BYT    0,0,0,0,170,5,10,170
C8E7 9D 2A BA           .BYT    157,42,186,149,174,
                                171,148,170
C8EF BA 90 6B           .BYT    186,144,107,170,22
                                4,86,170,160
C8F7 14 01 50           .BYT    020,1,80,5,0,64,1,1
C8FF 00 00 00           .BYT    0,0,0,0,0,0,0,233
C907 00 00 00           .BYT    0,0,0,0,0,0,0,0
C90F 00 00 00           .BYT    0,0,0,0,0,0,0,0
C917 00 00 00           .BYT    0,0,0,0,0,0,0,0
C91F 00 00 00           .BYT    0,0,0,0,170,0,10,170
C927 85 2A BA           .BYT    133,42,186,157,174,
                                171,149,170
C92F BA 94 6B           .BYT    186,148,107,170,22
                                4,22,170,160
C937 14 01 40           .BYT    020,1,64,80,5,0,0,4
C93F 00 00 00           .BYT    0,0,0,0,0,0,0,22
C947 00 00 00           .BYT    0,0,0,0,0,0,0,0
C94F 00 00 00           .BYT    0,0,0,0,0,0,0,16
C957 00 00 74           .BYT    0,0,116,0,0,20,0,0
C95F 05 00 00           .BYT    005,0,0,1,0,0,1,68
C967 40 01 95           .BYT    064,1,149,144,1,89,
                                80,0
C96F 44 64 00           .BYT    068,100,0,0,20,0,0,5
C977 00 00 00           .BYT    0,0,0,0,0,0,0,0
C97F 00 00 00           .BYT    0,0,0,0,0,0,0,0
C987 00 00 00           .BYT    0,0,0,0,0,0,0,0
C98F 00 00 00           .BYT    0,0,0,0,0,0,0,0
C997 00 00 00           .BYT    0,0,0,0,0,0,0,0
C99F 00 00 00           .BYT    0,0,0,16,0,0,116,145
C9A7 01 15 56           .BYT    001,21,86,85,9,101,
                                148,0
C9AF 11 40 00           .BYT    017,64,0,0,0,0,0,0
C9B7 00 00 00           .BYT    0,0,0,0,0,0,0,0
C9BF 00 00 00           .BYT    0,0,0,0,0,0,0,0
C9C7 00 00 00           .BYT    0,0,0,0,0,0,0,0
C9CF 00 00 00           .BYT    0,0,0,0,0,0,0,0
C9D7 00 04 00           .BYT    0,4,0,0,29,0,0,20
C9DF 00 00 50           .BYT    0,0,80,0,0,64,1,17
C9E7 80 05 95           .BYT    128,5,149,64,9,89,6
                                4,21
C9EF 11 00 24           .BYT    017,0,36,0,0,80,0,0
C9F7 00 00 00           .BYT    0,0,0,0,0,0,0,0
C9FF 00 00 00           .BYT    0,0,0,0,0,0,0,0
CA07 00 00 00           .BYT    0,0,0,0,0,0,0,0
CA0F 00 00 00           .BYT    0,0,0,0,0,0,0,0
```

```
CA17 00 00 00           .BYT    0,0,0,0,0,0,0,0
CA1F 00 00 00           .BYT    0,0,0,0,0,4,64,69
CA27 1D 65 65           .BYT    029,101,101,84,22,8
                                6,64,1
CA2F 44 00 00           .BYT    068,0,0,0,0,0,0,0
CA37 00 00 00           .BYT    0,0,0,0,0,0,0,0
CA3F 00 00 00           .BYT    0,0,0,0,0,0,0,0
CA47 00 00 00           .BYT    0,0,0,0,0,0,0,0
CA4F 00 00 00           .BYT    0,0,0,0,0,0,0,20
CA57 00 00 5D           .BYT    0,0,93,0,0,37,0,0
CA5F 05 00 00           .BYT    005,0,0,5,64,0,5,68
CA67 40 01 55           .BYT    064,1,85,80,2,86,80,
                                5
CA6F 65 54 05           .BYT    101,84,5,85,149,1,8
                                5,80
CA77 01 40 50           .BYT    001,64,80,0,80,20,0,
                                0
CA7F 00 00 00           .BYT    0,0,0,0,0,0,0,255
CA87 00 00 00           .BYT    0,0,0,0,0,0,0,0
CA8F 00 00 00           .BYT    0,0,0,0,20,0,0,93
CA97 00 00 25           .BYT    0,0,37,0,0,5,0,0
CA9F 05 00 00           .BYT    005,0,0,5,64,0,1,68
CAA7 40 01 55           .BYT    064,1,85,80,2,86,84,
                                5
CAAF 65 55 05           .BYT    101,85,5,85,144,1,8
                                5,80
CAB7 01 01 40           .BYT    001,1,64,1,1,64,0,64
CABF 50 00 00           .BYT    080,0,0,0,0,0,0,255
CAC7 00 00 00           .BYT    0,0,0,0,0,0,0,0
CACF 00 00 00           .BYT    0,0,0,0,0,0,0,0
CAD7 00 14 00           .BYT    0,20,0,0,117,0,0,88
CADF 00 00 50           .BYT    0,0,80,0,1,80,1,17
CAE7 50 05 55           .BYT    080,5,85,64,5,149,1
                                28,21
CAEF 59 50 56           .BYT    089,80,86,85,80,5,8
                                5,64
CAF7 05 01 40           .BYT    005,1,64,20,5,0,0,0
CAFF 00 00 00           .BYT    0,0,0,0,0,0,0,0
CB07 00 00 00           .BYT    0,0,0,0,0,0,0,0
CB0F 00 00 00           .BYT    0,0,0,0,0,0,20,0
CB17 00 75 00           .BYT    0,117,0,0,88,0,0,80
CB1F 00 00 50           .BYT    0,0,80,0,1,80,1,17
CB27 40 05 55           .BYT    064,5,85,64,21,149,
                                128,85
CB2F 59 50 06           .BYT    089,80,6,85,80,5,85,
                                64
CB37 01 40 40           .BYT    001,64,64,1,64,64,5,
                                1
CB3F 00 00 00           .BYT    0,0,0,0,0,0,0,0
CB47 00 00 00           .BYT    0,0,0,0,0,0,0,0
CB4F 00 00 00           .BYT    0,0,0,0,0,0,0,0
CB57 00 00 00           .BYT    0,0,0,0,0,0,0,0
CB5F 00 00 00           .BYT    0,0,0,0,0,0,0,0
CB67 00 00 00           .BYT    0,0,0,0,0,0,0,64
CB6F 00 00 50           .BYT    0,0,80,0,0,81,16,0
CB77 55 54 00           .BYT    085,84,0,89,100,0,8
                                5,85
CB7F 00 56 59           .BYT    0,86,89,0,85,84,0,12
                                5
CB87 00 00 00           .BYT    0,0,0,0,0,0,0,42
CB8F A0 00 00           .BYT    160,0,0,0,0,0,0,255
CB97 FF FF 00           .BYT    255,255,0,170,165,1
                                5,255,255
CB9F 00 00 50           .BYT    0,0,80,42,129,64,1,1
                                7
CBA7 50 05 5A           .BYT    080,5,90,160,240,63
                                ,255,21
CBAF 59 50 56           .BYT    089,80,86,170,170,5
                                ,85,64
CBB7 FF FF 40           .BYT    255,255,64,20,5,0,1
                                70,40
```

```
CBBF 00 00 00                .BYT    0,0,0,0,0,0,0,0
CBCD            XPOSH        *=      *+6
CBD3            XPOSL        *=      *+6
CBD9            YPOS         *=      *+6
CBDF            XDIR         *=      *+6
CBE5            XDIRH        *=      *+6
CBEB            YDIR         *=      *+6
CBEC            CROSSX       *=      *+1
CBED            CRXMSB       *=      *+1
CBEE            CROSSY       *=      *+1
CBEF            FIRE         *=      *+1
CBF1            SCORE        *=      *+2
CBF3            TIMER        *=      *+2
CBF4            LENS         *=      *+1
CBF5            FILM         *=      *+1
CBF6            SPEED        *=      *+1
CBF7            DUR          *=      *+1
CBF8            YESFIRE      *=      *+1
CC0C            GARBAGE      *=      *+20
CC20            BLUR         *=      *+20
CC34            BORDER       *=      *+20
CC48            HAIR         *=      *+20
CC4A            TEMP         *=      *+2
CC4B            MASK         *=      *+1
CC4C            WHICH        *=      *+1
CC4D            GAME         *=      *+1
```

Campaign Manager Source Code

Source code by Todd Heimarck

```
0801                      *=     2049
;
; bkcm (file 1)
;
0801         DEMCOL       =      $43
0801         REPCOL       =      $82
0801         Z1           =      $F7
0801         Z2           =      $F9
0801         Z3           =      $FB
; zero-page locations
0801         Z4           =      $FD
0801         LEANTABLE    =      $37C0
0801         MSGPLACE     =      $9800
0801         MAPTABLE     =      $3600
0801         REGUMEN      =      $8000
0801         RAND         =      $D41B
; translates to sys2061
0801 0B 08 0A             .BYTE  11,8,10,0,158,50,48
0808 36 31 00             .BYTE  54,49,0,0,0
; main routines
080D 20 6E 0C             JSR    PRELIM
0810 20 F1 0C             JSR    PICKCAND
0813 20 7A 11             JSR    CONTROL
0816 20 6C 1F             JSR    ELECTIONDAY
; crtbl -- crunched table (for map)
0819 45 FA CC CRTBL       .BYTE  69,250,204,204,204,
                                 204,220,192
0821 00 00 00             .BYTE  0,0,0,5,229,255,167,
                                 255
0829 FF FF FF             .BYTE  255,255,255,178,30,
                                 128,0,0
0831 DB 5F FA             .BYTE  219,95,250,31,255,2
                                 55,255,255
0839 8F 2D 00             .BYTE  143,45,0,4,245,37,2
                                 55,255
0841 1F FF FF             .BYTE  31,255,255,255,241,
                                 197,250,76
0849 FF F8 5F             .BYTE  255,248,95,95,255,2
                                 55,255,255
0851 8F BF AF             .BYTE  143,191,175,245,25
                                 5,115,37,245
0859 FF FF FF             .BYTE  255,255,255,252,20
                                 4,254,250,247
0861 23 4C 20             .BYTE  35,76,32,15,247,255
                                 ,255,255
0869 FF FA FE             .BYTE  255,250,254,162,25
                                 0,47,18,0
0871 5F 15 FF             .BYTE  95,21,255,255,227,2
                                 55,92,252
0879 CC 3C CC             .BYTE  204,60,204,0,0,127,
                                 175,255
0881 FF FF FA             .BYTE  255,255,250,255,20
                                 4,60,207,176
0889 00 01 F2             .BYTE  0,1,242,255,255,191
                                 ,255,239
0891 AF FA F7             .BYTE  175,250,247,224,0,0
                                 ,0,1
0899 33 7F FF             .BYTE  51,127,255,255,242,
                                 255,255,240
08A1 00 00 4F             .BYTE  0,0,79,160,128,0,11
                                 9,255
```

```
08A9 2F E1 23                .BYTE  47,225,35,127,0,0,1
                                    3,255
08B1 00 60 00                .BYTE  0,96,0,7,176,0,0,0
08B9 7E 00 00                .BYTE  126,0,0,211,58,0,11
                                    2,0
08C1 02 00 00                .BYTE  2,0,0,0,1,250,0,16
08C9 00 A0 00                .BYTE  0,160,0,0,0,0,0,0
08D1 00 01 20                .BYTE  0,1,32,0,0,0,0,0
;
; ** newchrset
08D9 20 E3 08 NEWCHRSET      JSR    NEWCHR1
08DC 20 29 09                JSR    NEWCHR2
08DF 20 4E 09                JSR    NEWCHR3
08E2 60                      RTS
;
; ** newchr1 -- relocate chrs 0-64
08E3 AD 0E DC NEWCHR1        LDA    $DC0E
08E6 29 FE                   AND    #$FE
08E8 8D 0E DC                STA    $DC0E
08EB A5 01                   LDA    $01
08ED 29 FB                   AND    #$FB
08EF 85 01                   STA    $01
08F1 A9 D1                   LDA    #$D1
08F3 85 FC                   STA    $FC
08F5 A9 39                   LDA    #$39
08F7 85 FE                   STA    $FE
08F9 A0 00                   LDY    #$00
08FB 84 FB                   STY    $FB
08FD 84 FD                   STY    $FD
08FF B1 FB    NEW1           LDA    ($FB),Y
0901 91 FD                   STA    ($FD),Y
0903 88                      DEY
0904 D0 F9                   BNE    NEW1
0906 C6 FC                   DEC    $FC
0908 C6 FE                   DEC    $FE
090A A9 37                   LDA    #$37
090C C5 FE                   CMP    $FE
090E D0 EF                   BNE    NEW1
0910 A5 01                   LDA    $01
0912 09 04                   ORA    #$04
0914 85 01                   STA    $01
0916 AD 0E DC                LDA    $DC0E
0919 09 01                   ORA    #1
091B 8D 0E DC                STA    $DC0E
091E AD 18 D0                LDA    $D018
0921 29 F0                   AND    #$F0
0923 09 0E                   ORA    #$0E
0925 8D 18 D0                STA    $D018
0928 60                      RTS
;
; ** newchr2 -- punctuation
0929 A9 39    NEWCHR2        LDA    #$39
092B 85 FC                   STA    $FC
092D 85 FE                   STA    $FE
092F A9 50                   LDA    #$50
0931 85 FB                   STA    $FB
0933 A9 D0                   LDA    #$D0
0935 85 FD                   STA    $FD
0937 20 44 09                JSR    NEW2
```

```
093A A9 18                   LDA   #$18
093C 85 FB                   STA   $FB
093E A9 D8                   LDA   #$D8
0940 85 FD                   STA   $FD
0942 C6 FE                   DEC   $FE
0944 A0 27      NEW2         LDY   #$27
0946 B1 FB      NEW3         LDA   ($FB),Y
0948 91 FD                   STA   ($FD),Y
094A 88                      DEY
094B 10 F9                   BPL   NEW3
094D 60                      RTS
;
; ** newchr3 -- quarter square graphics
094E A9 FF      NEWCHR3      LDA   #$FF
0950 8D 03 38                STA   $3803
0953 A9 F0                   LDA   #$F0
0955 8D 02 38                STA   $3802
0958 A9 0F                   LDA   #$0F
095A 8D 01 38                STA   $3801
095D A2 00                   LDX   #0
095F 8E 00 38                STX   $3800
0962 86 FB                   STX   $FB
0964 8A         NEW6         TXA
0965 20 75 09                JSR   NEW5
0968 8A                      TXA
0969 20 72 09                JSR   NEW4
096C E8                      INX
096D E0 10                   CPX   #$10
096F D0 F3                   BNE   NEW6
0971 60                      RTS
0972 EA         NEW4         NOP
0973 4A                      LSR
0974 4A                      LSR
0975 29 03      NEW5         AND   #3
0977 A8                      TAY
0978 B9 00 38                LDA   $3800,Y
097B A0 03                   LDY   #3
097D 91 FB      NEW7         STA   ($FB),Y
097F 88                      DEY
0980 10 FB                   BPL   NEW7
0982 E6 FB                   INC   $FB
0984 E6 FB                   INC   $FB
0986 E6 FB                   INC   $FB
0988 E6 FB                   INC   $FB
098A 60                      RTS
;
; ** uncrunchr -- set up table at $3600 for map
098B A9 36      UNCRUNCHR    LDA   #$36
098D 85 FC                   STA   $FC
098F A9 00                   LDA   #0
0991 85 FB                   STA   $FB
0993 A8                      TAY
0994 AA                      TAX
0995 E0 BC      CRUN1        CPX   #$BC
0997 D0 01                   BNE   CRUN2
; main rts of uncrunchr
0999 60                      RTS
; -- crunched table (nybbles)
099A BD 19 08   CRUN2        LDA   CRTBL,X
099D 48                      PHA
099E 4A                      LSR
099F 38                      SEC
09A0 6A                      ROR
09A1 4A                      LSR
09A2 4A                      LSR
09A3 91 FB                   STA   ($FB),Y
09A5 20 B5 09                JSR   UNC1
09A8 68                      PLA
09A9 29 0F                   AND   #$0F
09AB 09 20                   ORA   #$20
```

```
09AD 91 FB                   STA   ($FB),Y
09AF 20 B5 09                JSR   UNC1
09B2 E8                      INX
09B3 D0 E0                   BNE   CRUN1
09B5 C9 20      UNC1         CMP   #$20
09B7 D0 04                   BNE   UNC2
09B9 09 C0                   ORA   #$C0
09BB 91 FB                   STA   ($FB),Y
09BD C8         UNC2         INY
09BE C0 19                   CPY   #$19
09C0 F0 01                   BEQ   UNC3
; main rts of unc1
09C2 60                      RTS
09C3 A9 00      UNC3         LDA   #0
09C5 91 FB                   STA   ($FB),Y
09C7 A8                      TAY
09C8 18                      CLC
09C9 A9 1A                   LDA   #$1A
09CB 65 FB                   ADC   $FB
09CD 85 FB                   STA   $FB
09CF 90 02                   BCC   UNC4
09D1 E6 FC                   INC   $FC
09D3 60         UNC4         RTS
;
; ** colortogl
; color registers
09D4            COLREG       =     $D020
09D4 0C 01 03   COLORS       .BYTE 12,1,3,2,14
09D9 A0 04      COLORTOGL    LDY   #4
09DB B9 D4 09   COL1         LDA   COLORS,Y
09DE 99 20 D0                STA   COLREG,Y
09E1 88                      DEY
09E2 10 F7                   BPL   COL1
09E4 AD 11 D0                LDA   $D011
09E7 09 40                   ORA   #$40
; enable extended b/g color mode
09E9 8D 11 D0                STA   $D011
09EC 60                      RTS
;
; ** makemap
09ED 20 F7 09   MAKEMAP      JSR   CLRSCRCOL
09F0 20 6E 0A                JSR   CHRMAPPUT
09F3 20 B9 0A                JSR   SETCOLMEM
09F6 60                      RTS
;
; ** clrscrcol -- clear screen and set colors
09F7 A9 93      CLRSCRCOL    LDA   #$93
09F9 20 D2 FF                JSR   $FFD2
09FC A0 03                   LDY   #3
09FE 20 57 0A                JSR   CLSPACE2
0A01 A9 90                   LDA   #$90
0A03 20 D2 FF                JSR   $FFD2
0A06 A9 AB                   LDA   #$AB
0A08 20 D2 FF                JSR   $FFD2
0A0B A9 A3                   LDA   #$A3
0A0D 20 65 0A                JSR   SPACE25
0A10 A9 A7                   LDA   #$A7
0A12 20 D2 FF                JSR   $FFD2
0A15 A2 0F                   LDX   #$0F
0A17 A0 03      BARSPACE     LDY   #3
0A19 20 52 0A                JSR   CLSPACE1
0A1C A9 AA                   LDA   #$AA
0A1E 20 D2 FF                JSR   $FFD2
0A21 A9 9A                   LDA   #$9A
0A23 20 D2 FF                JSR   $FFD2
0A26 A9 A0                   LDDA   #$A0
0A28 20 65 0A                JSR   SPACE25
0A2B A9 90                   LDA   #$90
0A2D·20 D2 FF                JSR   $FFD2
0A30 A9 A5                   LDA   #$A5
```

```
0A32 20 D2 FF              JSR   $FFD2
0A35 CA                    DEX
0A36 D0 DF                 BNE   BARSPACE
0A38 A0 03                 LDY   #3
0A3A 20 52 0A              JSR   CLSPACE1
0A3D A9 AE                 LDA   #$AE
0A3F 20 D2 FF              JSR   $FFD2
0A42 A9 AC                 LDA   #$AC
0A44 20 65 0A              JSR   SPACE25
0A47 A9 AD                 LDA   #$AD
0A49 20 D2 FF              JSR   $FFD2
0A4C A9 92                 LDA   #146
0A4E 20 D2 FF              JSR   $FFD2
; main rts of clrscrcol
0A51 60                    RTS
;
0A52 A9 0D     CLSPACE1    LDA   #13
0A54 20 D2 FF              JSR   $FFD2
0A57 A9 20     CLSPACE2    LDA   #32
0A59 20 D2 FF  CLSPCLOOP   JSR   $FFD2
0A5C 88                    DEY
0A5D D0 FA                 BNE   CLSPCLOOP
0A5F A9 12                 LDA   #18
0A61 20 D2 FF              JSR   $FFD2
0A64 60                    RTS
;
0A65 A0 19     SPACE25     LDY   #25
0A67 20 D2 FF  SPCLP       JSR   $FFD2
0A6A 88                    DEY
0A6B D0 FA                 BNE   SPCLP
0A6D 60                    RTS
;
; ** chrmapput
0A6E A9 04     CHRMAPPUT   LDA   #4
0A70 85 FE                 STA   $FE
0A72 A9 2C                 LDA   #$2C
; location for screen
0A74 85 FD                 STA   $FD
0A76 A9 36                 LDA   #>MAPTABLE
0A78 85 FC                 STA   $FC
0A7A A9 00                 LDA   #<MAPTABLE
0A7C 85 FB                 STA   $FB
0A7E A9 00                 LDA   #0
0A80 A8                    TAY
0A81 A2 0F                 LDX   #$0F
0A83 B1 FB     MPPUT1      LDA   ($FB),Y
0A85 D0 07                 BNE   MPPUT2
0A87 20 A0 0A              JSR   MPSR1
0A8A CA                    DEX
0A8B D0 F6                 BNE   MPPUT1
0A8D 60                    RTS
0A8E 91 FD     MPPUT2      STA   ($FD),Y
0A90 C8                    INY
0A91 D0 F0                 BNE   MPPUT1
0A93 29 3F     UNSECT      AND   #$3F
0A95 AA                    TAX
0A96 BD C0 37              LDA   LEANTABLE,X
0A99 29 C0                 AND   #$C0
0A9B 11 F7     BLIX        ORA   ($F7),Y
0A9D 91 F7                 STA   ($F7),Y
0A9F 60                    RTS
0AA0 18        MPSR1       CLC
0AA1 A9 1A                 LDA   #$1A
0AA3 65 FB                 ADC   $FB
0AA5 85 FB                 STA   $FB
0AA7 90 02                 BCC   MPS2
0AA9 E6 FC                 INC   $FC
0AAB A9 28     MPS2        LDA   #$28
0AAD 18                    CLC
0AAE 65 FD                 ADC   $FD
```

```
0AB0 85 FD                 STA   $FD
0AB2 90 02                 BCC   MPS3
0AB4 E6 FE                 INC   $FE
0AB6 A0 00     MPS3        LDY   #0
0AB8 60                    RTS
;
; ** setcolmem
0AB9 A9 0F     SETCOLMEM   LDA   #15
0ABB 85 F9                 STA   $F9
0ABD A9 D8                 LDA   #$D8
0ABF 85 FE                 STA   $FE
0AC1 A9 2C                 LDA   #$2C
0AC3 85 FD                 STA   $FD
0AC5 85 F7                 STA   $F7
0AC7 A9 04                 LDA   #4
0AC9 85 F8                 STA   $F8
0ACB A9 22                 LDA   #>MAPAMAP
0ACD 85 FC                 STA   $FC
0ACF A9 AD                 LDA   #<MAPAMAP
0AD1 85 FB                 STA   $FB
0AD3 A0 18     SLECT       LDY   #24
0AD5 B1 FB     SETCLOOP    LDA   ($FB),Y
0AD7 C9 00                 CMP   #0
0AD9 F0 2B                 BEQ   SETC1
0ADB 85 02                 STA   $02
0ADD 29 3F                 AND   #$3F
0ADF AA                    TAX
0AE0 BD C0 37              LDA   LEANTABLE,X
0AE3 29 0F                 AND   #$0F
0AE5 91 FD                 STA   ($FD),Y
0AE7 A9 C0                 LDA   #$C0
0AE9 24 02                 BIT   $02
0AEB F0 19                 BEQ   SETC1
0AED 30 08                 BMI   SETC2
0AEF BD 78 22              LDA   SECTABLE,X
0AF2 20 93 0A              JSR   UNSECT
0AF5 D0 0F                 BNE   SETC1
0AF7 50 07     SETC2       BVC   SETC3
0AF9 A9 C0                 LDA   #$C0
0AFB 20 9B 0A              JSR   BLIX
0AFE D0 06                 BNE   SETC1
0B00 BD 79 22  SETC3       LDA   TERTABLE,X
0B03 20 93 0A              JSR   UNSECT
0B06 EA        SETC1       NOP
0B07 88                    DEY
0B08 10 CB                 BPL   SETCLOOP
0B0A A9 19     ENDLINEM    LDA   #25
0B0C 18                    CLC
0B0D 65 FB                 ADC   $FB
0B0F 85 FB                 STA   $FB
0B11 90 02                 BCC   ENDL1
0B13 E6 FC                 INC   $FC
0B15 C6 F9     ENDL1       DEC   $F9
0B17 D0 01                 BNE   ENDL2
; exit setcolmem
0B19 60                    RTS
0B1A A9 28     ENDL2       LDA   #$28
0B1C 18                    CLC
0B1D 65 F7                 ADC   $F7
0B1F 85 F7                 STA   $F7
0B21 90 02                 BCC   ENDL3
0B23 E6 F8                 INC   $F8
0B25 A9 28     ENDL3       LDA   #$28
0B27 18                    CLC
0B28 65 FD                 ADC   $FD
0B2A 85 FD                 STA   $FD
0B2C 90 A5                 BCC   SLECT
0B2E E6 FE                 INC   $FE
0B30 D0 A1                 BNE   SLECT
;
```

```
; ** dorainbow
0B32 AD 12 D0  DORAINBOW  LDA   $D012
0B35 48                   PHA
0B36 65 A2                ADC   $A2
0B38 4A                   LSR
0B39 4A                   LSR
0B3A 4A                   LSR
0B3B A8                   TAY
0B3C 68                   PLA
0B3D E5 A2                SBC   $A2
0B3F 4A                   LSR
0B40 8D 20 D0             STA   $D020
0B43 8C 24 D0             STY   $D024
; exit dorainbow
0B46 60                   RTS
0B47 1F 43 41  CAMMGR     .ASC  "{BLU}CAMPAIG
                                N MANAGER"
0B58 0D 00                .BYTE 13,0
0B5A A2 12     CMLOGOPUT  LDX   #18
0B5C A0 08                LDY   #8
0B5E 18                   CLC
; kernal plot
0B5F 20 F0 FF             JSR   $FFF0
0B62 A2 00                LDX   #0
0B64 BD 47 0B  LOGGLOOP   LDA   CAMMGR,X
0B67 F0 06                BEQ   OUTOLOGG
0B69 20 D2 FF             JSR   $FFD2
0B6C E8                   INX
0B6D D0 F5                BNE   LOGGLOOP
0B6F A0 05     OUTOLOGG   LDY   #5
0B71 A9 01                LDA   #1
0B73 8D 86 02             STA   646
0B76 A9 12                LDA   #18
; reverse on
0B78 20 D2 FF             JSR   $FFD2
0B7B A2 28     OUTPLOGG   LDX   #40
0B7D AD 86 02             LDA   646
0B80 49 03                EOR   #3
; toggle red and white
0B82 8D 86 02             STA   646
0B85 A9 A3                LDA   #163
0B87 20 D2 FF  FLAGLER    JSR   $FFD2
0B8A CA                   DEX
0B8B D0 FA                BNE   FLAGLER
0B8D 88                   DEY
0B8E D0 EB                BNE   OUTPLOGG
0B90 A9 92                LDA   #146
; exit cmlogoput via jump
0B92 4C D2 FF             JMP   $FFD2
;
; ** initpv -- sets up stpvotes table at $9100/popular votes
0B95 A9 92     INITPV     LDA   #>STPVOTES
0B97 85 FE                STA   $FE
0B99 A9 00                LDA   #<STPVOTES
0B9B 85 FD                STA   $FD
0B9D A2 00                LDX   #0
0B9F E8        PVMLOOP    INX
0BA0 EC 89 24             CPX   PVCRUNCH
0BA3 F0 2F                BEQ   EVDOT
0BA5 BD 89 24  DOPVM      LDA   PVCRUNCH,X
0BA8 85 F9                STA   $F9
0BAA 29 07                AND   #$07
; == democrats
0BAC 85 F7                STA   $F7
0BAE A5 F9                LDA   $F9
0BB0 4A                   LSR
0BB1 4A                   LSR
0BB2 4A                   LSR
0BB3 4A                   LSR
0BB4 29 07                AND   #$07
```

```
; == republicans
0BB6 85 F8                STA   $F8
0BB8 A0 02                LDY   #2
0BBA 20 E6 0B             JSR   PUCKER
0BBD A5 F7                LDA   $F7
0BBF A0 01                LDY   #1
0BC1 20 E6 0B             JSR   PUCKER
0BC4 A9 FF                LDA   #$FF
0BC6 A0 05                LDY   #5
0BC8 91 FD                STA   ($FD),Y
0BCA A9 05                LDA   #5
0BCC 18                   CLC
0BCD 65 FD                ADC   $FD
0BCF 85 FD                STA   $FD
0BD1 4C 9F 0B             JMP   PVMLOOP
; patch - set up stevote table/electoral votes
0BD4 A9 00     EVDOT      LDA   #0
0BD6 AA                   TAX
0BD7 A8                   TAY
0BD8 B9 44 22  EVOLUT     LDA   CREVOTE,Y
0BDB 9D 00 78             STA   STEVOTE,X
0BDE E8                   INX
0BDF E8                   INX
0BE0 C8                   INY
0BE1 C0 34                CPY   #52
0BE3 D0 F3                BNE   EVOLUT
0BE5 60                   RTS
0BE6 91 FD     PUCKER     STA   ($FD),Y
0BE8 C8                   INY
0BE9 C8                   INY
0BEA 91 FD                STA   ($FD),Y
0BEC 60                   RTS
; ** now initialize issues for each state
0BED A9 91     INITISS    LDA   #>STISSUES
0BEF 85 F8                STA   Z1+1
0BF1 A9 00                LDA   #<STISSUES
0BF3 85 F7                STA   Z1
0BF5 E6 F7                INC   Z1
0BF7 85 FE                STA   Z4+1
0BF9 AA                   TAX
;z1 is indirect pointer/z4+1 is statenumber
0BFA A2 00                LDX   #0
0BFC BD BD 24  BGINTISS   LDA   ISSU1,X
0BFF 85 F9                STA   Z2
0C01 4A                   LSR
0C02 4A                   LSR
0C03 4A                   LSR
0C04 4A                   LSR
0C05 85 FA                STA   Z2+1
0C07 BD F0 24             LDA   ISSU2,X
0C0A 85 FB                STA   Z3
0C0C 4A                   LSR
0C0D 4A                   LSR
0C0E 85 FC                STA   Z3+1
0C10 4A                   LSR
0C11 4A                   LSR
0C12 85 FD                STA   Z4
0C14 A0 04                LDY   #4
0C16 A2 04                LDX   #4
0C18 B5 F9     UNDPLOOOP  LDA   Z2,X
0C1A 48                   PHA
0C1B 29 03                AND   #3
0C1D 18                   CLC
0C1E 69 01                ADC   #1
; dems
0C20 91 F7                STA   (Z1),Y
0C22 68                   PLA
0C23 4A                   LSR
0C24 4A                   LSR
0C25 29 03                AND   #3
```

```
0C27 18                         CLC
0C28 69 03                      ADC   #3
0C2A 0A                         ASL
0C2B 0A                         ASL
0C2C 0A                         ASL
0C2D 0A                         ASL
0C2E 11 F7                      ORA   (Z1),Y
; reps
0C30 91 F7                      STA   (Z1),Y
0C32 88                         DEY
0C33 CA                         DEX
0C34 10 E2                      BPL   UNDPLOOOP
;
0C36 E6 F7                      INC   Z1
0C38 A0 02                      LDY   #2
0C3A A9 0F       UNDP           LDA   #$0F
0C3C 31 F7                      AND   (Z1),Y
0C3E AA                         TAX
0C3F E8                         INX
0C40 8A                         TXA
0C41 0A                         ASL
0C42 0A                         ASL
0C43 0A                         ASL
0C44 0A                         ASL
0C45 85 02                      STA   $02
0C47 8A                         TXA
0C48 05 02                      ORA   $02
0C4A 91 F7                      STA   (Z1),Y
0C4C 88                         DEY
0C4D D0 EB                      BNE   UNDP
0C4F E6 F7                      INC   Z1
0C51 E6 F7                      INC   Z1
0C53 E6 F7                      INC   Z1
0C55 E6 F7                      INC   Z1
0C57 E6 FE                      INC   Z4+1
0C59 A6 FE                      LDX   Z4+1
0C5B E0 33                      CPX   #51
0C5D D0 9D                      BNE   BGINTISS
0C5F 60                         RTS
;
0C60 A9 FF       RANDON         LDA   #$FF
0C62 8D 0F D4                   STA   $D40F
0C65 A9 80                      LDA   #$80
0C67 8D 12 D4                   STA   $D412
0C6A 8D 18 D4                   STA   $D418
0C6D 60                         RTS
; ** prelim -- sets up tables
0C6E A2 40       PRELIM         LDX   #64
0C70 A9 00                      LDA   #0
0C72 9D 00 8F    PRELPP         STA   DAY,X
0C75 9D 40 8F                   STA   OTHERCAND,X
0C78 CA                         DEX
0C79 D0 F7                      BNE   PRELPP
0C7B A9 80                      LDA   #$80
0C7D 8D 8A 02                   STA   650
0C80 A9 08                      LDA   #8
; set repeat key & disable c=shift
0C82 20 D2 FF                   JSR   $FFD2
0C85 20 95 0B                   JSR   INITPV
0C88 20 FA 1A                   JSR   CKDEMREP
; sets up initial leanings
0C8B 20 6C 1B                   JSR   TRANSLEANCOLR
; sets up menu table at $8000
0C8E 20 80 17                   JSR   SETREGMENU
0C91 20 8B 09                   JSR   UNCRUNCHR
0C94 20 1E 1C                   JSR   SETSEL
0C97 20 D9 08                   JSR   NEWCHRSET
0C9A 20 ED 0B                   JSR   INITISS
0C9D 20 D9 09    PRELIMXX       JSR   COLORTOGL
0CA0 20 ED 09                   JSR   MAKEMAP
```

```
0CA3 A9 9E                      LDA   #$9E
0CA5 20 D2 FF                   JSR   $FFD2
0CA8 20 5A 0B                   JSR   CMLOGOPUT
0CAB 20 1E 14                   JSR   SETUPMENU
0CAE 20 32 0B    P1             JSR   DORAINBOW
0CB1 20 1A 1F                   JSR   IJKL
0CB4 AD 6B 1F                   LDA   JB
0CB7 F0 F5                      BEQ   P1
0CB9 20 D9 09                   JSR   COLORTOGL
0CBC 20 60 0C                   JSR   RANDON
; ** poneparty -- party of player one
0CBF A2 04       PONEPARTY      LDX   #4
0CC1 A0 05                      LDY   #5
0CC3 20 A3 1C                   JSR   SELECT
0CC6 8D 15 8F                   STA   PARTY
0CC9 8D 23 25                   STA   FIRSTCAN
; ** bincum -- check for incumbent
0CCC A2 07       BINCUM         LDX   #7
0CCE A0 09                      LDY   #9
0CD0 20 A3 1C                   JSR   SELECT
0CD3 A2 00                      LDX   #0
0CD5 A0 00                      LDY   #0
0CD7 C9 00                      CMP   #0
0CD9 F0 07                      BEQ   NONINCUM
0CDB 29 01                      AND   #1
0CDD F0 02                      BEQ   TWOINCUM
0CDF CA                         DEX
0CE0 C8                         INY
0CE1 88          TWOINCUM       DEY
0CE2 8E 0F 8F    NONINCUM       STX   INCUMB
0CE5 8C 4F 8F                   STY   OTINCUMB
0CE8 20 2E 11    OKSTPAR        JSR   PRESSFIRE
0CEB D0 03                      BNE   GONNGY
0CED 4C 9D 0C                   JMP   PRELIMXX
0CF0 60          GONNGY         RTS
;
; bkcm.pick
;
; *** pickcand -- routine to pick a candidate
; first generate five candidates
0CF1 A9 00       PICKCAND       LDA   #0
0CF3 8D 24 25                   STA   WEEKNO
0CF6 A9 80                      LDA   #<CANSTACK
0CF8 85 F7                      STA   Z1
0CFA A9 8F                      LDA   #>CANSTACK
; holding area
0CFC 85 F8                      STA   Z1+1
0CFE A9 05                      LDA   #5
; number of candidates generated
0D00 85 02                      STA   $02
0D02 A0 05       ANIDLOOP       LDY   #5
0D04 A2 03       ANID5          LDX   #3
0D06 AD 1B D4    ANODLOOP       LDA   RAND
0D09 29 03                      AND   #3
0D0B 95 F9                      STA   Z2,X
0D0D CA                         DEX
0D0E D0 F6                      BNE   ANODLOOP
0D10 A9 01                      LDA   #1
0D12 25 FA                      AND   Z2+1
0D14 18                         CLC
0D15 69 01                      ADC   #1
0D17 65 FB                      ADC   Z3
0D19 65 FC                      ADC   Z3+1
0D1B 91 F7                      STA   (Z1),Y
0D1D 88                         DEY
0D1E D0 E4                      BNE   ANID5
; (attributes are done)
0D20 A0 06                      LDY   #6
0D22 AD 1B D4    ANNNLOOP       LDA   RAND
0D25 29 03                      AND   #3
```

```
0D27 AA                 TAX
0D28 C0 08              CPY   #8
0D2A F0 0A              BEQ   UPANN
0D2C C0 09              CPY   #9
0D2E F0 06              BEQ   UPANN
0D30 AD 15 8F           LDA   PARTY
0D33 F0 02              BEQ   UPANI2
0D35 E8                 INX
0D36 E8        UPANN    INX
0D37 E8        UPANI2   INX
0D38 8A                 TXA
0D39 91 F7              STA   (Z1),Y
0D3B C8                 INY
0D3C C0 0B              CPY   #11
0D3E D0 E2              BNE   ANNNLOOP
; (issues are set up)
0D40 AD 1B D4  STRNNN   LDA   RAND
0D43 29 3F              AND   #63
0D45 F0 F9              BEQ   STRNNN
0D47 C9 34              CMP   #52
0D49 B0 F5              BCS   STRNNN
0D4B 91 F7              STA   (Z1),Y
0D4D C8                 INY
; (and the home state)
0D4E AD 0F 8F           LDA   INCUMB
0D51 91 F7              STA   (Z1),Y
0D53 D0 09              BNE   COTHII
0D55 AD 1B D4           LDA   RAND
0D58 29 07              AND   #7
0D5A 0A                 ASL
0D5B 0A                 ASL
; (and current job)
0D5C 91 F7              STA   (Z1),Y
0D5E 20 DC 0E  COTHII   JSR   UPTZ1
0D61 C6 02              DEC   $02
0D63 D0 9D              BNE   ANIDLOOP
;
; now print to screen
0D65 A0 00     FRIIS    LDY   #0
0D67 8C 2D 11           STY   CANNRO
0D6A A9 80     NEXN9    LDA   #<CANSTACK
0D6C 85 F7              STA   Z1
0D6E A9 8F              LDA   #>CANSTACK
0D70 85 F8              STA   Z1+1
0D72 AD 2D 11           LDA   CANNRO
0D75 C9 05              CMP   #5
0D77 B0 EC              BCS   FRIIS
0D79 AA                 TAX
0D7A F0 06              BEQ   XFERX
0D7C 20 DC 0E  NUKKL    JSR   UPTZ1
0D7F CA                 DEX
0D80 D0 FA              BNE   NUKKL
0D82 EE 2D 11  XFERX    INC   CANNRO
0D85 A0 05              LDY   #5
0D87 B1 F7     ARTIB    LDA   (Z1),Y
0D89 99 0F 8F           STA   INCUMB,Y
0D8C 88                 DEY
; xfer attributes
0D8D D0 F8              BNE   ARTIB
0D8F A0 06              LDY   #6
0D91 A2 00              LDX   #0
0D93 B1 F7     ISIST2   LDA   (Z1),Y
0D95 9D 1B 8F           STA   STAND0,X
0D98 C8                 INY
0D99 E8                 INX
0D9A E0 05              CPX   #5
; xfer issues
0D9C D0 F5              BNE   ISIST2
0D9E B1 F7              LDA   (Z1),Y
0DA0 8D 0C 8F           STA   HOMESTATE
```

```
0DA3 8D 0A 8F              STA   LASTSTATE
0DA6 C8                    INY
0DA7 B1 F7                 LDA   (Z1),Y
; home&job
0DA9 8D 0D 8F              STA   JOB
0DAC 20 E4 0E              JSR   SETUPATTRIB
0DAF 20 EE 0E              JSR   PICKONE
0DB2 D0 03                 BNE   DUMPR
; not selected
0DB4 4C 6A 0D              JMP   NEXN9
; a candidate has been chosen
; now make up a name
0DB7 20 2E 11  DUMPR       JSR   PRESSFIRE
0DBA F0 A9                 BEQ   FRIIS
0DBC A2 00     REDUUT      LDX   #0
0DBE 86 F8                 STX   Z1+1
0DC0 A0 06                 LDY   #6
0DC2 18                    CLC
0DC3 20 F0 FF              JSR   $FFF0
0DC6 AD 15 8F              LDA   PARTY
0DC9 CD 23 25              CMP   FIRSTCAN
0DCC F0 02                 BEQ   PICKANAME
0DCE A2 0C                 LDX   #12
0DD0 86 F7     PICKANAME   STX   Z1
0DD2 BD 9E 14  PICKLED     LDA   P1NAME,X
0DD5 F0 06                 BEQ   QMARK
0DD7 20 D2 FF              JSR   $FFD2
0DDA E8                    INX
0DDB D0 F5                 BNE   PICKLED
0DDD A9 3F     QMARK       LDA   #63
0DDF 20 D2 FF              JSR   $FFDD2
0DE2 A6 F7                 LDX   Z1
0DE4 A0 0A                 LDY   #10
0DE6 A9 2C                 LDA   #44
0DE8 9D 9E 14  LPP69       STA   P1NAME,X
0DEB E8                    INX
0DEC 88                    DEY
0DED D0 F9                 BNE   LPP69
0DEF 20 E4 FF  GRENDL      JSR   $FFE4
0DF2 F0 FB                 BEQ   GRENDL
0DF4 C9 0D                 CMP   #13
0DF6 F0 27                 BEQ   DDOUS
0DF8 C9 20     PDOUS       CMP   #32
0DFA F0 08                 BEQ   PRENAMEE
0DFC C9 41                 CMP   #65
0DFE 90 EF                 BCC   GRENDL
0E00 C9 5B                 CMP   #91
0E02 B0 EB                 BCS   GRENDL
0E04 E6 F8     PRENAMEE    INC   Z1+1
0E06 A6 F8                 LDX   Z1+1
0E08 E0 0B                 CPX   #11
0E0A F0 13                 BEQ   DDOUS
0E0C A4 F7                 LDY   Z1
0E0E 99 9E 14              STA   P1NAME,Y
0E11 29 3F                 AND   #63
0E13 9D 05 04              STA   1029,X
0E16 E6 F7                 INC   Z1
0E18 A9 2F                 LDA   #47
0E1A 9D 06 04              STA   1030,X
0E1D D0 D0                 BNE   GRENDL
0E1F 20 26 0F  DDOUS       JSR   DOATTRIB
0E22 20 2E 11              JSR   PRESSFIRE
; name is all set
0E25 F0 95                 BEQ   REDUUT
;
; now translate other attributes
0E27 AD 0F 8F              LDA   INCUMB
0E2A 29 02                 AND   #2
0E2C 18                    CLC
0E2D 6D 10 8F              ADC   CHAR
```

```
0E30 0A                 ASL
0E31 6D 12 8F           ADC   INTL
; effectiveness
0E34 8D 16 8F           STA   CAMPEFF
0E37 AD 11 8F           LDA   STAM
0E3A 0A                 ASL
0E3B 0A                 ASL
0E3C 69 09              ADC   #9
0E3E 38                 SEC
0E3F ED 13 8F           SBC   EXPR
; strength
0E42 8D 17 8F           STA   STREN
0E45 AD 1B D4           LDA   RAND
0E48 29 1F              AND   #31
0E4A 0A                 ASL
0E4B 6D 17 8F           ADC   STREN
0E4E 69 20              ADC   #32
; seedhealth
0E50 8D 08 8F           STA   HEALTH
0E53 AD 0F 8F           LDA   INCUMB
0E56 29 04              AND   #4
0E58 6D 13 8F           ADC   EXPR
0E5B 0A                 ASL
0E5C 6D 13 8F           ADC   EXPR
0E5F 6D 14 8F           ADC   APPL
; fundraising
0E62 8D 18 8F           STA   FUNDR
0E65 0A                 ASL
0E66 6D 12 8F           ADC   INTL
0E69 69 30              ADC   #48
; seedmoney
0E6B 8D 09 8F           STA   MONEY
0E6E AD 14 8F           LDA   APPL
0E71 09 08              ORA   #8
0E73 6D 10 8F           ADC   CHAR
; tv ads
0E76 8D 19 8F           STA   TVADS
0E79 AD 0F 8F           LDA   INCUMB
0E7C 29 07              AND   #7
0E7E 18                 CLC
0E7F 6D 12 8F           ADC   INTL
0E82 6D 11 8F           ADC   STAM
; last week
0E85 8D 1A 8F           STA   LASTWK
0E88 A2 00              LDX   #0
0E8A AD 0C 8F           LDA   HOMESTATE
0E8D E8       CLKHHN    INX
0E8E DD 7F 24           CMP   REGFIRST,X
0E91 B0 FA              BCS   CLKHHN
0E93 8E 20 8F           STX   CURREG
0E96 8E 0B 8F           STX   LASTREGI
0E99 8E 21 8F           STX   HOMEREGI
; all done creating player 1
; now do the other candidate
0E9C 20 84 1B           JSR   TGLPARTY
0E9F AD 15 8F           LDA   PARTY
0EA2 CD 23 25           CMP   FIRSTCAN
0EA5 F0 03              BEQ   DONEWITH
0EA7 4C F1 0C           JMP   PICKCAND
0EAA AD 0F 8F DONEWITH  LDA   INCUMB
0EAD 29 03              AND   #3
0EAF 8D 81 8F           STA   CANSTACK+1
; initial campaign
0EB2 20 F3 1B           JSR   CAMPONCE
0EB5 A9 00              LDA   #0
; and the home region
0EB7 8D 81 8F           STA   CANSTACK+1
0EBA AE 21 8F           LDX   HOMEREGI
0EBD BD 7F 24           LDA   REGFIRST,X
0EC0 A8                 TAY
```

```
0EC1 CA                 DEX
0EC2 BD 7F 24           LDA   REGFIRST,X
0EC5 AA                 TAX
0EC6 CA                 DEX
0EC7 20 F7 1B           JSR   CAMPREGK
0ECA 20 84 1B           JSR   TGLPARTY
0ECD AD 15 8F           LDA   PARTY
0ED0 CD 23 25           CMP   FIRSTCAN
0ED3 D0 D5              BNE   DONEWITH
0ED5 20 FA 1A           JSR   CKDEMREP
0ED8 20 6C 1B           JSR   TRANSLEANCOLR
0EDB 60                 RTS
;
0EDC A9 10    UPTZ1     LDA   #16
0EDE 18                 CLC
0EDF 65 F7              ADC   Z1
0EE1 85 F7              STA   Z1
0EE3 60                 RTS
; do the attrib & stands
0EE4 20 ED 09 SETUPATTR JSR   MAKEMAP
0EE7 20 CD 15           JSR   MENUITIN
0EEA 20 26 0F           JSR   DOATTRIB
0EED 60                 RTS
;
0EEE A9 0F    PICKONE   LDA   #15
0EF0 85 FD              STA   $FD
0EF2 A9 16              LDA   #22
0EF4 85 FE              STA   $FE
0EF6 A9 1D              LDA   #29
0EF8 85 A7              STA   $A7
0EFA A2 F0              LDX   #<ASKIRT
0EFC A0 10              LDY   #>ASKIRT
0EFE 20 B8 14           JSR   PRINIT
0F01 AD 15 8F           LDA   PARTY
0F04 F0 0D              BEQ   PXZ9
0F06 A2 0A              LDX   #10
0F08 BD 75 14 OLLALE    LDA   PARNAME1,X
0F0B 29 3F              AND   #63
0F0D 9D 9C 06           STA   1692,X
0F10 CA                 DEX
0F11 D0 F5              BNE   OLLALE
0F13 AD 15 8F PXZ9      LDA   PARTY
0F16 CD 23 25           CMP   FIRSTCAN
0F19 F0 03              BEQ   SELCN4
0F1B EE 7D 06           INC   1661
0F1E A2 14    SELCN4    LDX   #20
0F20 A0 15              LDY   #21
0F22 20 A3 1C           JSR   SELECT
0F25 60                 RTS
;
; ** doattrib (sr for printing general info about candidate)
0F26 AE 15 8F DOATTRIB  LDX   PARTY
0F29 BD 28 25           LDA   PRTCO,X
; cyan (dems) or red (reps)
0F2C 20 D2 FF           JSR   $FFD2
0F2F A9 11              LDA   #17
0F31 85 FD              STA   $FD
0F33 A9 19              LDA   #25
0F35 85 FE              STA   $FE
0F37 A9 00              LDA   #0
0F39 85 A7              STA   $A7
0F3B A2 51              LDX   #<ATTRIB
0F3D A0 10              LDY   #>ATTRIB
0F3F 20 B8 14           JSR   PRINIT
0F42 A9 1F              LDA   #31
; blue color
0F44 20 D2 FF           JSR   $FFD2
0F47 A9 14              LDA   #20
0F49 85 FD              STA   $FD
0F4B A9 19              LDA   #25
```

```
0F4D 85 FE                   STA   $FE
0F4F A9 09                   LDA   #9
0F51 85 A7                   STA   $A7
0F53 A2 AB                   LDX   #<CANISS
0F55 A0 10                   LDY   #>CANISS
0F57 20 B8 14                JSR   PRINIT
; print list of issues
0F5A A2 13                   LDX   #19
0F5C E8          ONSTXL      INX
0F5D A0 0F                   LDY   #$0F
0F5F 18                      CLC
; plot to print the sliding scale
0F60 20 F0 FF                JSR   $FFF0
0F63 A2 31                   LDX   #$31
0F65 8A          PRLYWH      TXA
0F66 20 D2 FF                JSR   $FFD2
0F69 E8                      INX
0F6A E0 37                   CPX   #$37
; print 1-6
0F6C D0 F7                   BNE   PRLYWH
0F6E 38                      SEC
0F6F 20 F0 FF                JSR   $FFF0
0F72 E0 18                   CPX   #$18
; screen is almost set up
0F74 D0 E6                   BNE   ONSTXL
0F76 AD 0C 8F                LDA   HOMESTATE
0F79 0A                      ASL
0F7A AA                      TAX
0F7B BD DC 21                LDA   STNAME,X
0F7E 29 3F                   AND   #63
0F80 8D AB 06                STA   1707
0F83 E8                      INX
0F84 BD DC 21                LDA   STNAME,X
0F87 29 3F                   AND   #63
0F89 8D AC 06                STA   1708
0F8C A2 12                   LDX   #18
0F8E A0 02                   LDY   #2
0F90 18                      CLC
0F91 20 F0 FF                JSR   $FFF0
0F94 AE 15 8F                LDX   PARTY
0F97 BD 28 25                LDA   PRTCO,X
0F9A 20 D2 FF                JSR   $FFD2
0F9D AE 0D 8F                LDX   JOB
0FA0 30 0E                   BMI   SPECIINS
0FA2 A0 04                   LDY   #4
0FA4 BD 31 10    LSTLO       LDA   TITLES,X
0FA7 20 D2 FF                JSR   $FFD2
0FAA E8                      INX
0FAB 88                      DEY
0FAC D0 F6                   BNE   LSTLO
0FAE F0 0D                   BEQ   CBMM
0FB0 A2 00       SPECIINS    LDX   #0
0FB2 BD E4 10    PRESLOPH    LDA   ERPP,X
0FB5 F0 06                   BEQ   CBMM
0FB7 20 D2 FF                JSR   $FFD2
0FBA E8                      INX
0FBB D0 F5                   BNE   PRESLOPH
0FBD A9 9E       CBMM        LDA   #<P1NAME
0FBF 85 F7                   STA   Z1
0FC1 A9 14                   LDA   #>P1NAME
0FC3 85 F8                   STA   Z1+1
0FC5 A0 00                   LDY   #0
0FC7 AD 15 8F                LDA   PARTY
0FCA CD 23 25                CMP   FIRSTCAN
0FCD F0 02                   BEQ   DOREPTIL
0FCF A0 0C                   LDY   #12
0FD1 B1 F7       DOREPTIL    LDA   (Z1),Y
0FD3 F0 06                   BEQ   CHECKREPS
0FD5 20 D2 FF                JSR   $FFD2
0FD8 C8                      INY
```

```
0FD9 D0 F6                   BNE   DOREPTIL
0FDB AD 15 8F    CHECKREPS   LDA   PARTY
0FDE F0 20                   BEQ   ENDOFIT
0FE0 A2 0A                   LDX   #10
0FE2 BD 75 14    PARNAU      LDA   PARNAME1,X
0FE5 29 3F                   AND   #63
0FE7 9D F8 06                STA   1784,X
0FEA CA                      DEX
0FEB D0 F5                   BNE   PARNAU
0FED A9 14                   LDA   #20
0FEF 85 FD                   STA   $FD
0FF1 A9 16                   LDA   #22
0FF3 85 FE                   STA   $FE
0FF5 A9 09                   LDA   #9
0FF7 85 A7                   STA   $A7
0FF9 A2 D2                   LDX   #<RCANISS
0FFB A0 10                   LDY   #>RCANISS
0FFD 20 B8 14                JSR   PRINIT
1000 A2 04       ENDOFIT     LDX   #4
1002 A0 A0                   LDDY   #160
1004 BD 10 8F    AYYI        LDA   CHAR,X
1007 09 30                   ORA   #$30
; attribute #s
1009 99 27 07                STA   1831,Y
100C 98                      TYA
100D 38                      SEC
100E E9 28                   SBC   #40
1010 A8                      TAY
1011 CA                      DEX
1012 10 F0                   BPL   AYYI
1014 A2 04                   LDX   #4
1016 A0 A0                   LDY   #160
1018 98          STNR0       TYA
1019 18                      CLC
101A 7D 1B 8F                ADC   STAND0,X
101D A8                      TAY
101E B9 2E 07                LDA   1838,Y
1021 09 40                   ORA   #$40
; issues
1023 99 2E 07                STA   1838,Y
1026 98                      TYA
1027 38                      SEC
1028 E9 28                   SBC   #40
102A 29 F8                   AND   #$F8
102C A8                      TAY
102D CA                      DEX
102E 10 E8                   BPL   STNR0
1030 60                      RTS
; data tables for these routines
1031 53 45 4E    TITLES      .ASC  "SEN GOV REP RE
                                   V MS DR> V=
                                   P GEN"
1050 20                      .BYTE 32
1051 1B 2C 00    ATTRIB      .BYTE 27,44,0,27,44,0
1057 FF                      .BYTE 255
1058 2C 44 45                .ASC  ",DEMOCRATIC CAN
                                   DIDATE ......"
1074 00 FF                   .BYTE 0,255
1076 9C 2F 20                .ASC  "{PUR}/ CHAR X"
107F 00 FF                   .BYTE 0,255
1081 2F 20 53                .ASC  "/ STAM X"
1089 00 FF                   .BYTE 0,255
108B 2F 20 49                .ASC  "/ INTL X"
1093 00 FF                   .BYTE 0,255
1095 2F 20 45                .ASC  "/ EXPR X"
109D 00 FF                   .BYTE 0,255
109F 2F 20 41                .ASC  "/ APPL X"
10A7 00 00 00                .BYTE 0,0,0,0
10AB FF          CANISS      .BYTE 255
10AC 1F 55 4E                .ASC  "{BLU}UNEMP"
```

```
10B2 00 FF                .BYTE  0,255
10B4 50 4F 56             .ASC   "POVTY"
10B9 00 FF                .BYTE  0,255
10BB 41 47 52             .ASC   "AGRIC"
10C0 00 FF                .BYTE  0,255
10C2 45 44 55             .ASC   "EDUCN"
10C7 00 FF                .BYTE  0,255
10C9 44 46 45             .ASC   "DFENS"
10CE 00 00 00             .BYTE  0,0,0,0
10D2 FF       RCANISS     .BYTE  255
10D3 1F 49 4E             .ASC   "{BLU}INFLN"
10D9 00 FF                .BYTE  0,255
10DB 43 52 49             .ASC   "CRIME"
10E0 00 00 00             .BYTE  0,0,0,0
10E4 9D 50 52 ERPP        .ASC   "{LEFT}PRESIDENT
                                 "
10EE 20 00                .BYTE  32,0
;
10F0 FF       ASKIRT      .BYTE  255
10F1 97 50 4C             .ASC   " [<4>]PLAYER [1"
10FB 00 FF                .BYTE  0,255
10FD 44 45 4D             .ASC   "DEMOCRATIC"
1107 00 FF                .BYTE  0,255
1109 43 4F 4E             .ASC   "CONVENTION"
1113 00                   .BYTE  0
1114 0A A6 00             .BYTE  10,166,0,5,32,0,255
111B 1E 20 2F             .ASC   "{GRN} / NO"
1121 00 FF                .BYTE  0,255
1123 20 2F 20             .ASC   " / YES"
1129 00 00 00             .BYTE  0,0,0,0
;
112D 00       CANNRO      .BYTE  0
; firebutton to continue routine
112E AD 05 04 PRESSFIRE   LDA    1029
1131 48                   PHA
1132 A9 00                LDA    #0
1134 85 A2                STA    $A2
1136 85 C6                STA    $C6
1138 A9 20                LDA    #$20
113A C5 A2    DELPOL      CMP    $A2
113C D0 FC                BNE    DELPOL
113E A2 17                LDX    #23
1140 BD 62 11 PRINLU      LDA    FIREMSG,X
1143 29 3F                AND    #63
1145 9D 04 04             STA    1028,X
1148 CA                   DEX
1149 10 F5                BPL    PRINLU
114B 20 1A 1F LPPF        JSR    IJKL
114E AD 6B 1F             LDA    JB
1151 F0 F8                BEQ    LPPF
1153 A2 17                LDX    #23
1155 68                   PLA
1156 9D 04 04 RESTRM      STA    1028,X
1159 CA                   DEX
115A 10 FA                BPL    RESTRM
115C AD 6B 1F             LDA    JB
115F 29 10                AND    #16
1161 60                   RTS
1162 3A       FIREMSG     .BYTE  58
1163 46 49 52             .ASC   "FIREBUTTON TO CO
                                 NTINUE"
1179 3A                   .BYTE  58
;
; bkcm.control
;
; this is the main routine for selecting campaign stops and
     playing the game
117A AD 23 25 CONTROL     LDA    FIRSTCAN
117D CD 15 8F             CMP    PARTY
1180 D0 0B                BNE    RECON
```

```
; another week passes
1182 EE 24 25             INC    WEEKNO
1185 AD 24 25             LDA    WEEKNO
1188 C9 0A                CMP    #10
118A D0 01                BNE    RECON
; main rts out of campaign
118C 60                   RTS
118D 20 ED 09 RECON       JSR    MAKEMAP
1190 20 CD 15             JSR    MENUITIN
1193 20 26 0F             JSR    DOATTRIB
1196 A9 07                LDA    #7
1198 8D 00 8F             STA    DAY
;
; (main menu is on screen now
119B 20 F4 14 DEFAULT     JSR    MENUMAIN
119E A2 05    GOAGAINQ    LDX    #5
11A0 A0 0C                LDY    #12
; (cursor on screen
11A2 20 A3 1C             JSR    SELECT
11A5 AA       NQ0         TAX
11A6 D0 03                BNE    NQ1
; (campaign at 200= pick a state
11A8 4C 03 12             JMP    CAMPAIGN
11AB CA       NQ1         DEX
11AC D0 03                BNE    NQ2
; (tv ads at 400
11AE 4C 93 12             JMP    TVDAY
11B1 CA       NQ2         DEX
11B2 D0 03                BNE    NQ3
; (fundrais at line 600
11B4 4C C5 12             JMP    FUNDRAIT
11B7 CA       NQ3         DEX
11B8 D0 03                BNE    NQ4
; (rest routine at 800
11BA 4C EF 12             JMP    REST
11BD CA       NQ4         DEX
11BE D0 06                BNE    NQ5
11C0 20 2B 1D             JSR    MAPMAGIC
; (goto the map routine
11C3 4C 9B 11             JMP    DEFAULT
11C6 CA       NQ5         DEX
11C7 D0 08                BNE    NQ7
11C9 20 16 13             JSR    POLLSTER
11CC D0 CD                BNE    DEFAULT
; (poll routine at 1200
11CE 4C 03 12             JMP    CAMPAIGN
11D1 CA       NQ7         DEX
11D2 D0 0E                BNE    NQ8
11D4 20 2E 11             JSR    PRESSFIRE
11D7 F0 C5                BEQ    GOAGAINQ
11D9 AD 0B 8F RESETREG    LDA    LASTREGI
11DC 8D 20 8F             STA    CURREG
; reconsider/try again
11DF 4C 8D 11             JMP    RECON
11E2 CA       NQ8         DEX
11E3 F0 03                BEQ    NQEND
11E5 4C 9B 11             JMP    DEFAULT
; (travel routine at 1600
11E8 4C C8 13 NQEND       JMP    TRAVEL
;
; campaign
11EB A2 00    OREET       LDX    #<REGUMEN
11ED A9 80                LDA    #>REGUMEN
11EF 18                   CLC
11F0 6D 20 8F             ADC    CURREG
11F3 A8                   TAY
11F4 A9 00                LDA    #0
11F6 85 FD                STA    $FD
11F8 A9 0E                LDA    #14
11FA 85 FE                STA    $FE
```

```
11FC A9 1E                LDA   #30
11FE 85 A7                STA   $A7
; menu for current region/imply rts
1200 4C B8 14             JMP   PRINIT
1203 20 EB 11  CAMPAIGN   JSR   OREET
1206 AE 20 8F  RESLG      LDX   CURREG
1209 BD 7F 24             LDA   REGFIRST,X
120C CA                   DEX
120D 38                   SEC
120E FD 7F 24             SBC   REGFIRST,X
1211 48                   PHA
1212 69 03                ADC   #3
1214 A8                   TAY
1215 A2 03                LDX   #3
1217 20 A3 1C             JSR   SELECT
121A C9 00                CMP   #0
121C D0 07                BNE   CRR
121E 20 2B 1D             JSR   MAPMAGIC
1221 68                   PLA
1222 4C 06 12             JMP   RESLG
1225 85 02     CRR        STA   $02
1227 68                   PLA
1228 C5 02                CMP   $02
122A B0 03                BCS   CRR2
122C 4C 9B 11             JMP   DEFAULT
122F C6 02     CRR2       DEC   $02
1231 AE 20 8F             LDX   CURREG
1234 CA                   DEX
1235 BD 7F 24             LDA   REGFIRST,X
1238 18                   CLC
; state number in .a
1239 65 02                ADC   $02
123B AE 00 8F             LDX   DAY
123E 9D 00 8F             STA   DAY,X
1241 85 FB                STA   $FB
; state# & day
1243 86 FC                STX   $FC
1245 20 FA 13             JSR   ITINPLOT
1248 A9 1E                LDA   #30
124A 20 D2 FF             JSR   $FFD2
124D A5 FB                LDA   $FB
124F 0A                   ASL
1250 AA                   TAX
1251 BD DC 21             LDA   STNAME,X
; green, 1st letter
1254 20 D2 FF             JSR   $FFD2
1257 BD DD 21             LDA   STNAME+1,X
125A 20 D2 FF             JSR   $FFD2
125D A9 20                LDA   #32
; 2nd letter, space
125F 20 D2 FF             JSR   $FFD2
1262 BD 00 78             LDA   STEVOTE,X
1265 48                   PHA
1266 AA                   TAX
1267 A9 00                LDA   #0
; print # of electoral votes
1269 20 CD BD             JSR   $BDCD
126C 68                   PLA
126D C9 0A                CMP   #10
126F B0 05                BCS   CREGI
1271 A9 20                LDA   #32
1273 20 D2 FF             JSR   $FFD2
1276 A9 20     CREGI      LDA   #32
1278 20 D2 FF             JSR   $FFD2
127B A9 98                LDA   #152
127D 20 D2 FF             JSR   $FFD2
1280 AD 20 8F             LDA   CURREG
1283 09 30                ORA   #$30
1285 20 D2 FF             JSR   $FFD2
;
```

```
1288 CE 00 8F             DEC   DAY
128B D0 03                BNE   NEXIIX
;
128D 4C 07 14             JMP   EVALEND
1290 4C 06 12  NEXIIX     JMP   RESLG
;
; tv ads routine
; $f0 = tv ads
1293 AE 00 8F  TVDAY      LDX   DAY
1296 A9 F0                LDA   #$F0
1298 9D 00 8F             STA   DAY,X
129B 86 FC                STX   $FC
129D 20 FA 13             JSR   ITINPLOT
12A0 A9 81                LDA   #129
12A2 20 D2 FF             JSR   $FFD2
12A5 A2 00                LDX   #0
12A7 BD 70 15  TVROUL     LDA   REGLCL,X
12AA F0 06                BEQ   TV67
12AC 20 D2 FF             JSR   $FFD2
12AF E8                   INX
12B0 D0 F5                BNE   TVROUL
12B2 AD 20 8F  TV67       LDA   CURREG
12B5 09 30                ORA   #$30
12B7 20 D2 FF             JSR   $FFD2
12BA CE 00 8F             DEC   DAY
12BD D0 03                BNE   TV68
12BF 4C 07 14             JMP   EVALEND
12C2 4C 9B 11  TV68       JMP   DEFAULT
;
; fundrais -- adds money (beef up later)
; $ff = fundraising needed (in itinerary)
12C5 AE 00 8F  FUNDRAIT   LDX   DAY
12C8 A9 FF                LDA   #$FF
12CA 9D 00 8F             STA   DAY,X
12CD 86 FC                STX   $FC
12CF 20 FA 13             JSR   ITINPLOT
12D2 A9 9A                LDA   #154
12D4 20 D2 FF             JSR   $FFD2
12D7 A2 00                LDX   #0
12D9 BD 7D 15  FUNN6      LDA   REGLCM,X
12DC F0 06                BEQ   ENFUN6
12DE 20 D2 FF             JSR   $FFD2
12E1 E8                   INX
12E2 D0 F5                BNE   FUNN6
12E4 CE 00 8F  ENFUN6     DEC   DAY
12E7 D0 03                BNE   ENFUN7
12E9 4C 07 14             JMP   EVALEND
12EC 4C 9B 11  ENFUN7     JMP   DEFAULT
;
; rest
12EF AE 00 8F  REST       LDX   DAY
12F2 A9 00                LDA   #0
12F4 9D 00 8F             STA   DAY,X
12F7 86 FC                STX   $FC
12F9 20 FA 13             JSR   ITINPLOT
12FC A9 9B                LDA   #155
12FE 20 D2 FF             JSR   $FFD2
1301 A9 5A                LDA   #90
1303 A2 05                LDX   #5
1305 20 D2 FF  ZPLOT      JSR   $FFD2
1308 CA                   DEX
1309 D0 FA                BNE   ZPLOT
130B CE 00 8F             DEC   DAY
130E D0 03                BNE   ENRSTX
310  4C 07 14             JMP   EVALEND
; should be evalend ↑↑
1313 4C 9B 11  ENRSTX     JMP   DEFAULT
;
1316 AD 09 8F  POLLSTER   LDA   MONEY
1319 C9 0A                CMP   #10
```

```
131B B0 01                BCS   OKTOPOLL
131D 60                   RTS
131E CE 09 8F  OKTOPOLL   DEC   MONEY
; prints header
1321 20 99 21             JSR   CONSULHD
1324 AE 20 8F             LDX   CURREG
1327 BD 7F 24             LDA   REGFIRST,X
132A 85 F8                STA   Z1+1
132C CA                   DEX
132D BD 7F 24             LDA   REGFIRST,X
1330 85 F7                STA   Z1
; first & last z1/z1+1
1332 A9 96                LDA   #150
1334 85 F9                STA   Z2
1336 A9 04                LDA   #4
; z2=indirect for storing graph
1338 85 FA                STA   Z2+1
133A A9 2F     PSTLOOP1   LDA   #47
133C 85 FB                STA   Z3
; solids into z3/+1
133E 85 FC                STA   Z3+1
1340 A6 F7                LDX   Z1
1342 E4 F8                CPX   Z1+1
1344 D0 03                BNE   CNPST
; end of poll routine
1346 4C 2E 11             JMP   PRESSFIRE
1349 A5 F9     CNPST      LDA   Z2
134B 18                   CLC
134C 69 28                ADC   #40
134E 85 F9                STA   Z2
1350 90 02                BCC   CNPST1
1352 E6 FA                INC   Z2+1
1354 BD 00 90  CNPST1     LDA   REALLEAN,X
1357 85 FD                STA   Z4
1359 85 FE                STA   Z4+1
135B A2 04                LDX   #4
135D 06 FE     SMIL       ASL   Z4+1
135F CA                   DEX
1360 D0 FB                BNE   SMIL
1362 06 FE                ASL   Z4+1
1364 B0 1C                BCS   PRTDEMM
1366 A9 25                LDA   #37
1368 85 FB                STA   Z3
136A 06 FE                ASL   Z4+1
136C B0 14                BCS   PRTDEMM
136E A9 20                LDA   #32
1370 85 FB                STA   Z3
1372 06 FE                ASL   Z4+1
1374 B0 0C                BCS   PRTDEMM
1376 A9 25                LDA   #37
1378 85 FC                STA   Z3+1
137A 06 FE                ASL   Z4+1
137C B0 04                BCS   PRTDEMM
137E A9 20                LDA   #32
1380 85 FC                STA   Z3+1
1382 A0 00     PRTDEMM    LDY   #0
1384 A9 20                LDA   #32
1386 91 F9                STA   (Z2),Y
1388 C8                   INY
1389 A5 FB                LDA   Z3
138B 91 F9                STA   (Z2),Y
138D C8                   INY
138E A5 FC                LDA   Z3+1
1390 91 F9                STA   (Z2),Y
;
1392 A9 2F     PSTREPUB   LDA   #47
1394 85 FB                STA   Z3
; solid blocks into z3/+1
1396 85 FC                STA   Z3+1
1398 06 FD                ASL   Z4
```

```
139A B0 1C                BCS   ENREPUB
139C A9 2A                LDA   #42
139E 85 FC                STA   Z3+1
13A0 06 FD                ASL   Z4
13A2 B0 14                BCS   ENREPUB
13A4 A9 20                LDA   #32
13A6 85 FC                STA   Z3+1
13A8 06 FD                ASL   Z4
13AA B0 0C                BCS   ENREPUB
13AC A9 2A                LDA   #42
13AE 85 FB                STA   Z3
13B0 06 FD                ASL   Z4
13B2 B0 04                BCS   ENREPUB
13B4 A9 20                LDA   #32
13B6 85 FB                STA   Z3
13B8 A0 07     ENREPUB    LDY   #7
13BA A5 FB                LDA   Z3
13BC 91 F9                STA   (Z2),Y
13BE C8                   INY
13BF A5 FC                LDA   Z3+1
13C1 91 F9                STA   (Z2),Y
13C3 E6 F7                INC   Z1
13C5 4C 3A 13             JMP   PSTLOOP1
;
; ** travel
; regional main menu on screen
13C8 20 67 17  TRAVEL     JSR   REGLARMEN
13CB AE 20 8F             LDX   CURREG
13CE E8                   INX
13CF E8                   INX
13D0 E8                   INX
13D1 A0 1F                LDY   #31
13D3 18                   CLC
13D4 20 F0 FF             JSR   $FFF0
13D7 A9 3A                LDA   #$3A
13D9 20 D2 FF             JSR   $FFD2
13DC A2 03                LDX   #3
13DE A0 0D                LDY   #13
13E0 20 A3 1C             JSR   SELECT
13E3 C9 00                CMP   #0
13E5 D0 06                BNE   TRA
13E7 20 2B 1D             JSR   MAPMAGIC
13EA 4C C8 13             JMP   TRAVEL
13ED C9 0A     TRA        CMP   #10
13EF D0 03                BNE   GOREGG
13F1 4C 9B 11             JMP   DEFAULT
13F4 8D 20 8F  GOREGG     STA   CURREG
13F7 4C 9B 11             JMP   DEFAULT
;
13FA A9 16     ITINPLOT   LDA   #22
13FC 38                   SEC
13FD E5 FC                SBC   $FC
13FF AA                   TAX
1400 A0 20                LDY   #32
1402 18                   CLC
1403 20 F0 FF             JSR   $FFF0
1406 60                   RTS
; evalend ** after week is done, evaluate results
1407 20 2E 11  EVALEND    JSR   PRESSFIRE
140A D0 03                BNE   EVALUIT
140C 4C D9 11             JMP   RESETREG
140F 20 68 19  EVALUIT    JSR   EVALVOTES
1412 20 FA 1A             JSR   CKDEMREP
1415 20 6C 1B             JSR   TRANSLEANCOLR
; toggle the party...
1418 20 84 1B             JSR   TGLPARTY
; and jump back to the beginning of this .
141B 4C 7A 11             JMP   CONTROL
;
; bkcm.menu
```

```
;
141E A9 00     SETUPMENU  LDA    #0
; *first line of text
1420 85 FD                STA    $FD
1422 A9 0A                LDA    #10
; *last line
1424 85 FE                STA    $FE
1426 A9 1E                LDA    #30
; *column number
1428 85 A7                STA    $A7
142A A2 31                LDX    #<MENU1
142C A0 14                LDY    #>MENU1
142E 4C B8 14             JMP    PRINIT
;
1431 FF        MENU1      .BYTE  255
1432 12 90 A0             .AS    "{RVS}{BLK}{SHIF
                                 T-SPACE}USE{SHIF
                                 T-SPACE}JOY [<X
                                 >]{SHIFT-SPACE}"
143E 00                   .BYTE  0
143F FF                   .BYTE  255
1440 A0 D3 D4             .ASC   "{SHIFT-SPACE}STI
                                 CK{SHIFT-SPACE}O
                                 R{SHIFT-SPACE}"
144A 00                   .BYTE  0
144B FF                   .BYTE  255
144C C9 CA CB             .ASC   "IJKL{OFF}M{RVS}
                                 {SHIFT-SPACE}KEY
                                 S"
1458 00                   .BYTE  0
1459 FF                   .BYTE  255
145A 12 9B 50             .ASC   "{RVS} [<8>]PL[
                                 1 PARTY{OFF}"
1467 00                   .BYTE  0
1468 FF        PARNAME0   .BYTE  255
1469 1F 44 45             .ASC   "{BLU}DEMOCRATI
                                 C"
1474 00                   .BYTE  0
1475 FF        PARNAME1   .BYTE  255
1476 52 45 50             .ASC   "REPUBLICAN"
1480 00                   .BYTE  0
1481 FF                   .BYTE  255
1482 12 9B 20             .ASC   "{RVS} [<8>] INCU
                                 MBENT{OFF}"
148F 00                   .BYTE  0
1490 FF                   .BYTE  255
1491 1F 20 20             .ASC   "{BLU} NONE "
149C 00                   .BYTE  0
149D FF                   .BYTE  255
149E 20 50 4C  P1NAME     .ASC   " PLAYER 1 "
14A8 00 FF                .BYTE  0,255
14AA 20 50 4C  P2NAME     .ASC   " PLAYER 2 "
14B4 00 00 00             .BYTE  0,0,0,0
;
; ** prinit sr
14B8 86 FB     PRINIT     STX    $FB
; (lsb/msb of text address)
14BA 84 FC                STY    $FC
14BC D0 0B                BNE    MNPRIN
14BE C8        RESETMPTR  INY
14BF 98                   TYA
14C0 18                   CLC
14C1 65 FB                ADC    $FB
14C3 85 FB                STA    $FB
14C5 90 02                BCC    MNPRIN
14C7 E6 FC                INC    $FC
14C9 A6 FD     MNPRIN     LDX    $FD
14CB E4 FE                CPX    $FE
14CD D0 01                BNE    CNPRIN
; main rts of prinit
```

```
14CF 60                   RTS
14D0 E6 FD     CNPRIN     INC    $FD
; column #
14D2 A4 A7                LDY    $A7
14D4 18                   CLC
14D5 20 F0 FF             JSR    $FFF0
14D8 A0 00                LDY    #0
; kernal plot
14DA A2 FF                LDX    #$FF
14DC B1 FB                LDA    ($FB),Y
14DE 10 10                BPL    REPT
14E0 C8                   INY
14E1 B1 FB     JUMPIN     LDA    ($FB),Y
14E3 F0 D9                BEQ    RESETMPTR
14E5 20 D2 FF  PRNT       JSR    $FFD2
14E8 CA                   DEX
14E9 10 FA                BPL    PRNT
14EB F0 D1                BEQ    RESETMPTR
14ED C8                   INY
14EE D0 F1                BNE    JUMPIN
14F0 AA        REPT       TAX
14F1 C8                   INY
14F2 D0 ED                BNE    JUMPIN
; ** menumain
14F4 A9 00     MENUMAIN   LDA    #0
14F6 85 FD                STA    $FD
14F8 A9 0E                LDA    #14
; first and last lines
14FA 85 FE                STA    $FE
14FC A9 92                LDA    #146
14FE 20 D2 FF             JSR    $FFD2
1501 A9 90                LDA    #144
; set unreverse & black chr color
1503 20 D2 FF             JSR    $FFD2
1506 A9 1E                LDA    #30
1508 85 A7                STA    $A7
150A A2 48                LDX    #<MENUM
150C A0 15                LDY    #>MENUM
150E 20 B8 14             JSR    PRINIT
1511 AE 15 8F             LDX    PARTY
1514 BD 25 25             LDA    PRTIS,X
1517 29 3F                AND    #63
1519 8D 47 04             STA    1095
151C AD 24 25             LDA    WEEKNO
151F 09 30                ORA    #$30
1521 8D 4E 04             STA    1102
; party (d or r) and week number
1524 AD 20 8F             LDA    CURREG
1527 0A                   ASL
1528 0A                   ASL
1529 0A                   ASL
152A 18                   CLC
152B 6D 20 8F             ADC    CURREG
152E AA                   TAX
152F AD 20 8F             LDA    CURREG
1532 09 30                ORA    #$30
1534 8D 96 04             STA    1174
1537 A0 00                LDY    #0
1539 BD 25 24  RGGGLOP    LDA    REGIONAME,X
153C 29 3F                AND    #63
153E F0 07                BEQ    OUTRR
1540 99 98 04             STA    1176,Y
1543 E8                   INX
1544 C8                   INY
1545 D0 F2                BNE    RGGGLOP
1547 60        OUTRR      RTS
1548 09 3A 00  MENUM      .BYTE  9,58,0
154B FF                   .BYTE  255
154C 20 20 20             .ASC   " WEEK "
1556 00 09 3A             .BYTE  0,9,58,0
```

```
155A 09 20 00              .BYTE  9,32,0
155D 09 20 00              .BYTE  9,32,0
1560 FF                    .BYTE  255
1561 1F 20 20              .ASC   "{BLU} CAMPAIGN
156C 00 FF 20              .BYTE  0,255,32,32
1570 54 56 20  REGLCL      .ASC   "TV ADS "
1578 9D 00 FF              .BYTE  157,0,255,32,32
157D 46 55 4E  REGLCM      .ASC   "FUNDRAIS"
1585 00 FF                 .BYTE  0,255
1587 20 20 52              .ASC   " REST "
1591 00 FF                 .BYTE  0,255
1593 12 9B 20              .ASC   "{RVS} [<8>] MA
                                  P "
159F 00 FF                 .BYTE  0,255
15A1 20 50 4F              .ASC   " POLL "
15AB 00 FF                 .BYTE  0,255
15AD 92 96 52              .ASC   "{OFF} [<3>]RECO
                                  NSIDER"
15B9 00 FF                 .BYTE  0,255
15BB 54 52 41              .ASC   "TRAVEL [<7>]"
15C6 00 09 20              .BYTE  0,9,32,0,0,0,0
;
; ** menuitin/the itinerary
15CD A9 0E     MENUITIN    LDA    #14
15CF 85 FD                 STA    $FD
15D1 A9 19                 LDA    #25
15D3 85 FE                 STA    $FE
15D5 A9 1C                 LDA    #28
15D7 85 A7                 STA    $A7
15D9 A9 1E                 LDA    #30
; green
15DB 20 D2 FF              JSR    $FFD2
15DE A2 22                 LDX    #<ITINT
15E0 A0 17                 LDY    #>ITINT
15E2 20 B8 14              JSR    PRINIT
15E5 A9 2B                 LDA    #43
15E7 8D 4C 06              STA    1612
15EA A9 2E                 LDA    #46
; screen locations/corners
15EC 8D 8C 07              STA    1932
15EF AD 24 25              LDA    WEEKNO
15F2 D0 0B                 BNE    GRTON
15F4 A9 14                 LDA    #20
15F6 A2 08                 LDX    #8
15F8 20 D2 FF  OUNT        JSR    $FFD2
15FB CA                    DEX
15FC D0 FA                 BNE    OUNT
15FE 60                    RTS
15FF AE 08 8F  GRTON       LDX    HEALTH
1602 A9 00                 LDA    #0
; print the health
1604 20 CD BD              JSR    $BDCD
1607 A2 17                 LDX    #23
1609 A0 1E                 LDY    #30
160B 18                    CLC
160C 20 F0 FF              JSR    $FFF0
160F A9 1E                 LDA    #30
; green
1611 20 D2 FF              JSR    $FFD2
1614 AE 09 8F              LDX    MONEY
1617 E0 64                 CPX    #100
1619 B0 16                 BCS    MNPR9
161B A9 20                 LDA    #32
161D 20 D2 FF              JSR    $FFD2
1620 E0 0A                 CPX    #10
1622 B0 0D                 BCS    MNPR9
1624 20 D2 FF              JSR    $FFD2
1627 20 D2 FF              JSR    $FFD2
162A 8A                    TXA
162B 09 30                 ORA    #$30
```

```
162D 20 D2 FF              JSR    $FFD2
1630 60                    RTS
1631 A9 00     MNPR9       LDA    #0
1633 20 CD BD              JSR    $BDCD
1636 AD B8 07              LDA    1976
; add one for comma
1639 8D B9 07              STA    1977
163C A2 06                 LDX    #6
163E AD 1B D4  RNLOOPTP    LDA    RAND
1641 29 0F                 AND    #$0F
1643 C9 0A                 CMP    #10
1645 B0 F7                 BCS    RNLOOPTP
1647 09 30                 ORA    #$30
1649 9D B9 07              STA    1977,X
164C CA                    DEX
; prints random #s in money slot
164D D0 EF                 BNE    RNLOOPTP
164F A9 3C                 LDA    #60
1651 8D B8 07              STA    1976
; put commas back in
1654 8D BC 07              STA    1980
; barchart -- part of itin, but also works standalone at end of
 game
1657 20 7A 16  BARCHART    JSR    BAR1
165A A9 34                 LDA    #52
165C 85 F8                 STA    $F8
165E A9 00                 LDA    #0
1660 85 F7                 STA    $F7
1662 A8                    TAY
1663 A2 02                 LDX    #2
1665 95 FC     FLBARC1     STA    $FC,X
1667 CA                    DEX
1668 10 FB                 BPL    FLBARC1
166A 20 A6 16              JSR    BAR2
166D A9 20                 LDA    #32
166F A2 02                 LDX    #2
1671 95 F9     FLBARC      STA    $F9,X
1673 CA                    DEX
1674 10 FB                 BPL    FLBARC
1676 20 D3 16              JSR    BAR3
1679 60                    RTS
167A A9 11     BAR1        LDA    #17
167C 85 FC                 STA    $FC
167E A9 00                 LDA    #0
1680 85 FB                 STA    $FB
1682 A6 FB     ENBAR1      LDX    $FB
1684 E4 FC                 CPX    $FC
1686 D0 01                 BNE    CNBAR1
1688 60                    RTS
1689 A0 00     CNBAR1      LDY    #0
168B 18                    CLC
168C 20 F0 FF              JSR    $FFF0
168F A2 00                 LDX    #0
1691 BD 28 25  LPBAR1      LDA    PRTCO,X
1694 20 D2 FF              JSR    $FFD2
1697 A9 25                 LDA    #37
; bar & color
1699 20 D2 FF              JSR    $FFD2
169C E8                    INX
169D E0 03                 CPX    #3
169F D0 F0                 BNE    LPBAR1
16A1 E6 FB                 INC    $FB
16A3 4C 82 16              JMP    ENBAR1
16A6 A6 F7     BAR2        LDX    $F7
16A8 E8                    INX
16A9 E8                    INX
16AA 86 F7                 STX    $F7
16AC C8                    INY
16AD C4 F8                 CPY    $F8
16AF D0 01                 BNE    ENBAR2
```

```
; main rts
16B1 60                    RTS
16B2 BD 00 78  ENBAR2      LDA   STEVOTE,X
16B5 4A                    LSR
16B6 85 02                 STA   $02
16B8 B9 00 90              LDA   REALLEAN,Y
16BB A2 02                 LDX   #2
16BD 29 EE                 AND   #$EE
16BF F0 06                 BEQ   STBAR2
16C1 CA                    DEX
16C2 29 0E                 AND   #$0E
16C4 F0 01                 BEQ   STBAR2
16C6 CA                    DEX
16C7 B5 FC     STBAR2      LDA   $FC,X
16C9 18                    CLC
16CA 65 02                 ADC   $02
16CC B0 D8                 BCS   BAR2
16CE 95 FC                 STA   $FC,X
16D0 4C A6 16              JMP   BAR2
16D3 A9 04     BAR3        LDA   #4
16D5 85 F8                 STA   $F8
16D7 A9 00                 LDA   #0
16D9 85 F7                 STA   $F7
16DB A0 02                 LDY   #2
16DD A9 0F                 LDA   #15
16DF 85 02                 STA   $02
16E1 B9 25 25  LPHBAR3     LDA   PRTIS,Y
16E4 29 3F                 AND   #63
16E6 91 F7                 STA   ($F7),Y
16E8 88                    DEY
; d*r*u header
16E9 10 F6                 BPL   LPHBAR3
16EB A9 28     ENBAR3      LDA   #40
16ED 18                    CLC
16EE 65 F7                 ADC   $F7
16F0 85 F7                 STA   $F7
16F2 A9 00                 LDA   #0
16F4 65 F8                 ADC   $F8
16F6 85 F8                 STA   $F8
16F8 A0 02                 LDY   #2
16FA A2 02                 LDX   #2
16FC A9 10     CKBAR3      LDA   #16
16FE 18                    CLC
16FF 75 FC                 ADC   $FC,X
1701 95 FC                 STA   $FC,X
1703 B0 0D                 BCS   FLIPBAR3
1705 B5 F9                 LDA   $F9,X
1707 91 F7     BABAR3      STA   ($F7),Y
1709 CA                    DEX
170A 88                    DEY
170B 10 EF                 BPL   CKBAR3
170D C6 02                 DEC   $02
170F 10 DA                 BPL   ENBAR3
; main rts
1711 60                    RTS
1712 48        FLIPBAR3    PHA
1713 A9 25                 LDA   #37
1715 95 F9                 STA   $F9,X
1717 68                    PLA
1718 4A                    LSR
1719 4A                    LSR
171A 4A                    LSR
171B 29 01                 AND   #1
171D 09 24                 ORA   #36
171F 4C 07 17              JMP   BABAR3
;
1722 0B 23 00  ITINT       .BYTE 11,35,0
1725 FF 2A                 .BYTE 255,42
1727 20 53                 .ASC  " S"
1729 00 FF 2A              .BYTE 0,255,42
```

```
172C 20 4D                     .ASC  " M"
172E 00 FF 2A                  .BYTE 0,255,42
1731 20 54                     .ASC  " T"
1733 00 FF 2A                  .BYTE 0,255,42
1736 20 57                     .ASC  " W"
1738 00 FF 2A                  .BYTE 0,255,42
173B 20 54                     .ASC  " T"
173D 00 FF 2A                  .BYTE 0,255,42
1740 20 46                     .ASC  " F"
1742 00 FF 2A                  .BYTE 0,255,42
1745 20 53                     .ASC  " S"
1747 00 0B 2C                  .BYTE 0,11,44,0
174B FF                        .BYTE 255
174C 5C 20 30                  .ASC  "£ 00<000<000"
1758 00 FF                     .BYTE 0,255
175A 9A 20 48                  .ASC  " [<7>] HEALTH"
1762 20 00 00                  .BYTE 32,0,0,0,0
1767 A9 1C     REGLARMEN       LDA   #28
1769 20 D2 FF                  JSR   .$FFD2
176C A9 00                     LDA   #0
176E 85 FD                     STA   $FD
1770 A9 0F                     LDA   #15
; red & pointers
1772 85 FE                     STA   $FE
1774 A9 1E                     LDA   #30
1776 85 A7                     STA   $A7
1778 A2 00                     LDX   #<REGUMEN
177A A0 80                     LDY   #>REGUMEN
177C 20 B8 14                  JSR   PRINIT
177F 60                        RTS
;
; ** setregmenu
; sets up menu tables
; ** z1 = indirect for tables at 8000
; ** z2 = x counter/to regioname
; ** z2+1 = compare byte for end of region
; ** z3 = number of e-votes (begin with $30 = zero)
; ** z4 = region #
; ** z4+1 = state#
1780 A9 80     SETREGMEN       LDA   #>REGUMEN
1782 85 F8                     STA   Z1+1
1784 A9 00                     LDA   #<REGUMEN
1786 85 F7                     STA   Z1
1788 A9 00                     LDA   #0
178A 85 FA                     STA   Z2+1
178C 85 F9                     STA   Z2
178E A9 00                     LDA   #0
1790 85 FD                     STA   Z4
1792 85 FE                     STA   Z4+1
1794 A8                        TAY
1795 A2 01                     LDX   #1
; header for regional
1797 20 F1 17                  JSR   RHLOP
179A A2 00                     LDX   #0
179C A0 04                     LDY   #4
179E BD 25 24  RHDR            LDA   REGIONAME,X
17A1 91 F7                     STA   (Z1),Y
17A3 C8                        INY
17A4 E8                        INX
17A5 E0 08                     CPX   #8
17A7 D0 F5                     BNE   RHDR
17A9 A2 08                     LDX   #8
17AB AC 36 19                  LDY   RHTABL
17AE 88                        DEY
17AF E6 FD     NINELX          INC   Z4
17B1 A5 FD                     LDA   Z4
17B3 C9 0A                     CMP   #10
17B5 D0 03                     BNE   GOAHX
17B7 4C E1 17                  JMP   RETMENU
17BA A9 FF     GOAHX           LDA   #255
```

```
17BC 91 F7                  STA   (Z1),Y
17BE C8                     INY
17BF A9 1C                  LDA   #28
17C1 91 F7                  STA   (Z1),Y
; red
17C3 C8                     INY
17C4 A9 2A                  LDA   #42
17C6 91 F7                  STA   (Z1),Y
; left bar (red)
17C8 C8                     INY
17C9 A5 FD                  LDA   Z4
17CB 09 30                  ORA   #$30
17CD 91 F7                  STA   (Z1),Y
; region number
17CF C8                     INY
17D0 A9 1F                  LDA   #31
; now blue
17D2 91 F7                  STA   (Z1),Y
17D4 C8       LUPLO         INY
17D5 E8                     INX
17D6 BD 25 24               LDA   REGIONAME,X
17D9 91 F7                  STA   (Z1),Y
17DB D0 F7                  BNE   LUPLO
17DD C8                     INY
17DE 4C AF 17               JMP   NINELX
17E1 20 02 18 RETMENU       JSR   REMTIN
17E4 A9 00                  LDA   #0
17E6 A2 04                  LDX   #4
17E8 91 F7    SIMCOM        STA   (Z1),Y
17EA C8                     INY
17EB CA                     DEX
17EC D0 FA                  BNE   SIMCOM
; done with header, do the regions
17EE 4C 11 18               JMP   INDIVREGS
17F1 A2 01    RHLOP         LDX   #1
17F3 A0 00                  LDY   #0
17F5 BD 36 19 RHSOP         LDA   RHTABL,X
17F8 91 F7                  STA   (Z1),Y
17FA C8                     INY
17FB E8                     INX
17FC EC 36 19               CPX   RHTABL
17FF D0 F4                  BNE   RHSOP
1801 60                     RTS
; sets up word 'menu' at end
1802 A2 01    REMTIN        LDX   #1
1804 BD 59 19 LMNPP         LDA   MENWRD,X
1807 91 F7                  STA   (Z1),Y
1809 C8                     INY
180A E8                     INX
180B EC 59 19               CPX   MENWRD
180E D0 F4                  BNE   LMNPP
1810 60                     RTS
; indivregs do the individual regions
1811 A9 01    INDIVREGS     LDA   #1
1813 85 FD                  STA   Z4
1815 85 FE                  STA   Z4+1
1817 D0 09                  BNE   EREGT
1819 E6 FD    ENTRREG       INC   Z4
181B A5 FD                  LDA   Z4
181D C9 0A                  CMP   #10
181F D0 01                  BNE   EREGT
; ** main rts
1821 60                     RTS
1822 E6 F8    EREGT         INC   Z1+1
1824 A9 09                  LDA   #9
1826 18                     CLC
1827 65 F9                  ADC   Z2
1829 85 F9                  STA   Z2
182B A9 00                  LDA   #0
182D 65 FA                  ADC   Z2+1
```

```
182F 85 FA                  STA   Z2+1
1831 20 F1 17               JSR   RHLOP
1834 A6 F9                  LDX   Z2
1836 A0 04                  LDY   #4
1838 A5 FD                  LDA   Z4
183A 49 30                  EOR   #$30
183C 91 F7                  STA   (Z1),Y
183E C8                     INY
183F C8                     INY
1840 BD 25 24 RELOGP        LDA   REGIONAME,X
1843 F0 06                  BEQ   ORELF
1845 91 F7                  STA   (Z1),Y
1847 E8                     INX
1848 C8                     INY
1849 D0 F5                  BNE   RELOGP
184B A6 FD    ORELF         LDX   Z4
184D BD 7F 24               LDA   REGFIRST,X
1850 85 FA                  STA   Z2+1
1852 AC 36 19               LDY   RHTABL
1855 88                     DEY
1856 A5 FE    NEXSTTT       LDA   Z4+1
1858 0A                     ASL
1859 AA                     TAX
185A A9 30                  LDA   #$30
185C 85 FB                  STA   Z3
185E 85 FC                  STA   Z3+1
1860 A9 FF                  LDA   #255
1862 91 F7                  STA   (Z1),Y
; begin first state
1864 C8                     INY
1865 A9 1C                  LDA   #28
1867 91 F7                  STA   (Z1),Y
; red
1869 C8                     INY
186A A9 2A                  LDA   #42
186C 91 F7                  STA   (Z1),Y
; graphics chr
186E C8                     INY
186F A9 9A                  LDA   #154
1871 91 F7                  STA   (Z1),Y
; light blue
1873 C8                     INY
1874 A5 FE                  LDA   Z4+1
1876 C9 0A    BACKOR        CMP   #10
1878 90 07                  BCC   LSBSTN
187A E6 FC                  INC   Z3+1
187C E9 0A                  SBC   #10
187E 4C 76 18               JMP   BACKOR
1881 65 FB    LSBSTN        ADC   Z3
1883 85 FB                  STA   Z3
1885 A5 FC                  LDA   Z3+1
1887 91 F7                  STA   (Z1),Y
1889 C8                     INY
188A A5 FB                  LDA   Z3
188C 91 F7                  STA   (Z1),Y
; state number
188E C8                     INY
188F A9 20                  LDA   #32
1891 91 F7                  STA   (Z1),Y
1893 C8                     INY
1894 A9 97                  LDA   #151
1896 91 F7                  STA   (Z1),Y
; gray 1
1898 C8                     INY
1899 BD DC 21               LDA   STNAME,X
189C 91 F7                  STA   (Z1),Y
189E C8                     INY
189F E8                     INX
18A0 BD DC 21               LDA   STNAME,X
18A3 91 F7                  STA   (Z1),Y
```

```
18A5 C8                    INY
; 2-byte name of state
18A6 CA                    DEX
18A7 A9 20                 LDA   #32
18A9 91 F7                 STA   (Z1),Y
18AB C8                    INY
18AC A9 30                 LDA   #$30
18AE 85 FB                 STA   Z3
; reset z3 for e votes
18B0 85 FC                 STA   Z3+1
18B2 BD 00 78              LDA   STEVOTE,X
18B5 C9 0A     BACKOP      CMP   #10
18B7 90 07                 BCC   LSBSTE
18B9 E6 FC                 INC   Z3+1
18BB E9 0A                 SBC   #10
18BD 4C B5 18              JMP   BACKOP
18C0 65 FB     LSBSTE      ADC   Z3
18C2 85 FB                 STA   Z3
18C4 A5 FC                 LDA   Z3+1
18C6 91 F7                 STA   (Z1),Y
18C8 C8                    INY
18C9 A5 FB                 LDA   Z3
18CB 91 F7                 STA   (Z1),Y
; electoral votes
18CD C8                    INY
18CE A9 20                 LDA   #32
18D0 91 F7                 STA   (Z1),Y
18D2 C8                    INY
18D3 A9 00                 LDA   #0
18D5 91 F7                 STA   (Z1),Y
18D7 C8                    INY
;
; end of pass for this state
18D8 E6 FE                 INC   Z4+1
18DA A5 FE                 LDA   Z4+1
18DC C5 FA                 CMP   Z2+1
18DE F0 03                 BEQ   JNUUS
18E0 4C 56 18              JMP   NEXSTTT
;
; fill out w/spaces & do next region
18E3 20 02 18  JNUUS       JSR   REMTIN
18E6 A5 FA                 LDA   Z2+1
18E8 85 FE                 STA   Z4+1
18EA A6 FD                 LDX   Z4
18EC CA                    DEX
18ED BD 7F 24              LDA   REGFIRST,X
18F0 85 02                 STA   $02
18F2 E8                    INX
18F3 BD 7F 24              LDA   REGFIRST,X
18F6 38                    SEC
18F7 E5 02                 SBC   $02
18F9 85 02                 STA   $02
18FB A9 08                 LDA   #8
18FD E5 02                 SBC   $02
18FF 85 02                 STA   $02
1901 30 26                 BMI   EXRAT
1903 A9 09                 LDA   #9
1905 91 F7                 STA   (Z1),Y
1907 C8                    INY
1908 A9 23                 LDA   #35
190A 91 F7                 STA   (Z1),Y
190C C8                    INY
190D A9 00                 LDA   #0
190F 91 F7                 STA   (Z1),Y
1911 C8                    INY
1912 C6 02                 DEC   $02
1914 30 13                 BMI   EXRAT
1916 A9 09     FILIUV      LDA   #9
1918 91 F7                 STA   (Z1),Y
191A C8                    INY
```

```
191B A9 20                 LDA   #32
191D 91 F7                 STA   (Z1),Y
191F C8                    INY
1920 A9 00                 LDA   #0
1922 91 F7                 STA   (Z1),Y
1924 C8                    INY
1925 C6 02                 DEC   $02
1927 10 ED                 BPL   FILIUV
1929 A9 00     EXRAT       LDA   #0
192B A2 04                 LDX   #4
192D 91 F7     EXRLPP      STA   (Z1),Y
192F C8                    INY
1930 CA                    DEX
1931 D0 FA                 BNE   EXRLPP
; go back
1933 4C 19 18              JMP   ENTRREG
1936 23 09 23  RHTABL      .BYTE 35,9,35,0,255
193B 20 20 20              .BYTE 32,32,32,32,32
1940 20 20 20              .BYTE 32,32,32,32,32
1945 00 09 2C              .BYTE 0,9,44,0,255
194A 12 9A 25              .ASC  "{RVS} [<7>]%{BL
                                 K}MAP
1951 A0 A0 A0              .BYTE 160,160,160,160
1955 A0 A0 92              .BYTE 160,160,146,0
1959 0E FF     MENWRD      .BYTE 14,255
195B 1C 2A 4D              .ASC  "{RED}*MENU
1961 20 20 20              .BYTE 32,32,32,32,32,0,0
;
; bkcm.mcamp
;
1968           REALLEAN    =     $9000
; evalvotes -- enter after setting up itinerary
1968 AD 0B 8F  EVALVOTES   LDA   LASTREGI
196B 8D 20 8F              STA   CURREG
196E A9 08                 LDA   #8
1970 8D 00 8F              STA   DAY
1973 CE 00 8F  NEXDAY      DEC   DAY
1976 D0 01                 BNE   CNEVAL
; main rts
1978 60                    RTS
1979 AE 00 8F  CNEVAL      LDX   DAY
197C BD 00 8F              LDA   DAY,X
; if zero then its a restday
197F D0 09                 BNE   CKMONEY
; -- check # of states and store in zp $02:
1981 20 B1 1A              JSR   CKNOSTATES
; -- no campaigning, recharge batteries
1984 20 D0 1A              JSR   RESTDAY
1987 4C 73 19              JMP   NEXDAY
;
198A 10 17     CKMONEY     BPL   CKMDOL
198C 6A                    ROR
198D B0 03                 BCS   RAISBUCKS
198F 4C A6 1B              JMP   TELEVISI
1992 AD 18 8F  RAISBUCKS   LDA   FUNDR
1995 0A                    ASL
1996 6D 09 8F              ADC   MONEY
1999 90 02                 BCC   OVEROM
199B A9 FF                 LDA   #$FF
199D 8D 09 8F  OVEROM      STA   MONEY
19A0 4C 73 19              JMP   NEXDAY
19A3 AC 09 8F  CKMDOL      LDY   MONEY
; no money paralyzes campaign
19A6 F0 CB                 BEQ   NEXDAY
19A8 48                    PHA
; check for travel
19A9 A2 00                 LDX   #0
19AB E8        REGL99      INX
19AC DD 7F 24              CMP   REGFIRST,X
19AF B0 FA                 BCS   REGL99
```

```
19B1 EC 0B 8F             CPX   LASTREGI
; check region
19B4 F0 09                BEQ   PULCON
19B6 8E 20 8F             STX   CURREG
19B9 8E 0B 8F             STX   LASTREGI
19BC CE 09 8F             DEC   MONEY
19BF 68        PULCON     PLA
19C0 20 CF 19             JSR   SETZPP
; regular campaign day
19C3 20 E9 19             JSR   WASHUP
19C6 20 5D 1A             JSR   VOTECOUNT
19C9 20 8C 1A             JSR   SUBHNM
; addvotes
19CC 4C 73 19             JMP   NEXDAY
;
; begin main routine for campaign in state results
; accumulator holds state #
19CF 85 02     SETZPP     STA   $02
19D1 85 FB                STA   $FB
19D3 C6 FB                DEC   $FB
19D5 A5 FB                LDA   $FB
19D7 0A                   ASL
19D8 0A                   ASL
19D9 18                   CLC
19DA 65 FB                ADC   $FB
19DC 85 FB                STA   $FB
19DE 85 FD                STA   $FD
19E0 A9 92                LDA   #>STPVOTES
19E2 85 FC                STA   $FC
19E4 A9 91                LDA   #>STISSUES
19E6 85 FE                STA   $FE
; zp pointers are set
19E8 60                   RTS
;
; 1 check health
19E9 AD 08 8F  WASHUP     LDA   HEALTH
19EC 29 F8                AND   #$F8
19EE D0 05                BNE   CNLEAN
19F0 A9 01                LDA   #1
19F2 85 FF                STA   $FF
; if health < 8 then no go
19F4 60                   RTS
;
; 2 check leaning
19F5 A9 03     CNLEAN     LDA   #3
19F7 18                   CLC
19F8 6D 15 8F             ADC   PARTY
19FB A8                   TAY
19FC B1 FB                LDA   ($FB),Y
; first add basic votes
19FE 85 FF                STA   $FF
;
; 2 1/2 check for fish and visitors (3rd day doesn't count)
1A00 AD 0A 8F  VISITORS   LDA   LASTSTATE
1A03 10 03                BPL   NOTFISH
1A05 E6 FF                INC   $FF
1A07 60                   RTS
1A08 C5 02     NOTFISH    CMP   $02
1A0A D0 09                BNE   BIBULO
1A0C A9 02                LDA   #2
1A0E 20 57 1A             JSR   MORVOTES
1A11 A9 FF                LDA   #$FF
1A13 85 02                STA   $02
;
; 3 32 health = +1
1A15 A5 02     BIBULO     LDA   $02
1A17 8D 0A 8F             STA   LASTSTATE
1A1A AD 08 8F             LDA   HEALTH
1A1D A0 05                LDY   #5
1A1F 4A        HLTLOOP    LSR
```

```
1A20 88                   DEY
1A21 D0 FC                BNE   HLTLOOP
1A23 20 57 1A             JSR   MORVOTES
; 4 campaign effectiveness
1A26 AD 16 8F  DCDCFF     LDA   CAMPEFF
1A29 20 57 1A             JSR   MORVOTES
; 5 check five issues (indirect from fd/fe)
1A2C A0 06     ISSCHECK   LDY   #6
1A2E 88        CK5LOOP    DEY
1A2F D0 01                BNE   GCANDISSUE
; main rts
1A31 60                   RTS
1A32 B9 1A 8F  GCANDISSU  LDA   LASTWK,Y
1A35 D1 FD                CMP   ($FD),Y
1A37 D0 07                BNE   UPDOWN
1A39 A9 03                LDA   #3
1A3B 20 57 1A             JSR   MORVOTES
1A3E D0 EE                BNE   CK5LOOP
1A40 AA        UPDOWN     TAX
1A41 CA                   DEX
1A42 8A                   TXA
1A43 D1 FD                CMP   ($FD),Y
1A45 D0 07                BNE   DOWNUP
1A47 A9 01     ONLYONE    LDA   #1
1A49 20 57 1A             JSR   MORVOTES
1A4C D0 E0                BNE   CK5LOOP
1A4E E8        DOWNUP     INX
1A4F E8                   INX
1A50 8A                   TXA
1A51 D1 FD                CMP   ($FD),Y
1A53 D0 D9                BNE   CK5LOOP
1A55 F0 F0                BEQ   ONLYONE
;
; morvotes -- adds .a to $ff (counter of votes won)
1A57 18        MORVOTES   CLC
1A58 65 FF                ADC   $FF
1A5A 85 FF                STA   $FF
1A5C 60                   RTS
;
; votecount (makes votes won permanent in state vote count)
1A5D A0 05     VOTECOUNT  LDY   #5
1A5F B1 FB                LDA   ($FB),Y
1A61 38                   SEC
;check on votes in pool
1A62 E5 FF                SBC   $FF
1A64 B0 04                BCS   SUBTRPOOL
1A66 C6 FF                DEC   $FF
; subtract one, go back
1A68 D0 F3                BNE   VOTECOUNT
1A6A 91 FB     SUBTRPOOL  STA   ($FB),Y
1A6C A5 FF                LDA   $FF
1A6E AA                   TAX
1A6F AC 15 8F             LDY   PARTY
1A72 C8                   INY
1A73 18                   CLC
1A74 71 FB                ADC   ($FB),Y
1A76 90 02                BCC   SBPOL
1A78 A9 FF                LDA   #$FF
; add to party votes (in whatever state)
1A7A 91 FB     SBPOL      STA   ($FB),Y
1A7C 98                   TYA
1A7D 49 03                EOR   #3
1A7F A8                   TAY
1A80 8A        OTHERPART  TXA
1A81 4A                   LSR
1A82 4A                   LSR
1A83 71 FB                ADC   ($FB),Y
1A85 90 02                BCC   QUARTERDO
1A87 A9 FF                LDA   #$FF
1A89 91 FB     QUARTERDO  STA   ($FB),Y
```

```
; main rts
1A8B 60                     RTS
;
; subtract some health and money
1A8C 46 FF     SUBHNM       LSR   $FF
1A8E D0 01                  BNE   HLAHD
1A90 60                     RTS
1A91 AD 08 8F  HLAHD        LDA   HEALTH
1A94 38                     SEC
1A95 E5 FF                  SBC   $FF
1A97 B0 02                  BCS   LESSHEALTH
1A99 A9 00                  LDA   #0
1A9B 8D 08 8F  LESSHEALT    STA   HEALTH
1A9E 46 FF                  LSR   $FF
1AA0 D0 01                  BNE   MNAHD
1AA2 60                     RTS
1AA3 AD 09 8F  MNAHD        LDA   MONEY
1AA6 38                     SEC
1AA7 E5 FF                  SBC   $FF
1AA9 B0 02                  BCS   LESSMONEY
1AAB A9 00                  LDA   #0
1AAD 8D 09 8F  LESSMONEY    STA   MONEY
1AB0 60                     RTS
;
; cknostates -- general routine to check # of states
; y will contain # of states won
1AB1 A0 00     CKNOSTATE    LDY   #0
1AB3 A2 0F                  LDX   #$0F
1AB5 AD 15 8F               LDA   PARTY
1AB8 F0 02                  BEQ   CNCK
1ABA A2 F0                  LDX   #$F0
; mask (for checking state leaning)
1ABC 86 FB     CNCK         STX   $FB
1ABE A2 34                  LDX   #52
1AC0 CA        CKNOLOOP     DEX
1AC1 D0 03                  BNE   CNNOLOOP
1AC3 84 02                  STY   $02
1AC5 60                     RTS
1AC6 BD 00 90  CNNOLOOP     LDA   REALLEAN,X
1AC9 25 FB                  AND   $FB
1ACB F0 F3                  BEQ   CKNOLOOP
1ACD C8                     INY
1ACE D0 F0                  BNE   CKNOLOOP
;
; rest (no campaigning)
; get # of states
1AD0 A5 02     RESTDAY      LDA   $02
1AD2 18                     CLC
1AD3 6D 17 8F               ADC   STREN
; double basic strength + # states
1AD6 0A                     ASL
1AD7 6D 16 8F               ADC   CAMPEFF
; & add campaign eff, 2*states, health
1ADA 6D 08 8F               ADC   HEALTH
1ADD 90 03                  BCC   NEWHEALTH
1ADF 18                     CLC
1AE0 A9 FF                  LDA   #$FF
1AE2 8D 08 8F  NEWHEALTH    STA   HEALTH
1AE5 AD 0A 8F               LDA   LASTSTATE
1AE8 D0 0A                  BNE   ENDDHL
1AEA A9 10                  LDA   #16
1AEC 6D 08 8F               ADC   HEALTH
1AEF B0 03                  BCS   ENDDHL
1AF1 8D 08 8F               STA   HEALTH
1AF4 A9 00     ENDDHL       LDA   #0
1AF6 8D 0A 8F               STA   LASTSTATE
1AF9 60                     RTS
;
; ** ckdemrep
; sets stateleanings
```

```
1AFA A9 92     CKDEMREP     LDA   #>STPVOTES
1AFC 85 FC                  STA   $FC
1AFE A9 00                  LDA   #<STPVOTES
1B00 85 FB                  STA   $FB
1B02 A9 00                  LDA   #0
1B04 AA                     TAX
1B05 F0 07                  BEQ   STARTCKDR
1B07 A0 05     CNCKDR       LDY   #5
1B09 E6 FB     CNCLOOP      INC   $FB
1B0B 88                     DEY
1B0C D0 FB                  BNE   CNCLOOP
1B0E E8        STARTCKDR    INX
1B0F E0 34                  CPX   #52
1B11 D0 01                  BNE   CNSTCK
1B13 60                     RTS
1B14 A0 01     CNSTCK       LDY   #1
1B16 B1 FB                  LDA   ($FB),Y
1B18 C8                     INY
1B19 38                     SEC
1B1A F1 FB                  SBC   ($FB),Y
1B1C D0 06                  BNE   CKDRRESULT
; sort of random (flip flop) leaning
1B1E 20 58 1B               JSR   FLIPFLOPVOTES
1B21 4C 07 1B               JMP   CNCKDR
1B24 B0 0A     CKDRRESUL    BCS   DEMSWIN
1B26 EA        REPSWIN      NOP
1B27 49 FF                  EOR   #$FF
1B29 18                     CLC
1B2A 69 01                  ADC   #1
1B2C A0 80                  LDY   #$80
1B2E D0 02                  BNE   CKFERENCE
1B30 A0 08     DEMSWIN      LDY   #$08
1B32 85 FD     CKFERENCE    STA   $FD
1B34 84 FE                  STY   $FE
; is margin 32 or more?
1B36 29 E0                  AND   #$E0
1B38 F0 02                  BEQ   CNFF16
1B3A D0 14                  BNE   DOLEANING
1B3C 46 FE     CNFF16       LSR   $FE
1B3E A5 FD                  LDA   $FD
; is it 16-31?
1B40 29 10                  AND   #$10
1B42 F0 02                  BEQ   CNFF8
1B44 D0 0A                  BNE   DOLEANING
1B46 46 FE     CNFF8        LSR   $FE
1B48 A5 FD                  LDA   $FD
; is it maybe 8-15?
1B4A 29 08                  AND   #$08
1B4C D0 02                  BNE   DOLEANING
1B4E 46 FE                  LSR   $FE
1B50 A5 FE     DOLEANING    LDA   $FE
1B52 9D 00 90               STA   REALLEAN,X
1B55 4C 07 1B               JMP   CNCKDR
1B58 AD 00 90  FLIPFLOPV    LDA   REALLEAN
1B5B 29 F0                  AND   #$F0
1B5D F0 04                  BEQ   MAKEREP
1B5F A9 01     MAKEDEM      LDA   #$01
1B61 D0 02                  BNE   STRAND
1B63 A9 10     MAKEREP      LDA   #$10
1B65 8D 00 90  STRAND       STA   REALLEAN
1B68 9D 00 90               STA   REALLEAN,X
1B6B 60                     RTS
;
; transleancolr -- translates state leanings to color table
1B6C A2 34     TRANSLEAN    LDX   #52
1B6E CA        TRANSILOO    DEX
1B6F F0 12                  BEQ   ENDOTRANS
1B71 BD 00 90               LDA   REALLEAN,X
; --- dems are bits 0-3; if ne, then dems ate
1B74 29 0F                  AND   #$0F
```

```
1B76 F0 04                  BEQ    REPLEAN
1B78 A9 43     DEMLEAN      LDA    #67
1B7A D0 02                  BNE    STRLEAN
1B7C A9 82     REPLEAN      LDA    #130
1B7E 9D C0 37  STRLEAN      STA    LEANTABLE,X
1B81 D0 EB                  BNE    TRANSILOOP
1B83 60        ENDOTRANS    RTS
;
; tglparty -- switches banks in and out
1B84 AD 15 8F  TGLPARTY     LDA    PARTY
1B87 48                     PHA
1B88 A2 3F                  LDX    #63
1B8A BD 40 8F  TGLOOP       LDA    OTHERCAND,X
1B8D 9D 80 8F               STA    CANSTACK,X
1B90 BD 00 8F               LDA    DAY,X
1B93 9D 40 8F               STA    OTHERCAND,X
1B96 BD 80 8F               LDA    CANSTACK,X
1B99 9D 00 8F               STA    DAY,X
1B9C CA                     DEX
1B9D D0 EB                  BNE    TGLOOP
1B9F 68                     PLA
1BA0 49 01                  EOR    #1
1BA2 8D 15 8F               STA    PARTY
1BA5 60                     RTS
; do television ads
1BA6 AD 09 8F  TELEVISI     LDA    MONEY
1BA9 C9 28                  CMP    #40
1BAB 90 43                  BCC    ENDTVS
1BAD AE 20 8F               LDX    CURREG
1BB0 BD 7F 24               LDA    REGFIRST,X
1BB3 85 FA                  STA    $FA
1BB5 CA                     DEX
1BB6 BD 7F 24               LDA    REGFIRST,X
1BB9 85 F9                  STA    $F9
1BBB C6 F9                  DEC    $F9
1BBD AD 24 25  TLVLOOP      LDA    WEEKNO
1BC0 0A                     ASL
1BC1 18                     CLC
1BC2 6D 19 8F               ADC    TVADS
1BC5 85 FF                  STA    $FF
1BC7 E6 F9                  INC    $F9
1BC9 A5 F9                  LDA    $F9
1BCB C5 FA                  CMP    $FA
1BCD F0 0E                  BEQ    SUBTRTMON
1BCF 20 17 1C               JSR    GOOYRUM
1BD2 46 FF                  LSR    $FF
1BD4 20 5D 1A               JSR    VOTECOUNT
1BD7 20 8C 1A               JSR    SUBHNM
1BDA 4C BD 1B               JMP    TLVLOOP
1BDD AD 09 8F  SUBTRTMON    LDA    MONEY
1BE0 38                     SEC
1BE1 ED 19 8F               SBC    TVADS
1BE4 90 05                  BCC    ESCPIT
1BE6 ED 19 8F               SBC    TVADS
1BE9 B0 02                  BCS    ITCOSTS
1BEB A9 01     ESCPIT       LDA    #1
1BED 8D 09 8F  ITCOSTS      STA    MONEY
1BF0 4C 73 19  ENDTVS       JMP    NEXDAY
;
; camponce/first put base into canstack+1
1BF3 A2 00     CAMPONCE     LDX    #0
1BF5 A0 34                  LDY    #52
1BF7 86 F9     CAMPREGK     STX    $F9
1BF9 84 FA                  STY    $FA
; x = first (-1), y is last (+1)
1BFB E6 F9     CAMLINK      INC    $F9
1BFD A5 F9                  LDA    $F9
1BFF C5 FA                  CMP    $FA
1C01 F0 13                  BEQ    RUMENDL
1C03 AE 81 8F               LDX    CANSTACK+1
1C06 86 FF                  STX    $FF
1C08 20 17 1C               JSR    GOOYRUM
1C0B 20 2C 1A               JSR    ISSCHECK
1C0E 46 FF                  LSR    $FF
1C10 20 5D 1A               JSR    VOTECOUNT
1C13 4C FB 1B               JMP    CAMLINK
1C16 60        RUMENDL      RTS
1C17 20 CF 19  GOOYRUM      JSR    SETZPP
1C1A 20 26 1A               JSR    DCDCFF
1C1D 60                     RTS
;
; bkcm.select
;
1C1E           DEFSP0       =      $3F00
1C1E           DEFSP1       =      $3F40
1C1E           SPRCOL       =      $D027
1C1E           SPRHOR       =      $D01D
1C1E           SPRVER       =      $D017
1C1E           SPRLOC       =      $D000
1C1E           SPRMSB       =      $D010
1C1E           SPRTOG       =      $D015
1C1E           SPRLOCX      =      $D000
1C1E           SPRLOCY      =      $D001
; setsel -- sets up sprites
1C1E A2 00     SETSEL       LDX    #0
1C20 A9 00                  LDA    #0
1C22 9D 00 3F  ZEROUT       STA    DEFSP0,X
1C25 CA                     DEX
1C26 D0 FA                  BNE    ZEROUT
1C28 A9 00     FILSP0       LDA    #0
1C2A AA                     TAX
1C2B A8                     TAY
1C2C B9 0A 1F  CNFIL        LDA    SPDAT0,Y
1C2F 9D 00 3F               STA    DEFSP0,X
1C32 B9 12 1F               LDA    SPDAT1,Y
1C35 9D 40 3F               STA    DEFSP1,X
1C38 E8                     INX
1C39 E8                     INX
1C3A E8                     INX
1C3B C8                     INY
1C3C C0 07                  CPY    #7
1C3E D0 EC                  BNE    CNFIL
1C40 B9 0A 1F               LDA    SPDAT0,Y
1C43 9D 00 3F               STA    DEFSP0,X
1C46 9D 01 3F               STA    DEFSP0+1,X
1C49 9D 02 3F               STA    DEFSP0+2,X
1C4C B9 12 1F               LDA    SPDAT1,Y
1C4F 9D 40 3F               STA    DEFSP1,X
; sprites 0 & 1 are set up
1C52 A9 FC                  LDA    #252
1C54 8D F8 07               STA    2040
1C57 A9 FD                  LDA    #253
; set pointers
1C59 8D F9 07               STA    2041
1C5C A2 07                  LDX    #7
; set all colors to medium gray
1C5E A9 0C                  LDA    #12
1C60 9D 27 D0  COLSPRT      STA    SPRCOL,X
1C63 CA                     DEX
1C64 10 FA                  BPL    COLSPRT
1C66 A9 01                  LDA    #1
1C68 8D 1D D0               STA    SPRHOR
1C6B A9 01                  LDA    #1
1C6D 8D 10 D0               STA    SPRMSB
1C70 A9 04                  LDA    #$04
1C72 8D 00 D0               STA    SPRLOCX
1C75 A9 32                  LDA    #$32
1C77 8D 01 D0               STA    SPRLOCY
1C7A A9 36                  LDA    #54
1C7C 8D 02 D0               STA    SPRLOCX+2
```

```
1C7F A9 38                LDA    #56
1C81 8D 03 D0             STA    SPRLOCY+2
1C84 A9 00                LDA    #0
1C86 A0 04                LDY    #4
1C88 99 02 1F  ZRSPR1     STA    SPX1,Y
1C8B 88                   DEY
1C8C 10 FA                BPL    ZRSPR1
1C8E A9 22                LDA    #>MAPAMAP
1C90 8D 07 1F             STA    MAPA1+1
1C93 A9 AD                LDA    #<MAPAMAP
1C95 8D 06 1F             STA    MAPA1
1C98 A9 36                LDA    #>MAPTABLE
1C9A 8D 09 1F             STA    MAPT1+1
1C9D A9 00                LDA    #<MAPTABLE
1C9F 8D 08 1F             STA    MAPT1
1CA2 60                   RTS
;
1CA3 A9 00     SELECT     LDA    #0
1CA5 85 FD                STA    $FD
1CA7 A9 04                LDA    #4
1CA9 8D 00 D0             STA    SPRLOCX
1CAC 98                   TYA
1CAD 20 FA 1E             JSR    LOC50
1CB0 85 FC                STA    $FC
1CB2 8A                   TXA
1CB3 20 FA 1E             JSR    LOC50
1CB6 85 FB                STA    $FB
; sets max & min
1CB8 8D 01 D0             STA    SPRLOCY
1CBB A9 0C                LDA    #12
1CBD 8D 27 D0             STA    SPRCOL
1CC0 AD 10 D0             LDA    SPRMSB
1CC3 09 01                ORA    #1
1CC5 8D 10 D0             STA    SPRMSB
1CC8 AD 15 D0             LDA    SPRTOG
1CCB 09 01                ORA    #1
1CCD 8D 15 D0             STA    SPRTOG
1CD0 20 1A 1F  SELLOOP    JSR    IJKL
1CD3 AD 6B 1F             LDA    JB
1CD6 F0 F8                BEQ    SELLOOP
1CD8 29 13                AND    #19
1CDA F0 F4                BEQ    SELLOOP
1CDC AA                   TAX
1CDD 29 10                AND    #16
1CDF D0 27                BNE    SELBUTTON
1CE1 8A                   TXA
1CE2 29 01                AND    #1
1CE4 F0 11                BEQ    SELDN
1CE6 AD 01 D0  SELUP      LDA    SPRLOCY
1CE9 C5 FB                CMP    $FB
1CEB F0 E3                BEQ    SELLOOP
1CED C6 FD                DEC    $FD
1CEF 38                   SEC
1CF0 E9 08                SBC    #$08
1CF2 8D 01 D0             STA    SPRLOCY
1CF5 D0 D9                BNE    SELLOOP
1CF7 AD 01 D0  SELDN      LDA    SPRLOCY
1CFA C5 FC                CMP    $FC
1CFC F0 D2                BEQ    SELLOOP
1CFE E6 FD                INC    $FD
1D00 18                   CLC
1D01 69 08                ADC    #$08
1D03 8D 01 D0             STA    SPRLOCY
1D06 D0 C8                BNE    SELLOOP
1D08 A9 00     SELBUTTON  LDA    #0
; change to black color
1D0A 8D 27 D0             STA    SPRCOL
1D0D 20 1A 1F  SELSURE    JSR    IJKL
1D10 AD 6B 1F             LDA    JB
1D13 F0 F8                BEQ    SELSURE
```

```
1D15 29 10                AND    #16
1D17 D0 07                BNE    SELDONE
1D19 A9 0C                LDA    #12
1D1B 8D 27 D0             STA    SPRCOL
1D1E D0 B0                BNE    SELLOOP
1D20 AD 15 D0  SELDONE    LDA    SPRTOG
1D23 29 FE                AND    #254
1D25 8D 15 D0             STA    SPRTOG
1D28 A5 FD                LDA    $FD
; .a now holds select
1D2A 60                   RTS
;
1D2B A2 07     MAPMAGIC   LDX    #7
1D2D BD 02 1F  MAGPIE     LDA    SPX1,X
1D30 95 F7                STA    $F7,X
1D32 CA                   DEX
1D33 10 F8                BPL    MAGPIE
1D35 A9 01                LDA    #1
1D37 8D 28 D0             STA    SPRCOL+1
1D3A AD 15 D0             LDA    SPRTOG
1D3D 09 02                ORA    #2
1D3F 8D 15 D0             STA    SPRTOG
; sprite is turned on
1D42 20 1A 1F  GTKEYB     JSR    IJKL
1D45 AD 6B 1F             LDA    JB
1D48 F0 F8                BEQ    GTKEYB
1D4A 6A                   ROR
1D4B B0 14                BCS    MAGUP
1D4D 6A                   ROR
1D4E B0 43                BCS    MAGDN
1D50 6A                   ROR
1D51 B0 6E                BCS    MAGLF
1D53 6A                   ROR
1D54 B0 05                BCS    JMRTPL
1D56 6A                   ROR
1D57 B0 05                BCS    JMDNPL
1D59 90 E7                BCC    GTKEYB
1D5B 4C F0 1D  JMRTPL     JMP    MAGRT
1D5E 4C E7 1E  JMDNPL     JMP    MAGDONE
1D61 A5 F8     MAGUP      LDA    $F8
1D63 F0 DD                BEQ    GTKEYB
1D65 AD 03 D0             LDA    SPRLOCY+2
1D68 38                   SEC
1D69 E9 04                SBC    #4
1D6B 8D 03 D0             STA    SPRLOCY+2
1D6E C6 F8                DEC    $F8
1D70 A5 F8                LDA    $F8
1D72 6A                   ROR
1D73 B0 03                BCS    CONIT
1D75 4C 24 1E             JMP    STATENE
1D78 A5 FD     CONIT      LDA    $FD
1D7A E9 1A                SBC    #26
1D7C 85 FD                STA    $FD
1D7E B0 02                BCS    DNZMAP
1D80 C6 FE                DEC    $FE
1D82 A5 FB     DNZMAP     LDA    $FB
1D84 38                   SEC
1D85 E9 19                SBC    #25
1D87 85 FB                STA    $FB
1D89 90 03                BCC    DCF
1D8B 4C 24 1E             JMP    STATENE
1D8E C6 FC     DCF        DEC    $FC
1D90 4C 24 1E             JMP    STATENE
1D93 A5 F8     MAGDN      LDA    $F8
1D95 C9 1D                CMP    #29
1D97 F0 A9                BEQ    GTKEYB
1D99 AD 03 D0             LDA    SPRLOCY+2
1D9C 18                   CLC
1D9D 69 04                ADC    #4
1D9F 8D 03 D0             STA    SPRLOCY+2
```

```
1DA2 E6 F8                 INC   $F8
1DA4 A5 F8                 LDA   $F8
1DA6 6A                    ROR
1DA7 B0 7B                 BCS   STATENE
1DA9 A5 FD                 LDA   $FD
1DAB 69 1A                 ADC   #26
1DAD 85 FD                 STA   $FD
1DAF 90 02                 BCC   UPZMAP
1DB1 E6 FE                 INC   $FE
1DB3 A5 FB      UPZMAP     LDA   $FB
1DB5 18                    CLC
1DB6 69 19                 ADC   #25
1DB8 85 FB                 STA   $FB
1DBA 90 68                 BCC   STATENE
1DBC E6 FC                 INC   $FC
1DBE 4C 24 1E              JMP   STATENE
1DC1 A5 F7      MAGLF      LDA   $F7
1DC3 D0 03                 BNE   GHIKK
1DC5 4C 42 1D              JMP   GTKEYB
1DC8 AD 02 D0   GHIKK      LDA   SPRLOCX+2
1DCB 38                    SEC
1DCC E9 04                 SBC   #4
1DCE 8D 02 D0              STA   SPRLOCX+2
1DD1 C6 F7                 DEC   $F7
1DD3 A5 F7                 LDA   $F7
1DD5 6A                    ROR
1DD6 90 4C                 BCC   STATENE
1DD8 A5 FD                 LDA   $FD
1DDA E9 01                 SBC   #1
1DDC 85 FD                 STA   $FD
1DDE B0 02                 BCS   DCZMAP
1DE0 C6 FE                 DEC   $FE
1DE2 A5 FB      DCZMAP     LDA   $FB
1DE4 38                    SEC
1DE5 E9 01                 SBC   #1
1DE7 85 FB                 STA   $FB
1DE9 B0 39                 BCS   STATENE
1DEB C6 FC                 DEC   $FC
1DED 4C 24 1E              JMP   STATENE
1DF0 A5 F7      MAGRT      LDA   $F7
1DF2 C9 31                 CMP   #49
1DF4 D0 03                 BNE   GHW
1DF6 4C 42 1D              JMP   GTKEYB
1DF9 AD 02 D0   GHW        LDA   SPRLOCX+2
1DFC 18                    CLC
1DFD 69 04                 ADC   #4
1DFF 8D 02 D0              STA   SPRLOCX+2
1E02 E6 F7                 INC   $F7
1E04 A5 F7                 LDA   $F7
1E06 6A                    ROR
1E07 90 03                 BCC   CONRQ
1E09 4C 24 1E              JMP   STATENE
1E0C A5 FD      CONRQ      LDA   $FD
1E0E 69 01                 ADC   #1
1E10 85 FD                 STA   $FD
1E12 90 02                 BCC   INZMAP
1E14 E6 FE                 INC   $FE
1E16 A5 FB      INZMAP     LDA   $FB
1E18 18                    CLC
1E19 69 01                 ADC   #1
1E1B 85 FB                 STA   $FB
1E1D 90 05                 BCC   STATENE
1E1F E6 FC                 INC   $FC
1E21 4C 24 1E              JMP   STATENE
;
; now check for character shape, sec & ter, and state #
1E24 A9 01      STATENE    LDA   #1
1E26 85 F9                 STA   $F9
1E28 A5 F8                 LDA   $F8
1E2A 4A                    LSR
```

```
1E2B 90 04                 BCC   CKXVOSP
1E2D 06 F9                 ASL   $F9
1E2F 06 F9                 ASL   $F9
1E31 A5 F7      CKXVOSP    LDA   $F7
1E33 6A                    ROR
1E34 B0 02                 BCS   CONSF
1E36 06 F9                 ASL   $F9
; $f9 contains matcher
1E38 A0 00      CONSF      LDY   #0
1E3A B1 FB                 LDA   ($FB),Y
1E3C 85 02                 STA   $02
1E3E A5 F9                 LDA   $F9
1E40 31 FD                 AND   ($FD),Y
1E42 D0 26                 BNE   NOMESSTS
1E44 A9 C0                 LDA   #$C0
1E46 24 02                 BIT   $02
1E48 30 0D                 BMI   MAYBOCEAN
1E4A A5 02                 LDA   $02
1E4C 29 3F                 AND   #$3F
1E4E AA                    TAX
1E4F BD 78 22              LDA   SECTABLE,X
1E52 85 02                 STA   $02
1E54 4C 6A 1E              JMP   NOMESSTS
1E57 50 07      MAYBOCEAN  BVC   TERJUNK
1E59 A9 00                 LDA   #0
1E5B 85 02                 STA   $02
1E5D 4C 6A 1E              JMP   NOMESSTS
1E60 A5 02      TERJUNK    LDA   $02
1E62 29 3F                 AND   #$3F
1E64 AA                    TAX
1E65 BD 79 22              LDA   TERTABLE,X
1E68 85 02                 STA   $02
; found the state ($02) -- print it
1E6A A2 0F      NOMESSTS   LDX   #15
1E6C A0 10                 LDY   #16
1E6E 18                    CLC
; plot routine to set cursor
1E6F 20 F0 FF              JSR   $FFF0
1E72 A9 95                 LDA   #149
1E74 20 D2 FF              JSR   $FFD2
1E77 A9 20                 LDA   #32
1E79 A2 07                 LDX   #7
1E7B 20 D2 FF   BLNKWDW    JSR   $FFD2
1E7E CA                    DEX
1E7F 10 FA                 BPL   BLNKWDW
1E81 A9 9D                 LDA   #157
1E83 A2 07                 LDX   #7
1E85 20 D2 FF   CURKWDW    JSR   $FFD2
1E88 CA                    DEX
1E89 10 FA                 BPL   CURKWDW
1E8B A5 02                 LDA   $02
1E8D D0 03                 BNE   GHWW
1E8F 4C 42 1D              JMP   GTKEYB
1E92 29 3F      GHWW       AND   #$3F
1E94 0A                    ASL
1E95 AA                    TAX
1E96 BD DC 21              LDA   STNAME,X
1E99 20 D2 FF              JSR   $FFD2
1E9C BD DD 21              LDA   STNAME+1,X
1E9F 20 D2 FF              JSR   $FFD2
1EA2 A9 20                 LDA   #32
1EA4 20 D2 FF              JSR   $FFD2
1EA7 BD 00 78              LDA   STEVOTE,X
1EAA AA                    TAX
1EAB C9 0A                 CMP   #10
1EAD B0 05                 BCS   EVPX
1EAF A9 20                 LDA   #32
1EB1 20 D2 FF              JSR   $FFD2
1EB4 A9 00      EVPX       LDA   #0
1EB6 20 CD BD              JSR   $BDCD
```

```
1EB9 A9 1D               LDA   #29
1EBB 20 D2 FF            JSR   $FFD2
1EBE A9 90               LDA   #144
1EC0 20 D2 FF            JSR   $FFD2
1EC3 A9 12               LDA   #18
; cr/blk/reverseon
1EC5 20 D2 FF            JSR   $FFD2
1EC8 A9 A0               LDA   #160
1ECA 20 D2 FF            JSR   $FFD2
1ECD A5 02               LDA   $02
1ECF 29 3F               AND   #$3F
1ED1 A2 00               LDX   #0
1ED3 E8       RGNM       INX
1ED4 DD 7F 24            CMP   REGFIRST,X
1ED7 B0 FA               BCS   RGNM
1ED9 8A                  TXA
1EDA 69 B0               ADC   #176
1EDC 20 D2 FF            JSR   $FFD2
1EDF A9 92               LDA   #146
1EE1 20 D2 FF            JSR   $FFD2
1EE4 4C 42 1D            JMP   GTKEYB
1EE7 AD 15 D0 MAGDONE    LDA   SPRTOG
1EEA 29 FD               AND   #253
1EEC 8D 15 D0            STA   SPRTOG
1EEF A2 07               LDX   #7
1EF1 B5 F7    MAGPIX     LDA   $F7,X
1EF3 9D 02 1F            STA   SPX1,X
1EF6 CA                  DEX
1EF7 10 F8               BPL   MAGPIX
1EF9 60                  RTS
1EFA EA       LOC50      NOP
1EFB 0A                  ASL
1EFC 0A                  ASL
1EFD 0A                  ASL
1EFE 18                  CLC
1EFF 69 32               ADC   #$32
1F01 60                  RTS
1F02 00       SPX1       .BYTE 0
1F03 00       SPY1       .BYTE 0
1F04 00 00    MKR1       .BYTE 0,0
1F06 00 00    MAPA1      .BYTE 0,0
1F08 00 00    MAPT1      .BYTE 0,0
1F0A C0 C0 E0 SPDAT0     .BYTE 192,192,224,240,22
                               4
1F0F C0 C8 FF            .BYTE 192,200,255
1F12 FF 99 81 SPDAT1     .BYTE 255,153,129,195
1F16 C3 81 99            .BYTE 195,129,153,255
;
; bkcm.ijkl
;
1F1A A9 00    IJKL       LDA   #0
1F1C 8D 6B 1F            STA   JB
1F1F AD 00 DC JOYZ1      LDA   $DC00
1F22 29 1F               AND   #$1F
1F24 49 1F               EOR   #$1F
1F26 D0 2D               BNE   DEBOUNCE
1F28 AD 01 DC JOYZ2      LDA   $DC01
1F2B 29 1F               AND   #$1F
1F2D 49 1F               EOR   #$1F
1F2F D0 24               BNE   DEBOUNCE
; check inkey
1F31 20 E4 FF CHKBL      JSR   $FFE4
1F34 D0 01               BNE   CNIJKL
1F36 60                  RTS
1F37 38       CNIJKL     SEC
1F38 E9 49               SBC   #$49
1F3A 90 DE               BCC   IJKL
1F3C AA                  TAX
1F3D E8                  INX
1F3E E9 05               SBC   #5
```

```
1F40 B0 D8               BCS   IJKL
1F42 8A                  TXA
1F43 29 02               AND   #2
1F45 F0 04               BEQ   CLIJKL
1F47 8A                  TXA
1F48 49 01               EOR   #1
1F4A AA                  TAX
1F4B A9 00    CLIJKL     LDA   #0
1F4D 8D 6B 1F            STA   JB
1F50 38                  SEC
1F51 2A       MARKIJ     ROL
1F52 CA                  DEX
1F53 D0 FC               BNE   MARKIJ
1F55 8D 6B 1F DEBOUNCE   STA   JB
1F58 AD 00 DC DEBLOP     LDA   $DC00
1F5B 2D 01 DC            AND   $DC01
1F5E 29 10               AND   #16
1F60 F0 F6               BEQ   DEBLOP
1F62 A9 06               LDA   #6
1F64 65 A2               ADC   $A2
1F66 C5 A2    DLP33      CMP   $A2
1F68 D0 FC               BNE   DLP33
1F6A 60                  RTS
1F6B 00       JB         .BYTE 00
1F6C 20 FA 1A ELECTIOND  JSR   CKDEMREP
1F6F 20 B1 1A            JSR   CKNOSTATES
; number of states won
1F72 A5 02               LDA   $02
1F74 C9 1A               CMP   #26
1F76 90 03               BCC   LASTMINCM
1F78 20 84 1B            JSR   TGLPARTY
; now do the last minute campaigning
; for the loser (so far)
1F7B 20 7A 21 LASTMINCM  JSR   NINTHWEEK
1F7E 20 84 1B            JSR   TGLPARTY
; and the winner
1F81 20 7A 21            JSR   NINTHWEEK
; now split the undecided voters
1F84 A9 01               LDA   #1
; first states first $fb/stp votes $fd/stissues
1F86 20 CF 19            JSR   SETZPP
1F89 A0 05    SPLITLOOP  LDY   #5
1F8B B1 FB               LDA   ($FB),Y
1F8D 4A                  LSR
1F8E 4A                  LSR
; undecided/4
1F8F AA                  TAX
1F90 A0 02               LDY   #2
1F92 8A       SPLIT2     TXA
1F93 18                  CLC
1F94 71 FB               ADC   ($FB),Y
1F96 90 02               BCC   PUTTIN1
1F98 A5 FF               LDA   $FF
1F9A 91 FB    PUTTIN1    STA   ($FB),Y
1F9C 88                  DEY
1F9D D0 F3               BNE   SPLIT2
; now check for ties
1F9F A0 02               LDY   #2
1FA1 D1 FB               CMP   ($FB),Y
1FA3 D0 13               BNE   NEXLASUND
; resolve the tie
1FA5 A0 03               LDY   #3
1FA7 B1 FB               LDA   ($FB),Y
1FA9 C8                  INY
1FAA 38                  SEC
1FAB F1 FB               SBC   ($FB),Y
1FAD A9 80               LDA   #$80
1FAF 2A                  ROL
1FB0 A8                  TAY
1FB1 C8                  INY
```

```
1FB2 B1 FB               LDA  ($FB),Y
1FB4 E9 01               SBC  #1
1FB6 91 FB               STA  ($FB),Y
;
1FB8 A5 FB    NEXLASUND  LDA  $FB
1FBA 18                  CLC
1FBB 69 05               ADC  #5
1FBD 85 FB               STA  $FB
1FBF C9 FF               CMP  #255
1FC1 D0 C6               BNE  SPLITLOOP
; and the undecideds are split up
;
; now show the results
1FC3 20 FA 1A            JSR  CKDEMREP
1FC6 20 B1 1A            JSR  CKNOSTATES
1FC9 A2 33               LDX  #51
1FCB BD 00 90 SHIFTYLOO  LDA  REALLEAN,X
1FCE 29 11               AND  #$11
1FD0 F0 03               BEQ  TRYFEWMORE
1FD2 1E 00 90            ASL  REALLEAN,X
1FD5 CA       TRYFEWMOR  DEX
1FD6 D0 F3               BNE  SHIFTYLOOP
; eliminates undecideds
1FD8 20 6C 1B            JSR  TRANSLEANCOLR
1FDB 20 ED 09            JSR  MAKEMAP
1FDE 20 57 16            JSR  BARCHART
1FE1 20 5A 0B            JSR  CMLOGOPUT
1FE4 20 CD 15            JSR  MENUITIN
1FE7 A2 0F               LDX  #15
; (for border)
1FE9 86 02               STX  $02
1FEB A0 1D    DEOLH      LDY  #29
1FED 18                  CLC
; kernal plot
1FEE 20 F0 FF            JSR  $FFF0
1FF1 A9 98               LDA  #152
; gray 2
1FF3 20 D2 FF            JSR  $FFD2
1FF6 A9 20               LDA  #32
; blank space
1FF8 A2 0B               LDX  #11
1FFA 20 D2 FF BLANKITS   JSR  $FFD2
1FFD CA                  DEX
1FFE D0 FA               BNE  BLANKITS
2000 E6 02               INC  $02
2002 A6 02               LDX  $02
2004 E0 18               CPX  #24
2006 D0 E3               BNE  DEOLH
2008 AD 64 07            LDA  1892
200B 8D 8C 07            STA  1932
200E 8D B4 07            STA  1972
; even up edges
2011 8D DC 07            STA  2012
2014 A9 20               LDA  #32
2016 A2 0B               LDX  #11
2018 9D DC 07 BLNKBOT    STA  2012,X
201B CA                  DEX
; blanks at bottom/for totals
201C D0 FA               BNE  BLNKBOT
201E A9 14               LDA  #20
; t for total
2020 8D E2 07            STA  2018
2023 A9 00               LDA  #0
2025 A2 03               LDX  #3
2027 95 03    ZEREVOT    STA  $03,X
2029 CA                  DEX
202A 10 FB               BPL  ZEREVOT
202C A9 09               LDA  #9
202E 85 AE               STA  $AE
2030 A9 00    PREPGL     LDA  #0
```

```
; prepare for grand loop
2032 8D 20 8F            STA  CURREG
;
; ** region totals
2035 EE 20 8F REGIONTOT  INC  CURREG
2038 AD 20 8F            LDA  CURREG
203B C9 0A               CMP  #10
203D D0 03               BNE  GPRIX
203F 4C 30 20            JMP  PREPGL
2042 20 99 21 GPRIX      JSR  CONSULHD
2045 A9 00               LDA  #0
2047 85 B2               STA  $B2
2049 85 B3               STA  $B3
204B A2 04               LDX  #4
204D 86 FB               STX  Z3
204F A0 1F               LDY  #31
2051 84 FC               STY  Z3+1
2053 A9 BE               LDA  #190
2055 85 F7               STA  Z1
2057 85 F9               STA  Z2
2059 A9 04               LDA  #4
205B 85 F8               STA  Z1+1
; z page pointers
205D 85 FA               STA  Z2+1
; # of states in region
205F A6 A7               LDX  $A7
2061 A0 03    BLUNKOUT   LDY  #3
2063 A9 20               LDA  #32
2065 91 F7    BLUNKIN    STA  (Z1),Y
2067 88                  DEY
2068 10 FB               BPL  BLUNKIN
206A A5 F7               LDA  Z1
206C 18                  CLC
206D 69 28               ADC  #40
206F 85 F7               STA  Z1
2071 90 02               BCC  DOWNTZZ
2073 E6 F8               INC  Z1+1
2075 CA       DOWNTZZ    DEX
2076 D0 E9               BNE  BLUNKOUT
2078 AE 20 8F            LDX  CURREG
207B BD 7F 24            LDA  REGFIRST,X
207E 85 FE               STA  Z4+1
2080 CA                  DEX
2081 BD 7F 24            LDA  REGFIRST,X
2084 85 FD               STA  Z4
2086 A6 FB    EVPLOT     LDX  Z3
2088 A4 FC               LDY  Z3+1
208A 18                  CLC
; plot for e-votes
208B 20 F0 FF            JSR  $FFF0
208E A6 FD    LARGEXLOO  LDX  Z4
;
2090 BD 00 90            LDA  REALLEAN,X
2093 29 0F               AND  #$0F
2095 D0 03               BNE  ABC19
2097 4C 6F 21            JMP  DOTHEREPS
209A BD 44 22 ABC19      LDA  CREVOTE,X
209D AA                  TAX
209E 18                  CLC
209F 65 B2               ADC  $B2
20A1 85 B2               STA  $B2
20A3 8A                  TXA
20A4 C9 0A               CMP  #10
20A6 B0 05               BCS  PRNEVOT
20A8 A9 20               LDA  #32
; less than ten, print a space
20AA 20 D2 FF            JSR  $FFD2
20AD A9 9A    PRNEVOT    LDA  #154
20AF 20 D2 FF            JSR  $FFD2
; light blue
```

```
20B2 A9 00                 LDA   #0
; print e-votes
20B4 20 CD BD              JSR   $BDCD
20B7 A6 FB                 LDX   Z3
20B9 A0 25                 LDY   #37
20BB 18                    CLC
; plot for erase
20BC 20 F0 FF              JSR   $FFF0
20BF A0 03                 LDY   #3
20C1 A9 20                 LDA   #32
20C3 20 D2 FF   OBLITERT   JSR   $FFD2
20C6 88                    DEY
20C7 D0 FA                 BNE   OBLITERT
20C9 E6 FB      FINIRL3    INC   Z3
20CB E6 FD                 INC   Z4
20CD C6 A7                 DEC   $A7
20CF D0 B5                 BNE   EVPLOT
20D1 A5 AE                 LDA   $AE
20D3 D0 03                 BNE   ABC20
20D5 4C 69 21              JMP   PRESREG9
20D8 AD 20 8F   ABC20      LDA   CURREG
20DB 18                    CLC
20DC 69 0E                 ADC   #14
20DE AA                    TAX
20DF A0 1F                 LDY   #31
20E1 18                    CLC
; plot
20E2 20 F0 FF              JSR   $FFF0
20E5 A9 9A                 LDA   #154
20E7 20 D2 FF              JSR   $FFD2
20EA A5 B2                 LDA   $B2
20EC AA                    TAX
20ED C9 0A                 CMP   #10
20EF B0 05                 BCS   DEMRVOT
20F1 A9 20                 LDA   #32
20F3 20 D2 FF              JSR   $FFD2
20F6 A9 00      DEMRVOT    LDA   #0
; dem votes
20F8 20 CD BD              JSR   $BDCD
20FB A9 9C                 LDA   #156
20FD 20 D2 FF              JSR   $FFD2
2100 A9 20                 LDA   #32
2102 48                    PHA
2103 20 D2 FF              JSR   $FFD2
2106 AD 20 8F              LDA   CURREG
2109 09 30                 ORA   #$30
210B 20 D2 FF              JSR   $FFD2
210E 68                    PLA
210F 20 D2 FF              JSR   $FFD2
2112 20 D2 FF              JSR   $FFD2
2115 A5 B3                 LDA   $B3
2117 AA                    TAX
2118 C9 0A                 CMP   #10
211A B0 05                 BCS   REP92
211C A9 20                 LDA   #32
211E 20 D2 FF              JSR   $FFD2
2121 A9 1C      REP92      LDA   #28
2123 20 D2 FF              JSR   $FFD2
2126 A9 00                 LDA   #0
; rep votes
2128 20 CD BD              JSR   $BDCD
212B A2 18                 LDX   #24
212D A0 1E                 LDY   #30
212F 18                    CLC
2130 20 F0 FF              JSR   $FFF0
2133 A9 98                 LDA   #152
; gray 2
2135 20 D2 FF              JSR   $FFD2
2138 A5 B2                 LDA   $B2
213A 18                    CLC
```

```
213B 65 03                 ADC   $03
213D 85 03                 STA   $03
213F A9 00                 LDA   #0
2141 65 04                 ADC   $04
2143 85 04                 STA   $04
2145 A5 B3                 LDA   $B3
2147 65 05                 ADC   $05
2149 85 05                 STA   $05
214B A9 00                 LDA   #0
214D 65 06                 ADC   $06
214F 85 06                 STA   $06
2151 A6 03                 LDX   $03
2153 A5 04                 LDA   $04
2155 20 CD BD              JSR   $BDCD
2158 A2 18                 LDX   #24
215A A0 24                 LDY   #36
215C 18                    CLC
215D 20 F0 FF              JSR   $FFF0
2160 A6 05                 LDX   $05
2162 A5 06                 LDA   $06
2164 20 CD BD              JSR   $BDCD
2167 C6 AE                 DEC   $AE
2169 20 2E 11   PRESREG9   JSR   PRESSFIRE
216C 4C 35 20              JMP   REGIONTOTS
; the republicans won the state
216F BD 44 22   DOTHEREPS  LDA   CREVOTE,X
2172 18                    CLC
2173 65 B3                 ADC   $B3
2175 85 B3                 STA   $B3
2177 4C C9 20              JMP   FINIRL3
; ** ninth week routine
; add lastwk, twice issues, & campeff (3/2) to every state
217A AD 1A 8F   NINTHWEEK  LDA   LASTWK
217D 8D 81 8F              STA   CANSTACK+1
; campaign
2180 20 F3 1B              JSR   CAMPONCE
2183 A9 00                 LDA   #0
2185 8D 81 8F              STA   CANSTACK+1
2188 AE 20 8F              LDX   CURREG
218B BD 7F 24              LDA   REGFIRST,X
218E A8                    TAY
218F CA                    DEX
2190 BD 7F 24              LDA   REGFIRST,X
2193 AA                    TAX
; last region
2194 CA                    DEX
2195 20 F7 1B              JSR   CAMPREGK
2198 60                    RTS
;
2199 A9 9C      CONSULHD   LDA   #156
219B 20 D2 FF              JSR   $FFD2
; print purple region header
219E 20 EB 11              JSR   OREET
21A1 A2 03                 LDX   #3
21A3 A0 1E                 LDY   #30
21A5 18                    CLC
21A6 20 F0 FF              JSR   $FFF0
; for dem/rep top
21A9 20 C1 21              JSR   HDLLRD
21AC AE 20 8F              LDX   CURREG
21AF BD 7F 24              LDA   REGFIRST,X
21B2 CA                    DEX
21B3 38                    SEC
21B4 FD 7F 24              SBC   REGFIRST,X
21B7 85 A7                 STA   $A7
21B9 69 03                 ADC   #3
21BB AA                    TAX
21BC A0 1E                 LDY   #30
21BE 20 F0 FF              JSR   $FFF0
21C1 A2 00      HDLLRD     LDX   #0
```

```
21C3 BD CF 21  HDLLRE     LDA    DRHD,X
21C6 D0 01                BNE    BOTTOMQ
21C8 60                   RTS
21C9 20 D2 FF  BOTTOMQ    JSR    $FFD2
21CC E8                   INX
21CD D0 F4                BNE    HDLLRE
21CF 9A 44 45  DRHD       .ASC   " [<7>]DEM {RED}
                                 REP"
21DB 00                   .BYTE  0
;
; bkcm.tables
;
21DC 20 20 4D  STNAME     .ASC   " MENHVTMARICT"
21EA 4E 59 4E             .ASC   "NYNJPA"
21F0 4F 48 49             .ASC   "OHINILMIWI"
21FA 4D 4E 49             .ASC   "MNIAMONDSDNEK
                                 S"
2208 44 45 4D             .ASC   "DEMDDCVAWVNCS
                                 CGAFL"
221A 4B 59 54             .ASC   "KYTNALMS"
2222 41 52 4C             .ASC   "ARLAOKTX"
222A 4D 54 49             .ASC   "MTIDWYCONMAZU
                                 TNV"
223A 57 41 4F             .ASC   "WAORCAAKHI"
; below is crunched electoral votes
2244           STEVOTE    =      $7800
2244 00 04 04  CREVOTE    .BYTE  0,4,4,3,13,4,8
224B 24 10 19             .BYTE  36,16,25
224E 17 0C 18             .BYTE  23,12,24,20,11
2253 0A 08 0B             .BYTE  10,8,11,3,3,5,7
225A 03 0A 03             .BYTE  3,10,3,12,6,13,8,12,
                                 21
2263 09 0B 09             .BYTE  9,11,9,7
2267 06 0A 08             .BYTE  6,10,8,29
226B 04 04 03             .BYTE  4,4,3,8,5,7,5,4
2273 0A 07 2F             .BYTE  10,7,47,3,4
2278 00        SECTABLE   .BYTE  0
2279 01 03 03  TERTABLE   .BYTE  1,3,3,4,5,8,9,8
2281 0A 1A 1F             .BYTE  10,26,31,11,14,16,1
                                 4,12
2289 25 0C 13             .BYTE  37,12,19,16,17,20,2
                                 5,9
2291 20 19 19             .BYTE  32,25,25,29,27,30,2
                                 6,31
2299 1D 20 11             .BYTE  29,32,17,34,38,35,4
                                 0,40
22A1 29 2A 26             .BYTE  41,42,38,37,46,49,4
                                 0,40
22A9 2C 00 00             .BYTE  44,0,0,0
22AD EF EF 2F  MAPAMAP    .BYTE  239,239,47,111,231,
                                 231,231,231
22B5 E7 D2 D2             .BYTE  231,210,210,210,20
                                 7,207,207,0
22BD 00 00 00             .BYTE  0,0,0,0,0,0,0,193
22C5 C1 EF 2F             .BYTE  193,239,47,47,111,1
                                 03,39,39
22CD 27 27 12             .BYTE  39,39,18,18,18,15,7
                                 9,79
22D5 4D CD CD             .BYTE  77,205,205,0,0,0,0,0
22DD C1 C1 F0             .BYTE  193,193,240,48,48,1
                                 12,103,39
22E5 27 27 27             .BYTE  39,39,39,19,19,19,1
                                 5,15
22ED 4F 0E CE             .BYTE  79,14,206,205,0,0,0,
                                 199
22F5 07 42 C1             .BYTE  7,66,193,240,48,48,
                                 40,40
22FD 67 29 29             .BYTE  103,41,41,41,19,19,
                                 19,16
2305 10 4E 8C             .BYTE  16,78,140,205,13,20
                                 5,201,199
230D 07 07 04             .BYTE  7,7,4,196,241,49,11
                                 0,46
2315 2E 2D 29             .BYTE  46,45,41,41,41,20,2
                                 0,20
231D 54 10 50             .BYTE  84,16,80,12,76,11,1
                                 0,73
2325 09 09 47             .BYTE  9,9,71,70,197,241,4
                                 9,110
232D 2E 2E 2D             .BYTE  46,46,45,45,42,42,4
                                 2,149
2335 95 95 11             .BYTE  149,149,17,145,12,7
                                 6,11,138
233D 4A 89 58             .BYTE  74,137,88,151,200,0
                                 ,0,49
2345 31 6E 2E             .BYTE  49,110,46,45,45,42,
                                 42,42
234D 15 15 15             .BYTE  21,21,21,85,17,145,
                                 76,75
2355 1F 5F 5A             .BYTE  31,95,90,25,87,214,
                                 0,0
235D F1 31 6E             .BYTE  241,49,110,172,44,4
                                 4,43,43
2365 AB 65 25             .BYTE  171,101,37,37,81,99
                                 ,17,96
236D 60 60 59             .BYTE  96,96,89,91,91,91,0,
                                 0
2375 00 00 F1             .BYTE  0,0,241,49,113,44,4
                                 4,43
237D 2B 2B 26             .BYTE  43,43,38,37,37,165,
                                 35,35
2385 62 A1 A0             .BYTE  98,161,160,93,156,2
                                 7,219,0
238D 00 00 00             .BYTE  0,0,0,241,49,113,44,
                                 44
2395 2B 2B 6B             .BYTE  43,43,107,38,38,38,
                                 102,35
239D A3 22 21             .BYTE  163,34,33,97,29,92,
                                 156,0
23A5 00 00 00             .BYTE  0,0,0,0,0,0,0,236
23AD EC EB E6             .BYTE  236,235,230,38,38,3
                                 8,38,38
23B5 24 64 22             .BYTE  36,100,34,33,33,29,
                                 29,0
23BD 00 00 00             .BYTE  0,0,0,0,242,50,242,0
23C5 F3 00 00             .BYTE  243,0,0,0,230,230,3
                                 8,38
23CD E6 24 E4             .BYTE  230,36,228,226,225,
                                 222,222,30
23D5 00 00 00             .BYTE  0,0,0,0,0,242,50,50
23DD 00 00 F3             .BYTE  0,0,243,0,0,0,0,230
23E5 E6 00 00             .BYTE  230,0,0,0,0,0,0,0
23ED DE DE 00             .BYTE  222,222,0,0,0,0,242,
                                 242
23F5 F2 F2 00             .BYTE  242,242,0,0,243,0,0,
                                 0
23FD 00 E6 00             .BYTE  0,230,0,0,0,0,0,0
2405 00 DE 1E             .BYTE  0,222,30,222,0,0,24
                                 2,0
240D 00 00 F2             .BYTE  0,0,242,0,0,0,0,0
2415 00 00 00             .BYTE  0,0,0,0,0,0,0,0
241D 00 00 00             .BYTE  0,0,0,222,222,0,0,25
                                 5
2425 20 52 45  REGIONAME  .ASC   " REGIONS"
242D 00                   .BYTE  0
242E 4E 45 57             .ASC   "NEW ENGL"
2436 00                   .BYTE  0
2437 55 52 42             .ASC   "URBAN NE"
243F 00                   .BYTE  0
2440 48 45 41             .ASC   "HEARTLND"
```

```
2448 00                     .BYTE  0
2449 47 20 50               .ASC   "G PLAINS"
2451 00                     .BYTE  0
2452 41 54 4C               .ASC   "ATLANTIC"
245A 00                     .BYTE  0
245B 53 4F 55               .ASC   "SOUTHERN"
2463 00                     .BYTE  0
2464 41 52 4B               .ASC   "ARKLATEXY"
246C 00                     .BYTE  0
246D 4D 4F 55               .ASC   "MOUNTAIN"
2475 00                     .BYTE  0
2476 50 41 43               .ASC   "PACIFIC"
247D 20 00                  .BYTE  32,0
; first state #s in
247F 01 07 0A   REGFIRST    .BYTE  1,7,10,15,22,31,35,3
                                   9,47,52
;
; pvcrunch -- crunched leanings
; region 1
2489 34 DC F3   PVCRUNCH    .BYTE  52,220,243,243,47,6
                                   3,220
; region 2
2490 4E E4 4D               .BYTE  78,228,77
; region 3
2493 4D E4 E3               .BYTE  77,228,227,206,92
; region 4
2498 3E F3 5C               .BYTE  62,243,92,227,242,2
                                   27,243
; region 5
249F 63 3F 2F               .BYTE  99,63,47,228,63,69,
                                   100,190,69
; region 6
24A8 46 64 4D               .BYTE  70,100,77,77
; region 7
24AC 46 46 D4               .BYTE  70,70,212,78
; region 8
24B0 D4 F3 F3               .BYTE  212,243,243,197,21
                                   2,228,243,197
; region 9
24B8 EB F2 E4               .BYTE  235,242,228,242,47
24BD 21 3B 21   ISSU1       .BYTE  33,59,33,246,104,12
                                   6
24C3 CA BD 24               .BYTE  202,189,36
24C6 61 59 BD               .BYTE  97,89,189,220,52
24CB 76 7A 51               .BYTE  118,122,81,38,3,171
                                   ,186
24D2 EE FE CC               .BYTE  238,254,204,171,2,8
                                   0,70,70,235
24DB 00 91 45               .BYTE  0,145,69,1
24DF 01 86 57               .BYTE  1,134,87,203
24E3 61 60 77               .BYTE  97,96,119,223,66,23
                                   4,170,246
24EB F5 EA 9E               .BYTE  245,234,158,124,25
                                   4
24F0 6F F7 39   ISSU2       .BYTE  111,247,57,67,159,2
                                   11
24F6 42 1B 5F               .BYTE  66,27,95
24F9 1D 68 A4               .BYTE  29,104,164,179,5
24FE 41 34 E9               .BYTE  65,52,233,44,56,4,1
                                   36
2505 11 D2 42               .BYTE  17,210,66,230,63,16
                                   9,175,77,154
250E 39 3D 5C               .BYTE  57,61,92,140
2512 3E 2F 78               .BYTE  62,47,120,216
2516 25 3B 05               .BYTE  37,59,5,145,213,145
                                   ,243,187
251E F2 0B E6               .BYTE  242,11,230,131,193
;
;
2523 00         FIRSTCAN    .BYTE  0
```

```
2524 00         WEEKNO      .BYTE  0
2525 44 52 55   PRTIS       .ASC   "DRU"
2528 9F 1C 98   PRTCO       .ASC   "{CYN}{RED} [<5
                                   >]"
252B 00 00                  .BYTE  0,0
; sat-sun (backwards)
252D            DAY         =      $8F00
252D            HEALTH      =      $8F08
252D            MONEY       =      $8F09
252D            LASTSTATE   =      $8F0A
252D            LASTREGI    =      $8F0B
252D            HOMESTATE   =      $8F0C
252D            JOB         =      $8F0D
252D            INCUMB      =      $8F0F
252D            CHAR        =      $8F10
252D            STAM        =      $8F11
252D            INTL        =      $8F12
252D            EXPR        =      $8F13
252D            APPL        =      $8F14
; 0=dem 1=rep
252D            PARTY       =      $8F15
252D            CAMPEFF     =      $8F16
252D            STREN       =      $8F17
252D            FUNDR       =      $8F18
252D            TVADS       =      $8F19
252D            LASTWK      =      $8F1A
; the five stands
252D            STAND0      =      $8F1B
252D            CURREG      =      $8F20
252D            HOMEREGI    =      $8F21
252D            OTHERCAN    =      $8F40
252D            OTDAY       =      $8F40
252D            OTHEALTH    =      $8F48
252D            OTMONEY     =      $8F49
252D            OTLASTST    =      $8F4A
252D            OTLASTRE    =      $8F4B
252D            OTHOMEST    =      $8F4C
252D            OTJOB       =      $8F4D
252D            OTINCUMB    =      $8F4F
252D            OTCHAR      =      $8F50
252D            OTSTAM      =      $8F51
252D            OTINTL      =      $8F52
252D            OTEXPR      =      $8F53
252D            OTAPPL      =      $8F54
; 0=dem 1=rep
252D            OTPARTY     =      $8F55
252D            OTCAMPEF    =      $8F56
252D            OTSTREN     =      $8F57
252D            OTFUNDR     =      $8F58
252D            OTTVADS     =      $8F59
252D            OTLASTWK    =      $8F5A
252D            OTSTAND0    =      $8F5B
252D            OTCURREG    =      $8F60
252D            OTHOMEREG   =      $8F61
252D            CANSTACK    =      $8F80
252D            STISSUES    =      $9100
252D            STPVOTES    =      $9200
```

Heat Seeker Source Code

Source code by Tim Victor

```
0801                          *=     $801
;
; --- basic program line
0801 0D 08 01                .BYTE   $D,8,1,0,$9E,$32,$3
                                     0,$36,$31
080A 00 00 00                .BYTE   0,0,0
;
; --- equates
;
080D           SCRPTR         =      $FE
080D           COLPTR         =      $FC
080D           SOURCE         =      $FE
080D           DEST           =      $FC
;sprite 0,x
080D           SPRITEX        =      $D000
;sprite 0,y
080D           SPRITEY        =      $D001
;sprite 0,x
080D           PLANEX         =      SPRITEX
;sprite 0,y
080D           PLANEY         =      SPRITEY
;sprite 1,x
080D           CHASEX         =      SPRITEX+2
;sprite 1,y
080D           CHASEY         =      SPRITEY+2
;sprite bit 9
080D           MSB            =      $D010
;spr-spr collisions
080D           SSCREG         =      $D01E
;spr-bkg collisions
080D           SBCREG         =      $D01F
; sprite enable
080D           SPRITEN        =      $D015
;
080D           SID            =      $D400
080D           V1             =      SID
080D           V2             =      SID + 7
080D           V3             =      SID + $E
080D           JOY2           =      $DC00
080D           JOY1           =      $DC01
;
; video initialize
080D           CINT           =      $FFB5
;
080D           SCORE          =      $7C8
080D           SCORE2         =      $7E1
;
;***************************************************
;*                    code                         *
;***************************************************
;
080D           START          =      *
; --- set screen colors
080D A9 0E                    LDA    #14
080F 8D 21 D0                 STA    $D021
0812 A9 02                    LDA    #2
0814 8D 20 D0                 STA    $D020
;
; --- init sound
;
```

```
0817 A0 18                    LDY    #$18
0819 A9 00                    LDA    #0
081B 99 FF D3  SNDLUP         STA    SID-1,Y
081E 88                       DEY
081F D0 FA                    BNE    SNDLUP
;
0821 A9 02                    LDA    #2
0823 8D 17 D4                 STA    SID+$17
0826 A9 1F                    LDA    #$1F
0828 8D 18 D4                 STA    SID+$18
082B A9 08                    LDA    #8
082D 8D 16 D4                 STA    SID+$16
0830 A9 03                    LDA    #$3
0832 8D 08 D4                 STA    V2+1
0835 A9 3D                    LDA    #$3D
0837 8D 0C D4                 STA    V2+5
;
083A A9 00                    LDA    #$0
083C 8D 0F D4                 STA    V3+1
083F 8D 0E D4                 STA    V3
0842 A9 20                    LDA    #$20
0844 8D 13 D4                 STA    V3+5
0847 A9 7F                    LDA    #$7F
0849 8D 14 D4                 STA    V3+6
084C A9 81                    LDA    #$81
084E 8D 12 D4                 STA    V3+4
;
0851 A9 01                    LDA    #$1
0853 8D 03 D4                 STA    V1+3
0856 A9 19                    LDA    #$19
0858 8D 05 D4                 STA    V1+5
;
085B A9 00                    LDA    #0
085D 8D 24 18                 STA    ZIPFREQ
;
; set difficulties
0860 20 6B 13                 JSR    CONFIG
; draw screen
0863 20 E9 11                 JSR    DRAW
0866 A9 30                    LDA    #$30
0868 A0 06                    LDY    #6
086A 99 C8 07  SINITLUP       STA    SCORE,Y
086D 88                       DEY
086E D0 FA                    BNE    SINITLUP
0870 8C 20 18                 STY    PLAYNUM
;
0873 AC 28 18                 LDY    MULTI
0876 30 12                    BMI    SPFREE
0878 A0 06                    LDY    #6
087A 99 E1 07  S2INIT         STA    SCORE2,Y
087D 88                       DEY
087E D0 FA                    BNE    S2INIT
0880 A9 32                    LDA    #$32
0882 8D C6 07                 STA    $7C6
0885 A9 31                    LDA    #$31
0887 8D DF 07                 STA    $7DF
;
; --- free all sprites
088A A9 FC     SPFREE         LDA    #$FC
088C 8D 03 18                 STA    INUSE
```

```
;
088F A9 0B                     LDA    #11
0891 A2 04                     LDX    #4
0893 9D DB 17  IVLUP           STA    VELOCE,X
0896 E8                        INX
0897 E8                        INX
0898 E0 10                     CPX    #$10
089A D0 F7                     BNE    IVLUP
;
; --- start interrupt
089C 20 7F 0D                  JSR    PLANE
;
089F A9 08                     LDA    #8
08A1 8D 21 18                  STA    QUAN
08A4 8D 22 18                  STA    QUAN+1
08A7 4C 3D 0B                  JMP    PRESET
;
; ***********************************************
; *               initializations complete              *
; ***********************************************
;
; --- check sprite collisions
08AA A9 00     MAINENT         LDA    #0
08AC 8D 01 18                  STA    FIRE
08AF 8D 02 18                  STA    LASTSHOT
;
08B2 20 61 0D  MAINLUP         JSR    ZIPPER
08B5 AD 0E 18                  LDA    LNCHTIME
08B8 C9 FF                     CMP    #$FF
08BA D0 22                     BNE    ML1
08BC 20 3C 0D                  JSR    WHOOSH
08BF AD 19 18                  LDA    DELTAX
08C2 C9 08                     CMP    #$8
08C4 90 04                     BCC    CC1
08C6 C9 F8                     CMP    #$F8
08C8 90 14                     BCC    ML1
08CA AD 1A 18  CC1             LDA    DELTAY
08CD C9 08                     CMP    #$08
08CF 90 04                     BCC    CC2
08D1 C9 F8                     CMP    #$F8
08D3 90 09                     BCC    ML1
08D5 20 46 0D  CC2             JSR    WHOOSHOF
08D8 20 79 0D                  JSR    ZIPOFF
08DB 4C 14 0B                  JMP    MISSUCC
;
;
; --- check fire button
08DE AD 01 18  ML1             LDA    FIRE
08E1 F0 03                     BEQ    ML2
08E3 20 83 0A                  JSR    LAUNCH
;
; --- check pause
08E6 78        ML2             SEI
08E7 A9 FD                     LDA    #$FD
08E9 8D 00 DC                  STA    $DC00
08EC AD 01 DC                  LDA    $DC01
08EF 29 80                     AND    #$80
08F1 F0 F3                     BEQ    ML2
08F3 A9 F7                     LDA    #$F7
08F5 8D 00 DC                  STA    $DC00
08F8 58                        CLI
;
; --- check shot clocks
08F9 A9 04                     LDA    #4
08FB 8D 1D 18                  STA    MASKTEMP
08FE A2 00                     LDX    #0
0900 BD FB 17  SCLUP           LDA    SHOTCLOCK,X
0903 C9 7F                     CMP    #$7F
0905 90 06                     BCC    SCL1
0907 AD 1D 18                  LDA    MASKTEMP
```

```
090A 20 6C 0A                  JSR    DISABLE
090D 0E 1D 18  SCL1            ASL    MASKTEMP
0910 E8                        INX
0911 E0 06                     CPX    #6
0913 D0 EB                     BNE    SCLUP
;
0915 A5 A1                     LDA    $A1
0917 CD 1F 18                  CMP    OLDJIF
091A F0 06                     BEQ    FTEND
091C 8D 1F 18                  STA    OLDJIF
091F 20 71 0C                  JSR    SCORTEN
0922           FTEND           =      *
;
; --- check chaser timer
0922 AD 0E 18                  LDA    LNCHTIME
0925 10 21                     BPL    CHASEND
0927 C9 C0                     CMP    #$C0
0929 F0 1D                     BEQ    CHASEND
092B C9 FF                     CMP    #$FF
092D F0 19                     BEQ    CHASEND
092F 20 F4 0B                  JSR    CHASLNCH
0932 90 07                     BCC    LNCHSUCC
0934 A9 C0                     LDA    #$C0
0936 8D 0E 18                  STA    LNCHTIME
0939 D0 0D                     BNE    CHASEND
093B A9 FF     LNCHSUCC        LDA    #$FF
093D 8D 0E 18                  STA    LNCHTIME
0940 A9 80                     LDA    #$80
0942 8D 19 18                  STA    DELTAX
0945 8D 1A 18                  STA    DELTAY
0948           CHASEND         =      *
;
; --- check if bases destroyed
0948 A0 09                     LDY    #9
094A A9 FF                     LDA    #$FF
094C D9 04 18  DBCLUP          CMP    BASCLOCK,Y
094F F0 0D                     BEQ    DBCEND
0951 88                        DEY
0952 D0 F8                     BNE    DBCLUP
0954 AD 0E 18                  LDA    LNCHTIME
0957 C9 C0                     CMP    #$C0
0959 D0 03                     BNE    DBCEND
095B 4C 04 0B                  JMP    ENDROUND
095E           DBCEND          =      *
;
;
; --- check sprite - background
095E AD 1F D0                  LDA    SBCREG
0961 8D 0F 18                  STA    REGTEMP
0964 29 01                     AND    #1
0966 F0 09                     BEQ    SBC1
0968 20 39 0A                  JSR    PLANCRSH
096B 20 46 0D                  JSR    WHOOSHOF
096E 4C 3D 0B                  JMP    PRESET
0971 AD 0F 18  SBC1            LDA    REGTEMP
0974 29 02                     AND    #2
0976 F0 03                     BEQ    SBC2
0978 20 13 0A                  JSR    CHASCRSH
097B AD 0F 18  SBC2            LDA    REGTEMP
097E 29 FC                     AND    #$FC
0980 D0 03                     BNE    BACKCHECK
0982 4C B2 08                  JMP    MAINLUP
;
; --- successful shot
0985 8D 0F 18  BACKCHECK       STA    REGTEMP
0988 A9 04                     LDA    #$4
098A 8D 1E 18                  STA    SHOTMASK
098D AA                        TAX
;
; --- which shot
098E AD 1E 18  SCHKLUP         LDA    SHOTMASK
```

```
0991 2D 0F 18            AND    REGTEMP
0994 F0 03               BEQ    SCH1
0996 20 A5 09            JSR    BASCHK
0999 0E 1E 18  SCH1      ASL    SHOTMASK
099C E8                  INX
099D E8                  INX
099E E0 10               CPX    #$10
09A0 D0 EC               BNE    SCHKLUP
09A2 4C B2 08            JMP    MAINLUP
;
; --- which rocket
09A5 2D 10 D0  BASCHK    AND    MSB
09A8 F0 02               BEQ    BASCHK1
09AA 38                  SEC
09AB 24                  .BYTE  $24
09AC 18        BASCHK1   CLC
09AD BD 00 D0            LDA    PLANEX,X
09B0 6A                  ROR
09B1 38                  SEC
09B2 E9 08               SBC    #$8
09B4 B0 02               BCS    BC1
09B6 A9 00               LDA    #0
09B8 C9 A0     BC1       CMP    #$A0
09BA 90 02               BCC    BC2
09BC A9 90               LDA    #$90
09BE 4A        BC2       LSR
09BF 4A                  LSR
09C0 4A                  LSR
09C1 4A                  LSR
09C2 A8                  TAY
;
; --- is it a rocket
09C3 B9 04 18            LDA    BASCLOCK,Y
09C6 C9 FF               CMP    #$FF
09C8 D0 48               BNE    NOROCKET
09CA A9 C0               LDA    #$C0
09CC 99 04 18            STA    BASCLOCK,Y
09CF A9 00               LDA    #0
09D1 9D C5 17            STA    PXVHI,X
09D4 9D C6 17            STA    PYVHI,X
09D7 A9 E2               LDA    #226
09D9 9D 01 D0            STA    PLANEY,X
09DC BD 00 D0            LDA    PLANEX,X
09DF 38                  SEC
09E0 E9 10               SBC    #$10
09E2 29 E0               AND    #$E0
09E4 18                  CLC
09E5 69 1C               ADC    #$1C
09E7 9D 00 D0            STA    PLANEX,X
09EA 20 8C 0C            JSR    SCOR50
09ED 8A                  TXA
09EE 4A                  LSR
09EF AA                  TAX
09F0 A9 40               LDA    #$40
09F2 9D F9 17            STA    SHOTCLOCK−2,X
09F5 A9 FF               LDA    #$FF
09F7 9D F8 07            STA    2040,X
09FA 8A                  TXA
09FB 0A                  ASL
09FC AA                  TAX
;
; --- erase it
09FD 20 31 0D            JSR    CRASH
0A00 98                  TYA
0A01 0A                  ASL
0A02 0A                  ASL
0A03 A8                  TAY
0A04 A9 60               LDA    #$60
0A06 99 71 07            STA    $771,Y
0A09 99 72 07            STA    $772,Y
```

```
0A0C 99 99 07            STA    $799,Y
0A0F 99 9A 07            STA    $79A,Y
;
0A12 60        NOROCKET  RTS
;
; --- chaser crash
0A13 AD 0E 18  CHASCRSH  LDA    LNCHTIME
0A16 C9 FF               CMP    #$FF
0A18 D0 1E               BNE    CCDUN
0A1A 20 A7 0C            JSR    SCOR100
0A1D 20 31 0D            JSR    CRASH
0A20 A9 00               LDA    #0
0A22 8D C7 17            STA    PXVHI+2
0A25 8D C8 17            STA    PYVHI+2
0A28 8D 0E 18            STA    LNCHTIME
0A2B A9 E2               LDA    #226
0A2D 8D 03 D0            STA    CHASEY
0A30 A9 02               LDA    #2
0A32 8D 28 D0            STA    $D028
0A35 20 46 0D            JSR    WHOOSHOF
0A38 60        CCDUN     RTS
;
0A39 A9 00     PLANCRSH  LDA    #0
0A3B 8D C5 17            STA    PXVHI
0A3E 8D C6 17            STA    PYVHI
0A41 A9 E2               LDA    #226
0A43 8D 01 D0            STA    PLANEY
0A46 A9 02               LDA    #2
0A48 8D 27 D0            STA    $D027
0A4B A9 01               LDA    #1
0A4D 0D 1C D0            ORA    $D01C
0A50 8D 1C D0            STA    $D01C
;
0A53 20 79 0D            JSR    ZIPOFF
0A56 20 31 0D            JSR    CRASH
0A59 A0 C0               LDY    #$C0
0A5B 84 A2               STY    $A2
0A5D AD 1F D0  PCWAIT    LDA    SBCREG
0A60 29 02               AND    #2
0A62 F0 03               BEQ    PCFIN
0A64 20 13 0A            JSR    CHASCRSH
0A67 A4 A2     PCFIN     LDY    $A2
0A69 D0 F2               BNE    PCWAIT
0A6B 60                  RTS
;
; --- turn off shot
0A6C 0D 03 18  DISABLE   ORA    INUSE
0A6F 8D 03 18            STA    INUSE
0A72 AD 1D 18            LDA    MASKTEMP
0A75 49 FF               EOR    #$FF
0A77 2D 15 D0            AND    SPRITEN
0A7A 8D 15 D0            STA    SPRITEN
0A7D A9 FE               LDA    #$FE
0A7F 9D FA 07            STA    2042,X
0A82 60                  RTS
;
; --- launch another shot
0A83 AD 02 18  LAUNCH    LDA    LASTSHOT
0A86 C9 1C               CMP    #$1C
0A88 B0 06               BCS    LLAUNCH
0A8A A9 00               LDA    #0
0A8C 8D 01 18            STA    FIRE
0A8F 60                  RTS
0A90 AD 03 18  LLAUNCH   LDA    INUSE
0A93 D0 01               BNE    LAUNCH1
0A95 60                  RTS
;
; --- find unused shot
0A96 A9 00     LAUNCH1   LDA    #0
0A98 8D 01 18            STA    FIRE
```

```
0A9B 8D 02 18             STA   LASTSHOT
0A9E A9 04                LDA   #4
0AA0 AA                   TAX
0AA1 A8                   TAY
;
0AA2 2D 03 18  LNCHLUP    AND   INUSE
0AA5 D0 0A                BNE   LNCHIT
0AA7 98                   TYA
0AA8 0A                   ASL
0AA9 A8                   TAY
0AAA E8                   INX
0AAB E8                   INX
0AAC E0 10                CPX   #$10
0AAE D0 F2                BNE   LNCHLUP
0AB0 60                   RTS
;
; --- mark it and turn it on
0AB1 8D 1D 18  LNCHIT     STA   MASKTEMP
0AB4 0D 15 D0             ORA   SPRITEN
0AB7 8D 15 D0             STA   SPRITEN
0ABA AD 1D 18  LAB1       LDA   MASKTEMP
0ABD 49 FF                EOR   #$FF
0ABF A8                   TAY
0AC0 2D 03 18             AND   INUSE
0AC3 8D 03 18             STA   INUSE
;
; --- same side of seam
0AC6 AD 10 D0             LDA   MSB
0AC9 29 01                AND   #1
0ACB F0 0C                BEQ   LI1
0ACD AD 1D 18             LDA   MASKTEMP
0AD0 0D 10 D0             ORA   MSB
0AD3 8D 10 D0             STA   MSB
0AD6 4C E0 0A             JMP   LI2
0AD9 98        LI1        TYA
0ADA 2D 10 D0             AND   MSB
0ADD 8D 10 D0             STA   MSB
;
; --- same place, same speed
0AE0 AD C5 17  LI2        LDA   PXVHI
0AE3 9D C5 17             STA   PXVHI,X
0AE6 AD C6 17             LDA   PYVHI
0AE9 9D C6 17             STA   PYVHI,X
0AEC AD 00 D0             LDA   PLANEX
0AEF 9D 00 D0             STA   PLANEX,X
0AF2 AD 01 D0             LDA   PLANEY
0AF5 9D 01 D0             STA   PLANEY,X
;
; --- set disable timer
0AF8 8A                   TXA
0AF9 4A                   LSR
0AFA A8                   TAY
0AFB A9 00                LDA   #0
0AFD 99 F9 17             STA   SHOTCLOCK-2,Y
;
0B00 58                   CLI
0B01 4C 4C 0D             JMP   ZIP
;
0B04 AE 20 18  ENDROUND   LDX   PLAYNUM
0B07 FE 21 18             INC   QUAN,X
0B0A A0 0A                LDY   #10
0B0C 20 A7 0C  BONUSLUP   JSR   SCOR100
0B0F 88                   DEY
0B10 D0 FA                BNE   BONUSLUP
0B12 F0 29                BEQ   PRESET
;
; --- stop motion
0B14 A9 00     MISSUCC    LDA   #0
0B16 8D C5 17             STA   PXVHI
0B19 8D C7 17             STA   CXVHI
```

```
0B1C 8D C6 17             STA   PYVHI
0B1F 8D C8 17             STA   CYVHI
0B22 20 31 0D             JSR   CRASH
0B25 AD 1C D0             LDA   $D01C
0B28 09 01                ORA   #1
0B2A 8D 1C D0             STA   $D01C
0B2D A9 02                LDA   #2
0B2F 8D 27 D0             STA   $D027
0B32 8D 28 D0             STA   $D028
0B35 A9 C0                LDA   #$C0
0B37 85 A2                STA   $A2
0B39 A5 A2     SUCCWAIT   LDA   $A2
0B3B D0 FC                BNE   SUCCWAIT
;
; --- reset positions
0B3D 78        PRESET     SEI
0B3E A9 64                LDA   #100
0B40 8D 00 D0             STA   PLANEX
0B43 A9 64                LDA   #100
0B45 8D 01 D0             STA   PLANEY
;
0B48 A9 00                LDA   #0
0B4A 8D 10 D0             STA   MSB
0B4D A9 01                LDA   #1
0B4F 8D 15 D0             STA   SPRITEN
0B52 A9 F0                LDA   #240
0B54 8D F8 07             STA   2040
0B57 A9 0F                LDA   #$F
0B59 8D 27 D0             STA   $D027
0B5C A9 FE                LDA   #$FE
0B5E 2D 1C D0             AND   $D01C
0B61 8D 1C D0             STA   $D01C
;
0B64 A9 00                LDA   #0
0B66 85 A0                STA   $A0
0B68 85 A1                STA   $A1
0B6A 85 A2                STA   $A2
0B6C 8D 1F 18             STA   OLDJIF
;
; --- establish bases
0B6F 20 52 13             JSR   DRWSCREEN
0B72 A2 09                LDX   #9
0B74 A9 FF                LDA   #$FF
0B76 9D 04 18  NEWBASLUP  STA   BASCLOCK,X
0B79 CA                   DEX
0B7A 10 FA                BPL   NEWBASLUP
;
0B7C A9 00                LDA   #0
0B7E 8D 0E 18             STA   LNCHTIME
0B81 AD 1F D0             LDA   SBCREG
0B84 AD 1E D0             LDA   SSCREG
;
; --- change players
0B87 2C 28 18  CHANGEP    BIT   MULTI
0B8A 30 1C                BMI   LIFELOSS
0B8C A0 09                LDY   #9
0B8E B9 C5 07  SWITCHLP   LDA   SCORE-3,Y
0B91 AA                   TAX
0B92 B9 DE 07             LDA   SCORE2-3,Y
0B95 99 C5 07             STA   SCORE-3,Y
0B98 8A                   TXA
0B99 99 DE 07             STA   SCORE2-3,Y
0B9C 88                   DEY
0B9D D0 EF                BNE   SWITCHLP
0B9F A9 01                LDA   #1
0BA1 38                   SEC
0BA2 ED 20 18             SBC   PLAYNUM
0BA5 8D 20 18             STA   PLAYNUM
;
0BA8 AE 20 18  LIFELOSS   LDX   PLAYNUM
```

```
0BAB BD 21 18             LDA   QUAN,X
0BAE D0 13                BNE   PLAYIT
0BB0 A0 00                LDY   #0
0BB2 2C 28 18             BIT   MULTI
0BB5 30 01                BMI   ONEPLAY
0BB7 C8                   INY
0BB8 B9 21 18  ONEPLAY    LDA   QUAN,Y
0BBB D0 CA                BNE   CHANGEP
0BBD 88                   DEY
0BBE 10 F8                BPL   ONEPLAY
0BC0 4C AE 0C             JMP   ENDALL
0BC3 DE 21 18  PLAYIT     DEC   QUAN,X
0BC6 BD 21 18             LDA   QUAN,X
0BC9 18                   CLC
0BCA 69 31                ADC   #$31
0BCC 8D D6 07             STA   $7D6
;
; --- wait for button
0BCF AD 01 DC  WAITLUP    LDA   JOY1
0BD2 2D 00 DC             AND   JOY2
0BD5 29 10                AND   #$10
0BD7 D0 F6                BNE   WAITLUP
;
0BD9 AD 01 DC  WAITLUP1   LDA   JOY1
0BDC 2D 00 DC             AND   JOY2
0BDF 29 10                AND   #$10
0BE1 F0 F6                BEQ   WAITLUP1
;
; --- start again
0BE3 A9 00                LDA   #0
0BE5 8D C6 17             STA   PYVHI
0BE8 8D C7 17             STA   CXVHI
0BEB A9 55                LDA   #85
0BED 8D C5 17             STA   PXVHI
0BF0 58                   CLI
;
0BF1 4C AA 08             JMP   MAINENT
;
; --- launch chaser
0BF4 A0 09     CHASLNCH   LDY   #9
0BF6 B9 04 18  FBASLUP    LDA   BASCLOCK,Y
0BF9 C9 FF                CMP   #$FF
0BFB F0 0D                BEQ   CHASIT
0BFD 88                   DEY
0BFE 10 F6                BPL   FBASLUP
0C00 A9 FD                LDA   #$FD
0C02 2D 15 D0             AND   SPRITEN
0C05 8D 15 D0             STA   SPRITEN
0C08 38                   SEC
0C09 60                   RTS
;
0C0A A9 C0     CHASIT     LDA   #$C0
0C0C 99 04 18             STA   BASCLOCK,Y
0C0F 98                   TYA
0C10 0A                   ASL
0C11 0A                   ASL
0C12 A8                   TAY
0C13 A9 60                LDA   #$60
0C15 99 71 07             STA   $771,Y
0C18 99 72 07             STA   $772,Y
0C1B 99 99 07             STA   $799,Y
0C1E 99 9A 07             STA   $79A,Y
;
0C21 98                   TYA
0C22 0A                   ASL
0C23 0A                   ASL
0C24 0A                   ASL
0C25 48                   PHA
0C26 90 0A                BCC   CMSB1
0C28 A9 02                LDA   #2
```

```
0C2A 0D 10 D0             ORA   MSB
0C2D 8D 10 D0             STA   MSB
0C30 D0 08                BNE   CMSB2
0C32 A9 FD     CMSB1      LDA   #$FD
0C34 2D 10 D0             AND   MSB
0C37 8D 10 D0             STA   MSB
;
0C3A 68        CMSB2      PLA
0C3B 18                   CLC
0C3C 69 1C                ADC   #$1C
0C3E 8D 02 D0             STA   CHASEX
0C41 A9 E2                LDA   #226
0C43 8D 03 D0             STA   CHASEY
;
0C46 A9 F7                LDA   #247
0C48 8D F9 07             STA   2041
0C4B A9 AB                LDA   #171
0C4D 8D C8 17             STA   CYVHI
;
0C50 AD 1F D0             LDA   SBCREG
0C53 2C 11 D0  RWAITLUP   BIT   $D011
0C56 30 FB                BMI   RWAITLUP
0C58 AD 12 D0             LDA   $D012
0C5B C9 F2                CMP   #$F2
0C5D D0 F4                BNE   RWAITLUP
;
0C5F A9 07                LDA   #7
0C61 8D 28 D0             STA   $D028
0C64 A9 02                LDA   #2
0C66 0D 15 D0             ORA   SPRITEN
0C69 8D 15 D0             STA   SPRITEN
;
0C6C AD 1F D0             LDA   SBCREG
0C6F 18                   CLC
0C70 60                   RTS
;
; --- add to score and update
0C71 48        SCORTEN    PHA
0C72 8A                   TXA
0C73 48                   PHA
0C74 A2 05                LDX   #5
0C76 FE C8 07  CARRYLUP   INC   SCORE,X
0C79 A9 3A                LDA   #$3A
0C7B DD C8 07             CMP   SCORE,X
0C7E D0 08                BNE   CARRYEND
0C80 A9 30                LDA   #$30
0C82 9D C8 07             STA   SCORE,X
0C85 CA                   DEX
0C86 D0 EE                BNE   CARRYLUP
0C88 68        CARRYEND   PLA
0C89 AA                   TAX
0C8A 68                   PLA
0C8B 60                   RTS
;
0C8C 48        SCOR50     PHA
0C8D 8A                   TXA
0C8E 48                   PHA
0C8F AD CD 07             LDA   SCORE+5
0C92 18                   CLC
0C93 69 05                ADC   #5
0C95 C9 3A                CMP   #$3A
0C97 B0 05                BCS   CARRY50
0C99 8D CD 07             STA   SCORE+5
0C9C D0 EA                BNE   CARRYEND
0C9E E9 0A     CARRY50    SBC   #$A
0CA0 8D CD 07             STA   SCORE+5
0CA3 A2 04                LDX   #4
0CA5 D0 CF                BNE   CARRYLUP
;
0CA7 48        SCOR100    PHA
```

```
0CA8 8A                 TXA
0CA9 48                 PHA
0CAA A2 04              LDX     #4
0CAC D0 C8              BNE     CARRYLUP
;
; --- end of game
0CAE 78        ENDALL   SEI
0CAF A9 31              LDA     #$31
0CB1 8D 14 03           STA     $314
0CB4 A9 EA              LDA     #$EA
0CB6 8D 15 03           STA     $315
0CB9 58                 CLI
0CBA A9 30              LDA     #$30
0CBC 8D D6 07           STA     $7D6
0CBF 20 84 FF           JSR     $FF84
0CC2 A0 10              LDY     #MSGEND-MSG
0CC4 B9 20 0D  MSGLUP   LDA     MSG,Y
0CC7 C9 40              CMP     #$40
0CC9 90 03              BCC     MSGLPUT
0CCB 38                 SEC
0CCC E9 40              SBC     #$40
0CCE 99 0B 04  MSGLPUT  STA     $40B,Y
0CD1 A9 03              LDA     #3
0CD3 99 0B D8           STA     $D80B,Y
0CD6 88                 DEY
0CD7 D0 EB              BNE     MSGLUP
0CD9 C8                 INY
0CDA 8C 25 18           STY     RESP
0CDD AD 25 18  GETLUP   LDA     RESP
0CE0 D0 0C              BNE     GLTR
0CE2 A9 2D              LDA     #$2D
0CE4 8D 19 04           STA     $419
0CE7 A9 3E              LDA     #$3E
0CE9 8D 1A 04           STA     $41A
0CEC D0 0A              BNE     GETLUP2
0CEE A9 3C     GLTR     LDA     #$3C
0CF0 8D 19 04           STA     $419
0CF3 A9 2D              LDA     #$2D
0CF5 8D 1A 04           STA     $41A
0CF8 AD 00 DC  GETLUP2  LDA     JOY2
0CFB 2D 01 DC           AND     JOY1
0CFE 4A                 LSR
0CFF 4A                 LSR
0D00 4A                 LSR
0D01 B0 05              BCS     GL1
0D03 A0 01              LDY     #1
0D05 8C 25 18           STY     RESP
0D08 4A        GL1      LSR
0D09 B0 05              BCS     GL2
0D0B A0 00              LDY     #0
0D0D 8C 25 18           STY     RESP
0D10 4A        GL2      LSR
0D11 B0 CA              BCS     GETLUP
0D13 20 81 FF           JSR     $FF81
0D16 AD 25 18           LDA     RESP
0D19 F0 03              BEQ     GET1
0D1B 4C 0D 08           JMP     START
0D1E 85 C6     GET1     STA     $C6
0D20 00                 BRK
;
0D21           MSG      =       *-1
0D21 50 4C 41           .ASC    "play again? y n"
0D31           MSGEND   =       *-1
;
0D31 A9 80     CRASH    LDA     #$80
0D33 8D 0B D4           STA     V2+4
0D36 A9 81              LDA     #$81
0D38 8D 0B D4           STA     V2+4
0D3B 60                 RTS
;
```

```
0D3C A9 FF     WHOOSH     LDA     #$FF
0D3E 38                   SEC
0D3F ED 03 D0             SBC     CHASEY
0D42 8D 0F D4             STA     V3+1
0D45 60                   RTS
;
0D46 A9 00     WHOOSHOF   LDA     #0
0D48 8D 0F D4             STA     V3+1
0D4B 60                   RTS
;
0D4C A9 18     ZIP        LDA     #$18
0D4E 8D 24 18             STA     ZIPFREQ
0D51 A5 A2                LDA     $A2
0D53 8D 23 18             STA     ZIPTIME
0D56 A9 40                LDA     #$40
0D58 8D 04 D4             STA     V1+4
0D5B A9 41                LDA     #$41
0D5D 8D 04 D4             STA     V1+4
0D60 60                   RTS
;
0D61 A5 A2     ZIPPER     LDA     $A2
0D63 CD 23 18             CMP     ZIPTIME
0D66 F0 10                BEQ     ZIPEND
0D68 8D 23 18             STA     ZIPTIME
0D6B AD 24 18             LDA     ZIPFREQ
0D6E C9 30                CMP     #$30
0D70 B0 07                BCS     ZIPOFF
0D72 8D 01 D4             STA     V1+1
0D75 EE 24 18             INC     ZIPFREQ
;
0D78 60        ZIPEND     RTS
;
0D79 A9 00     ZIPOFF     LDA     #0
0D7B 8D 01 D4             STA     V1+1
0D7E 60                   RTS
;
; --- set up interrupt
0D7F 78        PLANE      SEI
0D80 A9 91                LDA     #<INTLUP
0D82 8D 14 03             STA     $314
0D85 A9 0D                LDA     #>INTLUP
0D87 8D 15 03             STA     $315
0D8A A9 0E                LDA     #14
0D8C 8D 05 DC             STA     56325
0D8F 58                   CLI
;
0D90 60                   RTS
;
;************************************************
;*                interrrupt movement code       *
;************************************************
;
0D91           INTLUP     =       *
;
; --- read joystick
; --- & change velocities
0D91 AD 00 DC             LDA     JOY2
0D94 2D 01 DC             AND     JOY1
0D97 8D 26 18             STA     JTEMP
0D9A AD C5 17             LDA     PXVHI
0D9D 0D C6 17             ORA     PYVHI
0DA0 D0 03                BNE     PM1
0DA2 4C 39 0E             JMP     CHASE
;
0DA5 AD 26 18  PM1        LDA     JTEMP
0DA8 29 10                AND     #$10
0DAA D0 03                BNE     NOTJSTICK
0DAC EE 01 18             INC     FIRE
;
0DAF AD 26 18  NOTJSTICK  LDA     JTEMP
```

```
0DB2 29 03                 AND   #3
0DB4 C9 03                 CMP   #3
0DB6 D0 03                 BNE   MAKPINCR
0DB8 4C 39 0E              JMP   CHASE
;
; --- make x velocity increment
0DBB AD D9 17   MAKPINCR   LDA   PRADIUS
0DBE 8D 14 18              STA   MULTEMP
0DC1 AD C5 17              LDA   PXVHI
0DC4 8D 12 18              STA   X1
0DC7 20 A7 11              JSR   MULTIPLY
0DCA AD 26 18              LDA   JTEMP
0DCD 29 02                 AND   #2
0DCF D0 16                 BNE   JDOWN1
;
; --- move up
0DD1 38                    SEC
0DD2 AD D6 17              LDA   PYVLO
0DD5 ED 10 18              SBC   PROD
0DD8 8D D6 17              STA   PYVLO
0DDB AD C6 17              LDA   PYVHI
0DDE ED 11 18              SBC   PROD+1
0DE1 8D C6 17              STA   PYVHI
;
0DE4 4C FA 0D              JMP   MAKPYINC
; --- move down
0DE7 18         JDOWN1     CLC
0DE8 AD D6 17              LDA   PYVLO
0DEB 6D 10 18              ADC   PROD
0DEE 8D D6 17              STA   PYVLO
0DF1 AD C6 17              LDA   PYVHI
0DF4 6D 11 18              ADC   PROD+1
0DF7 8D C6 17              STA   PYVHI
;
0DFA            MAKPYINC   =     *
;
; --- make y velocity increment
0DFA AD D9 17              LDA   PRADIUS
0DFD 8D 14 18              STA   MULTEMP
0E00 AD C6 17              LDA   PYVHI
0E03 8D 12 18              STA   X1
0E06 20 A7 11              JSR   MULTIPLY
0E09 AD 26 18              LDA   JTEMP
0E0C 29 02                 AND   #2
0E0E D0 16                 BNE   JDOWN2
;
; --- move up
0E10 18                    CLC
0E11 AD D5 17              LDA   PXVLO
0E14 6D 10 18              ADC   PROD
0E17 8D D5 17              STA   PXVLO
0E1A AD C5 17              LDA   PXVHI
0E1D 6D 11 18              ADC   PROD+1
0E20 8D C5 17              STA   PXVHI
;
0E23 4C 39 0E              JMP   CHASE
; --- move down
0E26 38         JDOWN2     SEC
0E27 AD D5 17              LDA   PXVLO
0E2A ED 10 18              SBC   PROD
0E2D 8D D5 17              STA   PXVLO
0E30 AD C5 17              LDA   PXVHI
0E33 ED 11 18              SBC   PROD+1
0E36 8D C5 17              STA   PXVHI
;
; ************************************************
;*             chase plane with missile            *
; ************************************************
;
0E39            CHASE      =     *
```

```
;
;
; --- find vector from
; --- chaser to plane
0E39 AD 00 D0              LDA   PLANEX
0E3C 38                    SEC
0E3D ED 02 D0              SBC   CHASEX
0E40 90 02                 BCC   FS1
0E42 18                    CLC
0E43 24                    .BYTE $24
0E44 38         FS1        SEC
0E45 6A                    ROR
0E46 8D 19 18              STA   DELTAX
;
0E49 AD 01 D0              LDA   PLANEY
0E4C 38                    SEC
0E4D ED 03 D0              SBC   CHASEY
0E50 90 02                 BCC   FS2
0E52 18                    CLC
0E53 24                    .BYTE $24
0E54 38         FS2        SEC
0E55 6A                    ROR
0E56 8D 1A 18              STA   DELTAY
;
; --- adjust for seam
0E59 A9 03                 LDA   #3
0E5B 2D 10 D0              AND   MSB
0E5E F0 2E                 BEQ   SCALIT
0E60 C9 03                 CMP   #3
0E62 F0 2A                 BEQ   SCALIT
0E64 C9 01                 CMP   #1
0E66 F0 13                 BEQ   PLUSEAM
;
; --- adjust for wrap
0E68 AD 19 18              LDA   DELTAX
0E6B 38                    SEC
0E6C E9 80                 SBC   #$80
0E6E C9 A5                 CMP   #$A5
0E70 B0 03                 BCS   MSNO
0E72 18                    CLC
0E73 69 AC                 ADC   #$AC
0E75 8D 19 18   MSNO       STA   DELTAX
0E78 4C A6 0E              JMP   SCALDUN
;
0E7B AD 19 18   PLUSEAM    LDA   DELTAX
0E7E 18                    CLC
0E7F 69 80                 ADC   #$80
0E81 C9 55                 CMP   #$55
0E83 90 03                 BCC   PSNO
0E85 38                    SEC
0E86 E9 AC                 SBC   #$AC
0E88 8D 19 18   PSNO       STA   DELTAX
0E8B 4C A6 0E              JMP   SCALDUN
;
0E8E AD 19 18   SCALIT     LDA   DELTAX
0E91 30 09                 BMI   SCMI
0E93 C9 55                 CMP   #$55
0E95 30 0C                 BMI   SCNO
0E97 38                    SEC
0E98 E9 AC                 SBC   #$AC
0E9A 90 07                 BCC   SCNO
0E9C C9 A5      SCMI       CMP   #$A5
0E9E 10 03                 BPL   SCNO
0EA0 18                    CLC
0EA1 69 AC                 ADC   #$AC
0EA3 8D 19 18   SCNO       STA   DELTAX
;
0EA6            SCALDUN    =     *
;
0EA6 AD 0E 18              LDA   LNCHTIME
```

```
0EA9 30 03               BMI   ACTV
0EAB 4C FD 0F            JMP   UPDATPOS
;
; --- decide which way to turn
0EAE A9 00      ACTV     LDA   #0
0EB0 2C C8 17            BIT   CYVHI
0EB3 30 09               BMI   QNEG1
0EB5 2C C7 17            BIT   CXVHI
0EB8 10 0E               BPL   QVFIN
0EBA A9 01               LDA   #1
0EBC D0 0A               BNE   QVFIN
;
0EBE 2C C7 17   QNEG1    BIT   CXVHI
0EC1 10 03               BPL   QPOS1
0EC3 A9 02               LDA   #2
0EC5 2C                  .BYTE $2C
0EC6 A9 03      QPOS1    LDA   #3
;
0EC8 8D 17 18   QVFIN    STA   VQUAD
0ECB A9 00               LDA   #0
0ECD 2C 1A 18            BIT   DELTAY
0ED0 30 09               BMI   QNEG2
0ED2 2C 19 18            BIT   DELTAX
0ED5 10 0E               BPL   QDFIN
0ED7 A9 01               LDA   #1
0ED9 D0 0A               BNE   QDFIN
;
0EDB 2C 19 18   QNEG2    BIT   DELTAX
0EDE 10 03               BPL   QPOS2
0EE0 A9 02               LDA   #2
0EE2 2C                  .BYTE $2C
0EE3 A9 03      QPOS2    LDA   #3
;
0EE5 8D 18 18   QDFIN    STA   DQUAD
;
; --- compare quadrants
0EE8 AD 17 18            LDA   VQUAD
0EEB 38                  SEC
0EEC ED 18 18            SBC   DQUAD
0EEF 4A                  LSR
0EF0 90 08               BCC   COMMORE
0EF2 29 01               AND   #1
0EF4 8D 16 18            STA   DIRTEMP
0EF7 4C 83 0F            JMP   CMF3
;
0EFA F0 12      COMMORE  BEQ   NOSWAP1
;
; --- if in opposite quads
0EFC A9 00               LDA   #0
0EFE 38                  SEC
0EFF ED 19 18            SBC   DELTAX
0F02 8D 19 18            STA   DELTAX
0F05 A9 00               LDA   #0
0F07 38                  SEC
0F08 ED 1A 18            SBC   DELTAY
0F0B 8D 1A 18            STA   DELTAY
;
0F0E A9 00      NOSWAP1  LDA   #0
0F10 8D 1B 18            STA   XSCALE
0F13 8D 1C 18            STA   YSCALE
;
0F16 AD 19 18            LDA   DELTAX
0F19 0D 1A 18            ORA   DELTAY
0F1C D0 03               BNE   SCALLUP
0F1E 4C FD 0F            JMP   UPDATPOS
;
; --- while xscale<xvel
; --- & yscale<yvel
0F21 AD 1B 18   SCALLUP  LDA   XSCALE
0F24 18                  CLC
```

```
0F25 6D 19 18            ADC   DELTAX
0F28 8D 1B 18            STA   XSCALE
;
0F2B AD 1C 18            LDA   YSCALE
0F2E 18                  CLC
0F2F 6D 1A 18            ADC   DELTAY
0F32 8D 1C 18            STA   YSCALE
;
0F35 AD 1B 18            LDA   XSCALE
0F38 38                  SEC
0F39 ED C7 17            SBC   CXVHI
0F3C 4D C7 17            EOR   CXVHI
0F3F 10 0E               BPL   COMMFIN1
0F41 AD 1C 18            LDA   YSCALE
0F44 38                  SEC
0F45 ED C8 17            SBC   CYVHI
0F48 4D C8 17            EOR   CYVHI
0F4B 30 D4               BMI   SCALLUP
0F4D 10 17               BPL   COMMFIN
; --- wend
;
; --- decide which way to go
0F4F AD 1C 18   COMMFIN1 LDA   YSCALE
0F52 38                  SEC
0F53 4D C8 17            EOR   CYVHI
0F56 30 0B               BMI   MOVOK
0F58 AC 17 18            LDY   VQUAD
0F5B CC 18 18            CPY   DQUAD
0F5E D0 03               BNE   MOVOK
0F60 4C FD 0F            JMP   UPDATPOS
0F63 A9 00      MOVOK    LDA   #0
0F65 2C                  .BYTE $2C
0F66 A9 01      COMMFIN  LDA   #1
;
; --- adjust for various signs
0F68            SADJ     =     *
0F68 2C 19 18            BIT   DELTAX
0F6B 10 02               BPL   CMF1
0F6D 49 01               EOR   #1
0F6F 2C 1A 18   CMF1     BIT   DELTAY
0F72 10 02               BPL   CMF2
0F74 49 01               EOR   #1
0F76 AC 17 18   CMF2     LDY   VQUAD
0F79 CC 18 18            CPY   DQUAD
0F7C F0 05               BEQ   CMF3
0F7E 49 01               EOR   #1
0F80 8D 16 18            STA   DIRTEMP
;
; --- make x increment
0F83 AD DA 17   CMF3     LDA   CRADIUS
0F86 8D 14 18            STA   MULTEMP
0F89 AD C7 17            LDA   CXVHI
0F8C 8D 12 18            STA   X1
0F8F 20 A7 11            JSR   MULTIPLY
0F92 AD 16 18            LDA   DIRTEMP
0F95 D0 16               BNE   MOVDOWN1
;
; --- move up
0F97 38                  SEC
0F98 AD D8 17            LDA   CYVLO
0F9B ED 10 18            SBC   PROD
0F9E 8D D8 17            STA   CYVLO
0FA1 AD C8 17            LDA   CYVHI
0FA4 ED 11 18            SBC   PROD+1
0FA7 8D C8 17            STA   CYVHI
0FAA 4C C0 0F            JMP   MAKCINCR
;
; --- move down
0FAD            MOVDOWN1 =     *
0FAD 18                  CLC
```

```
0FAE AD D8 17              LDA    CYVLO
0FB1 6D 10 18              ADC    PROD
0FB4 8D D8 17              STA    CYVLO
0FB7 AD C8 17              LDA    CYVHI
0FBA 6D 11 18              ADC    PROD+1
0FBD 8D C8 17              STA    CYVHI
;
; --- make y increment
0FC0 AD DA 17  MAKCINCR    LDA    CRADIUS
0FC3 8D 14 18              STA    MULTEMP
0FC6 AD C8 17              LDA    CYVHI
0FC9 8D 12 18              STA    X1
0FCC 20 A7 11              JSR    MULTIPLY
0FCF AD 16 18              LDA    DIRTEMP
0FD2 D0 16                 BNE    MOVDOWN2
;
; --- move up
0FD4 18                    CLC
0FD5 AD D7 17              LDA    CXVLO
0FD8 6D 10 18              ADC    PROD
0FDB 8D D7 17              STA    CXVLO
0FDE AD C7 17              LDA    CXVHI
0FE1 6D 11 18              ADC    PROD+1
0FE4 8D C7 17              STA    CXVHI
;
0FE7 4C FD 0F              JMP    UPDATPOS
;
; --- move down
0FEA           MOVDOWN2    =      *
0FEA 38                    SEC
0FEB AD D7 17              LDA    CXVLO
0FEE ED 10 18              SBC    PROD
0FF1 8D D7 17              STA    CXVLO
0FF4 AD C7 17              LDA    CXVHI
0FF7 ED 11 18              SBC    PROD+1
0FFA 8D C7 17              STA    CXVHI
;
; ***********************************************
;*             update object positions          *
; ***********************************************
;
0FFD           UPDATPOS    =      *
;
0FFD A2 00                 LDX    #0
;
; --- initialize factors
0FFF BD DB 17  UXMOV       LDA    VELOCE,X
1002 8D 14 18              STA    MULTEMP
1005 BD C5 17              LDA    XVHI,X
1008 8D 12 18              STA    X1
100B 20 A7 11              JSR    MULTIPLY
; --- update x position
100E A9 00                 LDA    #$00
1010 8D 13 18              STA    MSBTEMP
1013 BD B5 17              LDA    SETMASK,X
1016 2D 10 D0              AND    MSB
1019 F0 03                 BEQ    LEFTSIDE
101B EE 13 18              INC    MSBTEMP
101E 18        LEFTSIDE    CLC
101F BD EB 17              LDA    XLO,X
1022 6D 10 18              ADC    PROD
1025 9D EB 17              STA    XLO,X
1028 BD 00 D0              LDA    SPRITEX,X
102B 6D 11 18              ADC    PROD+1
102E 9D 00 D0              STA    SPRITEX,X
1031 90 03                 BCC    NM1
1033 EE 13 18              INC    MSBTEMP
1036 2C 11 18  NM1         BIT    PROD+1
1039 10 03                 BPL    NM2
103B CE 13 18              DEC    MSBTEMP
```

```
103E           NM2         =      *
;
103E 2C 13 18              BIT    MSBTEMP
1041 10 0E                 BPL    NOTLEFT
1043 A9 01                 LDA    #1
1045 8D 13 18              STA    MSBTEMP
1048 18                    CLC
1049 BD 00 D0              LDA    SPRITEX,X
104C 69 58                 ADC    #$58
104E 9D 00 D0              STA    SPRITEX,X
;
1051 A9 01     NOTLEFT     LDA    #1
1053 CD 13 18              CMP    MSBTEMP
1056 D0 13                 BNE    NOTRIGHT
1058 BD 00 D0              LDA    SPRITEX,X
105B C9 58                 CMP    #$58
105D 90 0C                 BCC    NOTRIGHT
105F CE 13 18              DEC    MSBTEMP
1062 38                    SEC
1063 BD 00 D0              LDA    SPRITEX,X
1066 E9 58                 SBC    #$58
1068 9D 00 D0              STA    SPRITEX,X
;
106B AD 13 18  NOTRIGHT    LDA    MSBTEMP
106E D0 0C                 BNE    SETMSB
1070 BD B6 17              LDA    CLRMASK,X
1073 2D 10 D0              AND    MSB
1076 8D 10 D0              STA    MSB
1079 4C 85 10              JMP    YMOVE
;
107C BD B5 17  SETMSB      LDA    SETMASK,X
107F 0D 10 D0              ORA    MSB
1082 8D 10 D0              STA    MSB
;
; --- initialize factors
1085 BD DB 17  YMOVE       LDA    VELOCE,X
1088 8D 14 18              STA    MULTEMP
108B BD C6 17              LDA    YVHI,X
108E 8D 12 18              STA    X1
1091 20 A7 11              JSR    MULTIPLY
;
; --- update y position
1094 18                    CLC
1095 BD EC 17              LDA    YLO,X
1098 6D 10 18              ADC    PROD
109B 9D EC 17              STA    YLO,X
109E BD 01 D0              LDA    SPRITEY,X
10A1 6D 11 18              ADC    PROD+1
10A4 C9 1D                 CMP    #$1D
10A6 B0 02                 BCS    YOK1
10A8 A9 1D                 LDA    #$1D
10AA C9 FA     YOK1        CMP    #$FA
10AC 90 02                 BCC    YOK
10AE A9 FA                 LDA    #$FA
10B0 9D 01 D0  YOK         STA    SPRITEY,X
;
10B3 E8                    INX
10B4 E8                    INX
10B5 E0 10                 CPX    #$10
10B7 F0 03                 BEQ    MFIN2
10B9 4C FF 0F              JMP    UXMOV
;
; --- tick timers
10BC 2C 02 18  MFIN2       BIT    LASTSHOT
10BF 30 03                 BMI    MFIN3
10C1 EE 02 18              INC    LASTSHOT
;
10C4 A2 05     MFIN3       LDX    #5
10C6 FE FB 17  INCLUP      INC    SHOTCLOCK,X
10C9 CA                    DEX
```

```
10CA 10 FA              BPL   INCLUP
;
10CC AD 0E 18  MFIN4    LDA   LNCHTIME
10CF 30 03              BMI   MFIN5
10D1 EE 0E 18           INC   LNCHTIME
;
; --- change sprite shapes
10D4 AD C7 17  MFIN5    LDA   CXVHI
10D7 D0 0D              BNE   MNZ1
10D9 AD C8 17           LDA   CYVHI
10DC D0 08              BNE   MNZ1
10DE A9 FF              LDA   #$FF
10E0 8D F9 07           STA   2041
10E3 4C 3C 11           JMP   MFIN6
;
10E6 A9 00     MNZ1     LDA   #0
10E8 8D 15 18           STA   SPRITMP
10EB AD C7 17           LDA   CXVHI
10EE 30 0C              BMI   DCHK1
10F0 C9 20              CMP   #32
10F2 90 11              BCC   DCHK2
10F4 A9 04              LDA   #4
10F6 8D 15 18           STA   SPRITMP
10F9 4C 05 11           JMP   DCHK2
;
10FC C9 E0     DCHK1    CMP   #224
10FE B0 05              BCS   DCHK2
1100 A9 06              LDA   #6
1102 8D 15 18           STA   SPRITMP
;
1105 AD C8 17  DCHK2    LDA   CYVHI
1108 30 10              BMI   DCHK3
110A C9 20              CMP   #32
110C 90 19              BCC   DCHK4
110E 18                 CLC
110F A9 08              LDA   #8
1111 6D 15 18           ADC   SPRITMP
1114 8D 15 18           STA   SPRITMP
1117 4C 27 11           JMP   DCHK4
;
111A C9 E0     DCHK3    CMP   #224
111C B0 09              BCS   DCHK4
111E 18                 CLC
111F A9 09              LDA   #9
1121 6D 15 18           ADC   SPRITMP
1124 8D 15 18           STA   SPRITMP
;
1127 AD 15 18  DCHK4    LDA   SPRITMP
112A C9 04              CMP   #4
112C D0 02              BNE   DCHK5
112E A9 0A              LDA   #10
1130 C9 06     DCHK5    CMP   #6
1132 D0 02              BNE   DCHK6
1134 A9 0B              LDA   #11
1136 18        DCHK6    CLC
1137 69 EE              ADC   #238
1139 8D F9 07           STA   2041
;
113C AD C5 17  MFIN6    LDA   PXVHI
113F D0 0D              BNE   MNZ2
1141 AD C6 17           LDA   PYVHI
1144 D0 08              BNE   MNZ2
1146 A9 FF              LDA   #$FF
1148 8D F8 07           STA   2040
114B 4C A4 11           JMP   MFIN7
;
114E A9 00     MNZ2     LDA   #0
1150 8D 15 18           STA   SPRITMP
1153 AD C5 17           LDA   PXVHI
1156 30 0C              BMI   ECHK1
```

```
1158 C9 23              CMP   #35
115A 90 11              BCC   ECHK2
115C A9 04              LDA   #4
115E 8D 15 18           STA   SPRITMP
1161 4C 6D 11           JMP   ECHK2
;
1164 C9 E0     ECHK1    CMP   #224
1166 B0 05              BCS   ECHK2
1168 A9 06              LDA   #6
116A 8D 15 18           STA   SPRITMP
;
116D AD C6 17  ECHK2    LDA   PYVHI
1170 30 10              BMI   ECHK3
1172 C9 20              CMP   #32
1174 90 19              BCC   ECHK4
1176 18                 CLC
1177 A9 08              LDA   #8
1179 6D 15 18           ADC   SPRITMP
117C 8D 15 18           STA   SPRITMP
117F 4C 8F 11           JMP   ECHK4
;
1182 C9 E0     ECHK3    CMP   #224
1184 B0 09              BCS   ECHK4
1186 18                 CLC
1187 A9 09              LDA   #9
1189 6D 15 18           ADC   SPRITMP
118C 8D 15 18           STA   SPRITMP
;
118F AD 15 18  ECHK4    LDA   SPRITMP
1192 C9 04              CMP   #4
1194 D0 02              BNE   ECHK5
1196 A9 0A              LDA   #10
1198 C9 06     ECHK5    CMP   #6
119A D0 02              BNE   ECHK6
119C A9 0B              LDA   #11
119E 18        ECHK6    CLC
119F 69 E6              ADC   #230
11A1 8D F8 07           STA   2040
;
11A4 4C 31 EA  MFIN7    JMP   $EA31
;
; ***********************************************
; 8 x 8 bit multiply
; ***********************************************
;
11A7           MULTIPLY =     *
11A7 AD 12 18           LDA   X1
11AA 8D 10 18           STA   PROD
11AD 30 03              BMI   XNEG
11AF A9 00              LDA   #0
; sign extend
11B1 2C                 .BYTE $2C
11B2 A9 FF     XNEG     LDA   #$FF
11B4 8D 11 18           STA   PROD+1
;
; --- shift out leading zeroes
11B7 A0 08              LDY   #$8
11B9 88        MAGLUP   DEY
11BA 0E 14 18           ASL   MULTEMP
11BD 90 FA              BCC   MAGLUP
11BF C0 00              CPY   #$0
11C1 F0 25              BEQ   NOMULT
;
; --- double product
11C3 0E 10 18  MULLUP   ASL   PROD
11C6 2E 11 18           ROL   PROD+1
;
; --- add factors to products
11C9 0E 14 18           ASL   MULTEMP
11CC 90 17              BCC   NOADD
```

```
11CE 18                   CLC
11CF AD 10 18             LDA   PROD
11D2 6D 12 18             ADC   X1
11D5 8D 10 18             STA   PROD
11D8 90 03                BCC   IMA
11DA EE 11 18             INC   PROD+1
11DD AD 12 18  IMA        LDA   X1
11E0 10 03                BPL   IMB
11E2 CE 11 18             DEC   PROD+1
11E5           IMB        =     *
;
11E5 88        NOADD      DEY
11E6 D0 DB                BNE   MULLUP
;
11E8 60        NOMULT     RTS
;
;**********************************************
;*               initialize video             *
;**********************************************
;
11E9           DRAW       =     *
;
11E9 20 B5 FF             JSR   CINT
;
; --- set multicolor mode
11EC 78                   SEI
11ED AD 16 D0             LDA   $D016
11F0 09 10                ORA   #$10
11F2 8D 16 D0             STA   $D016
11F5 A9 1D                LDA   #$1D
11F7 8D 18 D0             STA   $D018
11FA A9 07                LDA   #$7
11FC 8D 23 D0             STA   $D023
11FF A9 00                LDA   #0
1201 8D 25 D0             STA   $D025
1204 A9 07                LDA   #7
1206 8D 26 D0             STA   $D026
1209 A9 0F                LDA   #$F
120B 8D 27 D0             STA   $D027
;
; --- define sprites
120E A9 05                LDA   #<SPRITES
1210 85 FE                STA   SOURCE
1212 A9 15                LDA   #>SPRITES
1214 85 FF                STA   SOURCE+1
1216 A9 80                LDA   #$80
1218 85 FC                STA   DEST
121A A9 3B                LDA   #$3B
121C 85 FD                STA   DEST+1
121E A0 00                LDY   #0
;
1220 B1 FE     SPRILUP    LDA   (SOURCE),Y
1222 D0 18                BNE   NOTZEROS
1224 E6 FE                INC   SOURCE
1226 D0 02                BNE   ZL1
1228 E6 FF                INC   SOURCE+1
122A B1 FE     ZL1        LDA   (SOURCE),Y
122C AA                   TAX
122D A9 00     ZEROLOOP   LDA   #0
122F 91 FC                STA   (DEST),Y
1231 E6 FC                INC   DEST
1233 D0 02                BNE   ZL3
1235 E6 FD                INC   DEST+1
1237 CA        ZL3        DEX
1238 D0 F3                BNE   ZEROLOOP
123A F0 08                BEQ   ZNEXT
;
123C 91 FC     NOTZEROS   STA   (DEST),Y
123E E6 FC                INC   DEST
1240 D0 02                BNE   ZNEXT
```

```
1242 E6 FD                INC   DEST+1
1244 E6 FE     ZNEXT      INC   SOURCE
1246 D0 02                BNE   ZN2
1248 E6 FF                INC   SOURCE+1
124A A5 FC     ZN2        LDA   DEST
124C C9 40                CMP   #$40
124E D0 D0                BNE   SPRILUP
;
1250 A9 FE                LDA   #$FE
1252 8D 1C D0             STA   $D01C
;
; --- redefine characters
1255 A5 01                LDA   $1
1257 29 FB                AND   #$FB
1259 85 01                STA   $1
125B A0 00                LDY   #0
125D B9 00 DC  ROMLUP     LDA   $DC00,Y
1260 99 00 30             STA   $3000,Y
1263 B9 00 DD             LDA   $DD00,Y
1266 99 00 31             STA   $3100,Y
1269 C8                   INY
126A D0 F1                BNE   ROMLUP
126C A5 01                LDA   $1
126E 09 04                ORA   #$4
1270 85 01                STA   $1
;
1272 A0 00                LDY   #0
1274 B9 11 13  CHARLUP    LDA   CHARS,Y
1277 99 00 32             STA   $3200,Y
127A C8                   INY
127B C0 20                CPY   #$20
127D D0 F5                BNE   CHARLUP
;
127F A0 00                LDY   #0
1281 98                   TYA
1282 99 00 33  BLNKLUP    STA   $3300,Y
1285 C8                   INY
1286 C0 08                CPY   #$8
1288 D0 F8                BNE   BLNKLUP
;
; --- clear text screen
128A A9 04                LDA   #$4
128C 85 FF                STA   SCRPTR+1
128E A9 D8                LDA   #$D8
1290 85 FD                STA   COLPTR+1
1292 A9 00                LDA   #0
1294 85 FE                STA   SCRPTR
1296 85 FC                STA   COLPTR
1298 A8                   TAY
;
1299 A9 60     SCRLUP     LDA   #$60
129B 91 FE                STA   (SCRPTR),Y
129D A9 08                LDA   #8
129F 91 FC                STA   (COLPTR),Y
12A1 C8                   INY
12A2 D0 F5                BNE   SCRLUP
12A4 E6 FF                INC   SCRPTR+1
12A6 E6 FD                INC   COLPTR+1
12A8 A5 FF                LDA   SCRPTR+1
12AA C9 07                CMP   #7
12AC D0 EB                BNE   SCRLUP
;
12AE A9 60     SCRLUPA    LDA   #$60
12B0 99 00 07             STA   $700,Y
12B3 A9 08                LDA   #8
12B5 99 00 DB             STA   $DB00,Y
12B8 C8                   INY
12B9 C0 C0                CPY   #$C0
12BB D0 F1                BNE   SCRLUPA
;
```

```
12BD 20 52 13              JSR    DRWSCREEN
;
12C0 A0 00                 LDY    #0
12C2 A9 FE     FBLUP3      LDA    #$FE
12C4 99 FA 07              STA    2042,Y
12C7 A9 02                 LDA    #2
12C9 99 29 D0              STA    $D029,Y
12CC C8                    INY
12CD C0 06                 CPY    #6
12CF D0 F1                 BNE    FBLUP3
12D1 A9 FF                 LDA    #$FF
12D3 8D F8 07              STA    2040
12D6 8D F9 07              STA    2041
;
12D9 A0 00                 LDY    #0
12DB B9 31 13  TITLUP      LDA    TITLE,Y
12DE C9 40                 CMP    #$40
12E0 90 03                 BCC    TTL
12E2 38                    SEC
12E3 E9 40                 SBC    #$40
12E5 99 C0 07  TTL         STA    $7C0,Y
12E8 A9 03                 LDA    #3
12EA 99 C0 DB              STA    $DBC0,Y
12ED C8                    INY
12EE 2C 28 18              BIT    MULTI
12F1 10 06                 BPL    MTL
12F3 C0 18                 CPY    #$18
12F5 D0 E4                 BNE    TITLUP
12F7 F0 04                 BEQ    ENDIT
12F9 C0 21     MTL         CPY    #$21
12FB D0 DE                 BNE    TITLUP
;
12FD A9 03     ENDIT       LDA    #3
12FF 99 C0 DB              STA    $DBC0,Y
1302 C8                    INY
1303 C0 28                 CPY    #$28
1305 D0 F8                 BNE    ENDIT+2
;
1307 58                    CLI
1308 60                    RTS
;
1309 60 42 40  MSTRING1    .BYTE  96,66,64,96
130D 60 43 41  MSTRING2    .BYTE  96,67,65,96
;
1311 00 00 00  CHARS       .BYTE  0,0,0,0,0,$3,$3,$3
1319 03 03 0B              .BYTE  $3,$3,$B,$B,$2B,$2B
                                  ,$28,$28
1321 00 00 00              .BYTE  0,0,0,0,0,$C0,$C0,$
                                  C0
1329 C0 C0 E0              .BYTE  $C0,$C0,$E0,$E0,$E
                                  8,$E8,$28,$28
;
1331 50 4C 41  TITLE       .ASC   "player1: ships: play
                                  er2:"
;
1352 A0 00     DRWSCREEN   LDY    #0
1354 A2 04     SCRNLUP1    LDX    #4
1356 BD 08 13  SCRNLUP     LDA    MSTRING1-1,X
1359 99 70 07              STA    $770,Y
135C BD 0C 13              LDA    MSTRING2-1,X
135F 99 98 07              STA    $798,Y
1362 C8                    INY
1363 CA                    DEX
1364 D0 F0                 BNE    SCRNLUP
1366 C0 28                 CPY    #$28
1368 D0 EA                 BNE    SCRNLUP1
136A 60                    RTS
;
; ************************************************
; *               configure difficulty           *
; ************************************************
;
136B A9 00     CONFIG      LDA    #0
136D 8D 27 18              STA    DIFF
;
1370 A0 6E                 LDY    #END1-MSG1
1372 A2 1E                 LDX    #<MSG1
1374 A9 14                 LDA    #>MSG1
1376 20 BB 14              JSR    MESSAGE
;
1379 AD 27 18  DIFFLUP     LDA    DIFF
137C 0A                    ASL
137D AA                    TAX
137E BD E7 04              LDA    $4E7,X
1381 49 80                 EOR    #$80
1383 9D E7 04              STA    $4E7,X
1386 20 C8 14              JSR    JOY
1389 F0 1E                 BEQ    DUN1
138B BD E7 04              LDA    $4E7,X
138E 49 80                 EOR    #$80
1390 9D E7 04              STA    $4E7,X
1393 98                    TYA
1394 18                    CLC
1395 6D 27 18              ADC    DIFF
1398 C9 FF                 CMP    #$FF
139A D0 02                 BNE    OK1
139C A9 00                 LDA    #$0
139E C9 03     OK1         CMP    #3
13A0 D0 02                 BNE    OKA
13A2 A9 01                 LDA    #1
13A4 8D 27 18  OKA         STA    DIFF
13A7 D0 D0                 BNE    DIFFLUP
;
13A9 AD 27 18  DUN1        LDA    DIFF
13AC 18                    CLC
13AD 69 04                 ADC    #4
13AF 8D DB 17              STA    PSPEED
13B2 69 02                 ADC    #2
13B4 8D DD 17              STA    CSPEED
13B7 69 03                 ADC    #3
13B9 8D DA 17              STA    CRADIUS
13BC 69 01                 ADC    #1
13BE 8D D9 17              STA    PRADIUS
13C1 A9 02                 LDA    #2
13C3 CD 27 18              CMP    DIFF
13C6 D0 03                 BNE    NOINC
13C8 EE D9 17              INC    PRADIUS
;
13CB A9 FF     NOINC       LDA    #$FF
13CD 8D 28 18              STA    MULTI
13D0 A0 19                 LDY    #END2-MSG2
13D2 A2 8C                 LDX    #<MSG2
13D4 A9 14                 LDA    #>MSG2
13D6 20 BB 14              JSR    MESSAGE
13D9 AD 28 18  NUMLUP      LDA    MULTI
13DC 0A                    ASL
13DD AA                    TAX
13DE E8                    INX
13DF E8                    INX
13E0 BD 2C 05              LDA    $52C,X
13E3 49 80                 EOR    #$80
13E5 9D 2C 05              STA    $52C,X
13E8 20 C8 14              JSR    JOY
13EB F0 23                 BEQ    DUN2
13ED 18                    CLC
13EE 6D 28 18              ADC    MULTI
13F1 BD 2C 05              LDA    $52C,X
13F4 49 80                 EOR    #$80
13F6 9D 2C 05              STA    $52C,X
;
```

```
13F9 98                      TYA
13FA 18                      CLC
13FB 6D 28 18                ADC    MULTI
13FE C9 FE                   CMP    #$FE
1400 D0 02                   BNE    OK2
1402 A9 FF                   LDA    #$FF
1404 C9 01      OK2          CMP    #1
1406 D0 02                   BNE    OKB
1408 A9 00                   LDA    #0
140A 8D 28 18   OKB          STA    MULTI
140D 4C D9 13                JMP    NUMLUP
;
1410 A0 15      DUN2         LDY    #END3—MSG3
1412 A2 A5                   LDX    #<MSG3
1414 A9 14                   LDA    #>MSG3
1416 20 BB 14                JSR    MESSAGE
1419 20 C8 14   ZZZLLUP       JSR    JOY
141C D0 FB                   BNE    ZZZLUP
141E 60                      RTS
;
141F            MSG1         =      *—1
141F C5 20 C9                .ASC   "E I N :trepxe ,etaide
                                    mretni ,ecivon
                                    {DOWN}"
1444 0D                      .BYTE  $D
1445 2D 54 43                .ASC   "-tceles ot nottub dn
                                    a kcitsyoj esu
                                    {DOWN}"
1469 0D                      .BYTE  $D
146A 35 38 39                .ASC   "5891 !etupmoC ,RE
                                    KEES TAEH [<8
                                    >] {DOWN} {CLR}
                                    "
148C 0E                      .BYTE  $E
148D            END1         =      *—1
148D            MSG2         =      *—1
148D 32 20 31                .ASC   "2 1 :sreyalp fo rebm
                                    un {DOWN}"
14A5 0D                      .BYTE  $D
14A6            END2         =      *—1
14A6            MSG3         =      *—1
14A6 4E 49 47                .ASC   "nigeb ot erif sser
                                    p {DOWN}"
14BA 0D                      .BYTE  $D
14BB            END3         =      *—1
14BB 86 FE      MESSAGE      STX    $FE
14BD 85 FF                   STA    $FF
14BF B1 FE      ZMSGLUP      LDA    ($FE),Y
14C1 20 D2 FF                JSR    $FFD2
14C4 88                      DEY
14C5 D0 F8                   BNE    ZMSGLUP
14C7 60                      RTS
;
14C8 AD 00 DC   JOY          LDA    JOY2
14CB 2D 01 DC                AND    JOY1
14CE 29 1C                   AND    #$1C
14D0 C9 1C                   CMP    #$1C
14D2 D0 F4                   BNE    JOY
14D4 A9 00                   LDA    #0
14D6 85 A2                   STA    $A2
14D8 A9 1C                   LDA    #$1C
14DA C5 A2      JOYWAIT      CMP    $A2
14DC D0 FC                   BNE    JOYWAIT
14DE AD 00 DC   J0           LDA    JOY2
14E1 2D 01 DC                AND    JOY1
14E4 29 04                   AND    #4
14E6 D0 03                   BNE    J1
14E8 A0 FF                   LDY    #$FF
14EA 60                      RTS
14EB AD 00 DC   J1           LDA    JOY2
```

```
14EE 2D 01 DC                AND    JOY1
14F1 29 08                   AND    #8
14F3 D0 03                   BNE    J2
14F5 A0 01                   LDY    #1
14F7 60                      RTS
14F8 AD 00 DC   J2           LDA    JOY2
14FB 2D 01 DC                AND    JOY1
14FE 29 10                   AND    #$10
1500 D0 DC                   BNE    J0
1502 A0 00                   LDY    #0
1504 60                      RTS
;
1505            SPRITES      =      *
;
; --- plane1
1505 00 08                   .BYTE  0,8
1507 C0 00 01                .BYTE  192,0,1,27,192,0,1,6
                                    3,192,0,1
1512 3F 80 00                .BYTE  63,128,0,1,62,0,2,60
                                    ,0,2
151C 3C 00 02                .BYTE  60,0,2,60,0,2,62
1523 00 02 3F                .BYTE  0,2,63,0,2,63,0,2
152B 3F 00 02                .BYTE  63,0,2,62,0,2,62,0,2
1534 3C 00 02                .BYTE  60,0,2,60,0,2,56
153B 00 02 30                .BYTE  0,2,48,0,9
; --- plane2
1540 0C 00 02                .BYTE  12,0,2,28
1544 00 02 3C                .BYTE  0,2,60,0,2,60,0,2
154C 7C 00 02                .BYTE  124,0,2,124,0,2,252,
                                    0,2
1555 FC 00 02                .BYTE  252,0,2,252,0,2,124
155C 00 02 3C                .BYTE  0,2,60,0,2,60,0,2
1564 3C 00 02                .BYTE  60,0,2,124,0,1,1,252
                                    ,0,1
156E 03 FC 00                .BYTE  3,252,0,1,3,216,0,1,
                                    3,0,27
; --- plane3
1579 E0 00 02                .BYTE  224,0,2,248,0,2
157F 7E 03 E0                .BYTE  126,3,224,063,255,2
                                    52,127,255
1587 FF 3F FF                .BYTE  255,63,255,255,0,55
158D FF FF FC                .BYTE  255,255,252,255,25
                                    5
1592 FE 3F FF                .BYTE  254,63,255,252,7,19
                                    2,126,0,2
159B 1F 00 02                .BYTE  31,0,2,7,0,22
; --- plane5
15A1 03 00 02                .BYTE  3,0,2,7,0,2
15A7 0F 00 02                .BYTE  15,0,2,15,0,2,63
15AE 00 02 3F                .BYTE  0,2,63,128,0,1,15,22
                                    4,0,1
15B8 03 F8 00                .BYTE  3,248,0,2,255,128,0,
                                    1,63
15C1 E0 00 01                .BYTE  224,0,1,15,224,0,1,3
                                    ,248,0,2
15CC F8 00 02                .BYTE  248,0,2,60,0,2,12
15D3 00 20                   .BYTE  0,32
; --- plane6
15D5 03 FC 00                .BYTE  3,252,0,1,15,252,0,1
                                    ,31,240
15DF 00 01 1F                .BYTE  0,1,31,192,224,127,
                                    0,1,249,252
15E9 00 01 7F                .BYTE  0,1,127,240,0,1,127,
                                    192,0,1,63
15F4 00 02 3C                .BYTE  0,2,60,0,2,16,0,35
; --- plane7
15FC 08 00 02                .BYTE  8,0,2,60,0,2,252
1603 00 01 03                .BYTE  0,1,3,254,0,1,15,254
                                    ,0,1,63
160E 9F 00 01                .BYTE  159,0,1,254,7,3,248,
                                    0,1,15
```

```
1618 F8 00 01                .BYTE  248,0,1,63,240,0,1,6
                                    3,192,0,32
; --- plane8
1623 30 00 02                .BYTE  48,0,2,60,0,2,31,0,2
162C 1F C0 00                .BYTE  31,192,0,1,7,240,0,1
                                    ,7,252
1636 00 01 01                .BYTE  0,1,1,255,0,2,31,192
                                    ,0,1
1640 07 F0 00                .BYTE  7,240,0,1,1,252,0,2,
                                    252
1649 00 02 F0                .BYTE  0,2,240,0,2,240,0,2
1651 E0 00 02                .BYTE  224,0,2,192,0,19
; --- chaser1
1657 28                      .BYTE  40
1658 28 00 01                .BYTE  40,0,1,40,40,0,1,41,
                                    104,0,1
1663 29 68 00                .BYTE  41,104,0,1,9,96,0,1,
                                    9,96
166D 00 01 01                .BYTE  0,1,1,64,0,1,1,64,0,1
                                    ,1
1678 40 00 01                .BYTE  64,0,1,1,64,0,1,1,64,
                                    0,32
; --- chaser2
1683 01 40 00                .BYTE  1,64,0,1,1,64,0,1,1,6
                                    4,0,1
168F 01 40 00                .BYTE  1,64,0,1,1,64,0,1,9,9
                                    6
1699 00 01 09                .BYTE  0,1,9,96,0,1,41,104,
                                    0,1,41
16A4 68 00 01                .BYTE  104,0,1,40,40,0,1,40
                                    ,40,0,35
; --- chaser3
16AF AA 00 02                .BYTE  170,0,2,170,128,0,1
16B6 15 55 00                .BYTE  21,85,0,1,21,85,0,1,
                                    21,85
16C0 00 01 AA                .BYTE  0,1,170,128,0,1,170,
                                    0,46
; --- chaser4
16C9 AA 00 01                .BYTE  170,0,1,2,170,0,1
16D0 55 54 00                .BYTE  85,84,0,1,85,84,0,1,
                                    85,84
16DA 00 01 02                .BYTE  0,1,2,170,0,2,170,0,
                                    44
; --- chaser5
16E3 08 00 02                .BYTE  8,0,2,10,0,2
16E9 06 80 00                .BYTE  6,128,0,1,21,128,0,1
                                    ,165,64
16F3 00 01 29                .BYTE  0,1,41,80,0,1,10,84,
                                    0,2
16FD 15 00 02                .BYTE  21,0,2,5,0,39
; --- chaser6
1703 05 00 02                .BYTE  5,0,2,21,0,1
1709 0A 54 00                .BYTE  10,84,0,1,41,80,0,1,
                                    165,64
1713 00 01 15                .BYTE  0,1,21,128,0,1,6,128
                                    ,0,1,10
171E 00 02 08                .BYTE  0,2,8,0,40
; --- chaser7
1723 20 00 02                .BYTE  32,0,2,160,0,1
1729 02 90 00                .BYTE  2,144,0,1,2,84,0,1,1,
                                    90
1733 00 01 05                .BYTE  0,1,5,104,0,1,21,160
                                    ,0,1,84
173E 00 02 50                .BYTE  0,2,80,0,39
; --- chaser8
1743 50 00 02                .BYTE  80,0,2,84,0,2
1749 15 A0 00                .BYTE  21,160,0,1,5,104,0,1
                                    ,1,90
1753 00 01 02                .BYTE  0,1,2,84,0,1,2,144,0,
                                    2
```

```
175D A0 00 02                .BYTE  160,0,2,32,0,26
; --- fireball
1763 08 80                   .BYTE  8,128
1765 00 01 0A                .BYTE  0,1,10,168,0,1,43,22
                                    4,0,1,11
1770 E0 00 01                .BYTE  224,0,1,11,232,0,1,4
                                    2,160,0,1
177B 02 20 00                .BYTE  2,32,0,51
; --- blast
177F 88 00 01                .BYTE  136,0,1,2
1783 AA 00 01                .BYTE  170,0,1,2,174,0,1,2,
                                    238,128
178D 0A FF A0                .BYTE  10,255,160,10,255,2
                                    24,11,254
1795 A8 0B FF                .BYTE  168,11,255,224,42,2
                                    55,168,43
179D FF E0 2B                .BYTE  255,224,43,255,232,
                                    11,255,224
17A5 2F FF A0                .BYTE  47,255,160,42,255,2
                                    24,47,255
17AD F8 2B BB                .BYTE  248,43,187,224,10,1
                                    70,168,255
;
; --- bit manipulation masks
17B5             SETMASK     =      *
17B5             CLRMASK     =      *+1
17B5 01 FE 02    MASKS       .BYTE  $1,$FE,$2,$FD
17B9 04 FB 08                .BYTE  $4,$FB,$8,$F7
17BD 10 EF 20                .BYTE  $10,$EF,$20,$DF
17C1 40 BF 80                .BYTE  $40,$BF,$80,$7F
;
;************************************************
;*                  variables                   *
;************************************************
;
;object velocities
17C5             XVHI        =      *
17C5             YVHI        =      *+1
17D5                         *=     *+16
17D5             PXVHI       =      XVHI
17D5 00          PXVLO       .BYTE  0
17D6             PYVHI       =      YVHI
17D6 00          PYVLO       .BYTE  0
17D7             CXVHI       =      XVHI+2
17D7 00          CXVLO       .BYTE  0
17D8             CYVHI       =      YVHI+2
17D8 00          CYVLO       .BYTE  0
;
17D9 00          PRADIUS     .BYTE  0
17DA 00          CRADIUS     .BYTE  0
;
;object speeds
17DB             VELOCE      =      *
17EB                         *=     *+16
17EB             PSPEED      =      VELOCE
17EB             CSPEED      =      VELOCE+2
;
;subpixel positions
17EB             XLO         =      *
17EB             YLO         =      *+1
17FB                         *=     *+16
;
;missile timers
17FB             SHOTCLOCK   =      *
1801                         *=     *+6
;
1801 00          FIRE        .BYTE  0
1802 00          LASTSHOT    .BYTE  0
; bits 2-7 ... 1=free
```

```
1803 00     INUSE      .BYTE  0
;
;launch variables
1804        BASCLOCK   =      *
180E                   *=     *+10
180E 00     LNCHTIME   .BYTE  0
;
;multiply variables
180F 00     REGTEMP    .BYTE  0
1810 00     PROD       .BYTE  0
1811 00                .BYTE  0
1812 00     X1         .BYTE  0
1813 00     MSBTEMP    .BYTE  0
1814 00     MULTEMP    .BYTE  0
1815 00     SPRITMP    .BYTE  0
;
;chaser variables
1816 00     DIRTEMP    .BYTE  0
1817 00     VQUAD      .BYTE  0
1818 00     DQUAD      .BYTE  0
1819 00     DELTAX     .BYTE  0
181A 00     DELTAY     .BYTE  0
181B 00     XSCALE     .BYTE  0
181C 00     YSCALE     .BYTE  0
;
;game control
181D 00     MASKTEMP   .BYTE  0
181E 00     SHOTMASK   .BYTE  0
181F 00     OLDJIF     .BYTE  0
1820 00     PLAYNUM    .BYTE  0
1821 00     QUAN       .BYTE  0
1822 00                .BYTE  0
1823 00     ZIPTIME    .BYTE  0
1824 00     ZIPFREQ    .BYTE  0
1825 00     RESP       .BYTE  0
1826 00     JTEMP      .BYTE  0
1827 00     DIFF       .BYTE  0
1828 00     MULTI      .BYTE  0
```

Laser Beam Source Code

Source code by Kevin Mykytyn

```
C000                    *=     $C000
C000          DIR        =     6
C000          FIREFLAG   =     10
C000          FLOP       =     12
C000          TEMP       =     16
C000          ODIR       =     63
C000          FLOPTIMER  =     64
C000          JOYL       =     65
C000          JOYH       =     66
C000          BALLL      =     78
C000          BALLH      =     79
C000          LASERL     =     679
C000          LASERH     =     680
C000          ALREADFIR  =     83
C000          DIR2       =     84
C000          LOW        =     170
C000          HIGH       =     171
C000          D1         =     147
C000          D2         =     148
C000          PIECE      =     149
C000          Q1         =     251
C000          Q2         =     252
C000          Q3         =     253
C000          Q4         =     254
C000          SPRNUM     =     164
C000          TEMP2      =     165
C000          SPRNUM2    =     166
C000          GOODNUM    =     167
C000          COUNT      =     168
C000          FLAG2      =     169
C000          FLAG2LOW   =     682
C000          NUMSPR     =     690
C000          NUMMEN     =     691
C000          IGNORE     =     692
C000          XPOS       =     828
C000          LEVEL      =     693
C000          VOUNTL     =     694
C000          VOUNT      =     695
C000          VOUNTH     =     696
C000          SCOREL     =     697
C000          SCOREH     =     698
C000          YPOS       =     838
C000          XPOSH      =     848
C000          XDIR       =     858
C000          YDIR       =     868
C000          DELTAX     =     878
C000          DELTAY     =     888
C000          XPOSL      =     898
C000          YPOSL      =     908
C000          SPOTTER    =     918
C000          UPLIMIT    =     55
C000          DOWNLIMIT  =     215
C000          LEFTLIMIT  =     55
C000          RIGHTLIMI  =     55
C000          FLOPPER    =     9
C000          NUMBALLS   =     7
C000          LOWBYTE    =     $CF00
C000          HIGHBYTE   =     $CF80
C000          LASERDIR   =     $CE80
C000          BOX        =     $CEA0
```

```
C000               LASER      =     $CE00
C000               NUMLASERS  =     1
C000 A9 04                    LDA   #4
C002 8D B5 02                 STA   LEVEL
C005 20 3A C5      START      JSR   SETUP
C008 20 CC C4                 JSR   INIT
C00B A9 00                    LDA   #0
C00D 8D 15 D0                 STA   53269
C010 A9 93                    LDA   #"{CLR}"
C012 20 D2 FF                 JSR   $FFD2
C015 A9 00                    LDA   #0
C017 8D 21 D0                 STA   53281
C01A 8D 20 D0                 STA   53280
C01D A9 2E                    LDA   #46
C01F 8D A0 CE                 STA   BOX
C022 85 0A                    STA   FIREFLAG
C024 20 CA CC                 JSR   LASERED
C027 A2 09                    LDX   #9
C029 A0 11                    LDY   #17
C02B 18                       CLC
C02C 20 F0 FF                 JSR   $FFF0
C02F A9 DD                    LDA   #<MESS1
C031 A0 CC                    LDY   #>MESS1
C033 20 1E AB                 JSR   $AB1E
C036 A9 6D                    LDA   #109
C038 8D A0 CE                 STA   BOX
C03B 85 0A                    STA   FIREFLAG
C03D 20 CA CC                 JSR   LASERED
C040 A2 0C                    LDX   #12
C042 A0 12                    LDY   #18
C044 18                       CLC
C045 20 F0 FF                 JSR   $FFF0
C048 A9 E9                    LDA   #<MESS2
C04A A0 CC                    LDY   #>MESS2
C04C 20 1E AB                 JSR   $AB1E
C04F A2 0B         WTY2       LDX   #11
C051 A5 A2         X          LDA   162
C053 C5 A2         W          CMP   162
C055 F0 FC                    BEQ   W
C057 CA                       DEX
C058 D0 F7                    BNE   X
C05A AD B5 02                 LDA   LEVEL
C05D 09 30                    ORA   #48
C05F 8D 70 06                 STA   1648
C062 AD 00 DC                 LDA   $DC00
C065 4A                       LSR
C066 B0 0A                    BCS   NTUP
C068 AE B5 02                 LDX   LEVEL
C06B E0 09                    CPX   #9
C06D F0 03                    BEQ   NTUP
C06F EE B5 02                 INC   LEVEL
C072 4A            NTUP       LSR
C073 B0 08                    BCS   NTDOWN
C075 AE B5 02                 LDX   LEVEL
C078 F0 F8                    BEQ   NTUP
C07A CE B5 02                 DEC   LEVEL
C07D 4A            NTDOWN     LSR
C07E 4A                       LSR
C07F 4A                       LSR
C080 B0 CD                    BCS   WTY2
```

```
C082 AD B5 02            LDA   LEVEL
C085 29 0F               AND   #15
C087 8D B5 02            STA   LEVEL
C08A A9 03               LDA   #3
C08C 8D B3 02            STA   NUMMEN
C08F A9 00               LDA   #0
C091 8D B9 02            STA   SCOREL
C094 8D BA 02            STA   SCOREH
C097 A9 0A     NEWL      LDA   #10
C099 38                  SEC
C09A ED B5 02            SBC   LEVEL
C09D 0A                  ASL
C09E 85 4F               STA   BALLH
C0A0 85 4E               STA   BALLL
C0A2 A9 07               LDA   #7
C0A4 8D B2 02            STA   NUMSPR
C0A7 20 E2 CA            JSR   CHAR
C0AA 20 19 C7            JSR   SCREENSET
C0AD 20 CC C4            JSR   INIT
C0B0 20 46 C3            JSR   ADDIT
C0B3 A9 14               LDA   #20
C0B5 8D B7 02            STA   VOUNT
C0B8 20 7B C7  BUCK      JSR   SPRITE
C0BB AC B3 02            LDY   NUMMEN
C0BE C8                  INY
C0BF A9 20               LDA   #32
C0C1 99 DF 07            STA   2015,Y
C0C4 88                  DEY
C0C5 A9 28     VLP       LDA   #40
C0C7 99 DF 07            STA   2015,Y
C0CA A9 07               LDA   #7
C0CC 99 DF DB            STA   2015+54272,Y
C0CF 88                  DEY
C0D0 D0 F3               BNE   VLP
C0D2 20 A9 C5            JSR   SPRMOVE
C0D5 A5 A2               LDA   162
C0D7 C5 A2     WT        CMP   162
C0D9 F0 FC               BEQ   WT
C0DB AD 1E D0            LDA   53278
C0DE AD 1F D0            LDA   53279
C0E1 20 A9 C5  MAIN      JSR   SPRMOVE
C0E4 C6 41               DEC   JOYL
C0E6 D0 15               BNE   NOJOY
C0E8 A5 42               LDA   JOYH
C0EA 85 41               STA   JOYL
C0EC 20 E7 C5            JSR   JOYSTICK
C0EF 20 85 CC            JSR   TIMER
C0F2 20 45 CD            JSR   BONUS
C0F5 A9 00               LDA   #0
C0F7 8D B4 02            STA   IGNORE
C0FA 20 48 C1            JSR   SHOOT
C0FD C6 4E     NOJOY     DEC   BALLL
C0FF D0 07               BNE   NOBALL
C101 A5 4F               LDA   BALLH
C103 85 4E               STA   BALLL
C105 20 C2 C3            JSR   BALL
C108 EA        NOBALL    NOP
C109 20 1F C1  NOLASER   JSR   CHCOLOR
C10C 20 5F CB            JSR   COLLISION
C10F 20 37 CC            JSR   FORBIDDEN
C112 AD 8D 02            LDA   653
C115 F0 CA               BEQ   MAIN
C117 AD 8D 02  VUTU      LDA   653
C11A D0 FB               BNE   VUTU
C11C 4C E1 C0            JMP   MAIN
; change ball color
C11F           CHCOLOR   =     *
C11F A2 02               LDX   #2
C121 AD 1F D0            LDA   53279
C124 4A                  LSR
```

```
C125 4A        VP        LSR
C126 90 1A               BCC   NP
C128 48                  PHA
C129 BD 26 D0            LDA   53286,X
C12C 29 0F               AND   #15
C12E C9 01               CMP   #1
C130 F0 0F               BEQ   NR
C132 A9 01               LDA   #1
C134 9D 26 D0            STA   53286,X
C137 CA                  DEX
C138 E4 A7               CPX   GOODNUM
C13A D0 04               BNE   BACK
C13C A9 18               LDA   #24
C13E 85 A9               STA   FLAG2
C140 E8        BACK      INX
C141 68        NR        PLA
C142 E8        NP        INX
C143 E0 09               CPX   #9
C145 D0 DE               BNE   VP
C147 60                  RTS
;
;shoot the laser
;
C148           SHOOT     =     *
C148 A5 0A               LDA   FIREFLAG
C14A D0 03               BNE   PART2
C14C 4C 16 C2            JMP   AROU
C14F AD B4 02  PART2     LDA   IGNORE
C152 D0 08               BNE   PART
C154 20 EC C6            JSR   CHECK
C157 90 03               BCC   PART
C159 4C 1A C2            JMP   LETGO
C15C A5 53     PART      LDA   ALREADFIRE
C15E D0 2D               BNE   FINISHIT
C160 AE A0 CE            LDX   BOX
C163 BD 00 CF            LDA   LOWBYTE,X
C166 85 AA               STA   LOW
C168 85 FB               STA   Q1
C16A BD 80 CF            LDA   HIGHBYTE,X
C16D 85 AB               STA   HIGH
C16F 85 FC               STA   Q2
C171 BD 00 CE            LDA   LASER,X
C174 85 54               STA   DIR2
C176 4A                  LSR
C177 90 08               BCC   SHORT
C179 A9 40               LDA   #64
C17B 85 95               STA   PIECE
C17D A9 24               LDA   #36
C17F D0 06               BNE   P1
C181 A9 42     SHORT     LDA   #66
C183 85 95               STA   PIECE
C185 A9 16               LDA   #22
C187 85 93     P1        STA   D1
C189 85 94               STA   D2
C18B 85 53               STA   ALREADFIRE
C18D A5 54     FINISHIT  LDA   DIR2
C18F C9 23               CMP   #35
C191 B0 17               BCS   SUBTRACT
C193 C9 21               CMP   #33
C195 D0 04               BNE   TUTTLE
C197 A9 01               LDA   #1
C199 D0 02               BNE   P2
C19B A9 28     TUTTLE    LDA   #40
C19D 18        P2        CLC
C19E 65 AA               ADC   LOW
C1A0 85 AA               STA   LOW
C1A2 A5 AB               LDA   HIGH
C1A4 69 00               ADC   #0
C1A6 85 AB               STA   HIGH
C1A8 D0 19               BNE   DUNT
```

```
C1AA C9 24    SUBTRACT  CMP  #36
C1AC F0 04              BEQ  BIG
C1AE A9 01              LDA  #1
C1B0 D0 02              BNE  P3
C1B2 A9 28    BIG       LDA  #40
C1B4 85 10    P3        STA  TEMP
C1B6 A5 AA              LDA  LOW
C1B8 38                 SEC
C1B9 E5 10              SBC  TEMP
C1BB 85 AA              STA  LOW
C1BD A5 AB              LDA  HIGH
C1BF E9 00              SBC  #0
C1C1 85 AB              STA  HIGH
C1C3 A5 95    DUNT      LDA  PIECE
C1C5 A0 00              LDY  #0
C1C7 91 AA              STA  (LOW),Y
C1C9 A5 AA              LDA  LOW
C1CB 18                 CLC
C1CC 69 00              ADC  #<54272
C1CE 85 FD              STA  Q3
C1D0 A5 AB              LDA  HIGH
C1D2 69 D4              ADC  #>54272
C1D4 85 FE              STA  Q4
C1D6 A9 0A              LDA  #10
C1D8 A0 00              LDY  #0
C1DA 91 FD              STA  (Q3),Y
C1DC A9 80              LDA  #128
C1DE 8D 0B D4           STA  $D40B
C1E1 A9 08              LDA  #8
C1E3 8D 0C D4           STA  $D40C
C1E6 A5 93              LDA  D1
C1E8 18                 CLC
C1E9 69 0A              ADC  #10
C1EB 0A                 ASL
C1EC 8D 08 D4           STA  $D408
C1EF A9 81              LDA  #129
C1F1 8D 0B D4           STA  $D40B
C1F4 C6 93              DEC  D1
C1F6 D0 1E              BNE  AROU
C1F8 A5 95              LDA  PIECE
C1FA C9 20              CMP  #32
C1FC F0 12              BEQ  FINIS
C1FE A9 20              LDA  #32
C200 85 95              STA  PIECE
C202 A5 FB              LDA  Q1
C204 85 AA              STA  LOW
C206 A5 FC              LDA  Q2
C208 85 AB              STA  HIGH
C20A A5 94              LDA  D2
C20C 85 93              STA  D1
C20E D0 06              BNE  AROU
C210 A9 00    FINIS     LDA  #0
C212 85 0A              STA  FIREFLAG
C214 85 53              STA  ALREADFIRE
C216 60       AROU      RTS
C217 4C D4 C2 JRITZ     JMP  RITZ
C21A A9 00    LETGO     LDA  #0
C21C 85 0A              STA  FIREFLAG
C21E A6 A6              LDX  SPRNUM2
C220 BD 96 03           LDA  SPOTTER,X
C223 C9 02              CMP  #2
C225 90 F0              BCC  JRITZ
C227 AD 50 03           LDA  XPOSH
C22A F0 EB              BEQ  JRITZ
C22C AD 3C 03           LDA  XPOS
C22F C9 37              CMP  #RIGHTLIMIT
C231 D0 E4              BNE  JRITZ
C233 AD 46 03           LDA  YPOS
C236 C9 D6              CMP  #DOWNLIMIT-1
C238 90 DD              BCC  JRITZ
```

```
C23A 8A                 TXA
C23B 48                 PHA
C23C 20 17 C3           JSR  MAKESAFE
C23F 68                 PLA
C240 AA                 TAX
C241 A9 1A              LDA  #26
C243 85 A9              STA  FLAG2
C245 86 10              STX  TEMP
C247 A2 00              LDX  #0
C249 A0 06              LDY  #6
C24B B9 28 D0 TMP       LDA  53288,Y
C24E 29 0F              AND  #15
C250 C9 01              CMP  #1
C252 D0 01              BNE  NPI
C254 E8                 INX
C255 88       NPI       DEY
C256 10 F3              BPL  TMP
C258 BD 32 C5           LDA  SCORTAB,X
C25B 18                 CLC
C25C 6D B9 02           ADC  SCOREL
C25F 8D B9 02           STA  SCOREL
C262 AD BA 02           LDA  SCOREH
C265 69 00              ADC  #0
C267 8D BA 02           STA  SCOREH
C26A 20 78 CD           JSR  SCOROUT
C26D A6 10              LDX  TEMP
C26F A9 FF              LDA  #255
C271 9D 46 03           STA  YPOS,X
C274 CE B2 02           DEC  NUMSPR
C277 D0 50              BNE  NOTOVER
C279 A9 00              LDA  #0
C27B 8D 15 D0           STA  53269
C27E A9 93              LDA  #"{CLR}"
C280 20 D2 FF           JSR  $FFD2
C283 A2 0A              LDX  #10
C285 A0 0F              LDY  #15
C287 18                 CLC
C288 20 F0 FF           JSR  $FFF0
C28B 20 D5 C2           JSR  LEVLSOUND
C28E A9 20              LDA  #<MESS4
C290 A0 CD              LDY  #>MESS4
C292 20 1E AB           JSR  $AB1E
C295 AD B7 02           LDA  VOUNT
C298 0A                 ASL
C299 85 10              STA  TEMP
C29B 0A                 ASL
C29C 0A                 ASL
C29D 18                 CLC
C29E 65 10              ADC  TEMP
C2A0 AA                 TAX
C2A1 48                 PHA
C2A2 A9 00              LDA  #0
C2A4 20 CD BD           JSR  $BDCD
C2A7 68                 PLA
C2A8 18                 CLC
C2A9 6D B9 02           ADC  SCOREL
C2AC 8D B9 02           STA  SCOREL
C2AF AD BA 02           LDA  SCOREH
C2B2 69 00              ADC  #0
C2B4 8D BA 02           STA  SCOREH
C2B7 A9 00              LDA  #0
C2B9 85 A2              STA  162
C2BB A5 A2    WISH      LDA  162
C2BD C9 78              CMP  #120
C2BF D0 FA              BNE  WISH
C2C1 68                 PLA
C2C2 68                 PLA
C2C3 EE B5 02           INC  LEVEL
C2C6 4C 97 C0           JMP  NEWL
C2C9 20 23 C3 NOTOVER   JSR  DNSOUND
```

```
C2CC A9 01                   LDA   #1
C2CE 9D 96 03                STA   SPOTTER,X
C2D1 20 FF C2                JSR   PICK
C2D4 60         RITZ         RTS
;
; next level sound
;
C2D5            LEVLSOUND    =     *
C2D5 A9 14                   LDA   #20
C2D7 8D 04 D4                STA   $D404
C2DA 8D 0B D4                STA   $D40B
C2DD A9 0A                   LDA   #10
C2DF 8D 05 D4                STA   $D405
C2E2 8D 0C D4                STA   $D40C
C2E5 AD B5 02                LDA   LEVEL
C2E8 0A                      ASL
C2E9 18                      CLC
C2EA 69 05                   ADC   #5
C2EC 8D 01 D4                STA   $D401
C2EF A9 23                   LDA   #35
C2F1 8D 08 D4                STA   $D408
C2F4 A9 11                   LDA   #17
C2F6 8D 04 D4                STA   $D404
C2F9 A9 15                   LDA   #21
C2FB 8D 0B D4                STA   $D40B
C2FE 60                      RTS
;
C2FF E6 A8      PICK         INC   COUNT
C301 A6 A8                   LDX   COUNT
C303 BD F9 C7                LDA   COLTAB,X
C306 86 A7                   STX   GOODNUM
C308 8D 20 D0                STA   53280
C30B A0 07                   LDY   #7
C30D B9 F9 C7   MMM          LDA   COLTAB,Y
C310 99 27 D0                STA   53287,Y
C313 88                      DEY
C314 10 F7                   BPL   MMM
C316 60                      RTS
C317 A2 18      MAKESAFE     LDX   #24
C319 A0 05      UP           LDY   #5
C31B 8A                      TXA
C31C 20 A5 CC                JSR   FUT
C31F CA                      DEX
C320 D0 F7                   BNE   UP
C322 60                      RTS
;
; got one ball in -- sound
;
C323            DNSOUND      =     *
C323 A9 14                   LDA   #20
C325 8D 0B D4                STA   $D40B
C328 AD B2 02                LDA   NUMSPR
C32B 18                      CLC
C32C 69 04                   ADC   #4
C32E 8D 01 D4                STA   $D401
C331 AD B3 02                LDA   NUMMEN
C334 0A                      ASL
C335 0A                      ASL
C336 0A                      ASL
C337 0A                      ASL
C338 8D 08 D4                STA   $D408
C33B A9 0C                   LDA   #12
C33D 8D 0C D4                STA   $D40C
C340 A9 15                   LDA   #21
C342 8D 0B D4                STA   $D40B
C345 60                      RTS
;
; move the lasers
;
C346            LASERMOVE    =     *
```

```
C346 EE A0 CE   ADDIT        INC   BOX
C349 AD A0 CE                LDA   BOX
C34C C9 7A                   CMP   #122
C34E D0 11                   BNE   OK7
C350 A9 00                   LDA   #0
C352 8D A0 CE                STA   BOX
C355 F0 0A                   BEQ   OK7
C357 CE A0 CE   SUBIT3       DEC   BOX
C35A 10 05                   BPL   OK7
C35C A9 79                   LDA   #121
C35E 8D A0 CE                STA   BOX
C361 AD A0 CE   OK7          LDA   BOX
C364 A8                      TAY
C365 20 6F C3                JSR   NOSIT
C368 B9 00 CE                LDA   LASER,Y
C36B 20 9C C3                JSR   PUTIT
C36E 60                      RTS
;
; laser moving sound
;
C36F            NOSIT        =     *
C36F AD 9B C3                LDA   PULSE
C372 C9 19                   CMP   #25
C374 B0 05                   BCS   SNDLAS
C376 EE 9B C3                INC   PULSE
C379 D0 05                   BNE   LAS2
C37B A9 05      SNDLAS       LDA   #5
C37D 8D 9B C3                STA   PULSE
C380 A9 14      LAS2         LDA   #20
C382 8D 0B D4                STA   $D40B
C385 A9 07                   LDA   #7
C387 8D 01 D4                STA   $D401
C38A AD 9B C3                LDA   PULSE
C38D 8D 08 D4                STA   $D408
C390 A9 26                   LDA   #32+6
C392 8D 0C D4                STA   $D40C
C395 A9 15                   LDA   #21
C397 8D 0B D4                STA   $D40B
C39A 60                      RTS
C39B 05         PULSE        .BYT  5
;
C39C 48         PUTIT        PHA
C39D AD A0 CE                LDA   BOX
C3A0 A8                      TAY
C3A1 B9 00 CF                LDA   LOWBYTE,Y
C3A4 85 04                   STA   4
C3A6 B9 80 CF                LDA   HIGHBYTE,Y
C3A9 85 05                   STA   5
C3AB 68                      PLA
C3AC A0 00                   LDY   #0
C3AE 91 04                   STA   (4),Y
C3B0 A5 04                   LDA   4
C3B2 18                      CLC
C3B3 69 00                   ADC   #<54272
C3B5 85 04                   STA   4
C3B7 A5 05                   LDA   5
C3B9 69 D4                   ADC   #>54272
C3BB 85 05                   STA   5
C3BD A9 07                   LDA   #7
C3BF 91 04                   STA   (4),Y
C3C1 60                      RTS
;
; ballmoving routine
;
C3C2            BALL         =     *
C3C2 A2 07                   LDX   #NUMBALLS
C3C4 BD 96 03   BALLOOP1     LDA   SPOTTER,X
C3C7 F0 03                   BEQ   BOU
C3C9 4C 83 C4                JMP   DN
C3CC BD 46 03   BOU          LDA   YPOS,X
```

```
C3CF C9 37              CMP   #UPLIMIT
C3D1 D0 05              BNE   NOB1
C3D3 FE 46 03           INC   YPOS,X
C3D6 D0 07              BNE   BOUNCE
C3D8 C9 D7     NOB1     CMP   #DOWNLIMIT
C3DA D0 11              BNE   NOBOUNCE
C3DC DE 46 03           DEC   YPOS,X
C3DF BD 64 03  BOUNCE   LDA   YDIR,X
C3E2 49 01              EOR   #1
C3E4 9D 64 03           STA   YDIR,X
C3E7 20 8A C4           JSR   RANDOMIZE
C3EA 20 B7 C4           JSR   BONK
C3ED BD 3C 03  NOBOUNCE LDA   XPOS,X
C3F0 C9 37              CMP   #LEFTLIMIT
C3F2 D0 0A              BNE   NE1
C3F4 BD 50 03           LDA   XPOSH,X
C3F7 D0 05              BNE   NE1
C3F9 FE 3C 03           INC   XPOS,X
C3FC D0 0F              BNE   BOUNCE2
C3FE BD 3C 03  NE1      LDA   XPOS,X
C401 C9 37              CMP   #RIGHTLIMIT
C403 D0 16              BNE   NOBO2
C405 BD 50 03           LDA   XPOSH,X
C408 F0 11              BEQ   NOBO2
C40A DE 3C 03           DEC   XPOS,X
C40D BD 5A 03  BOUNCE2  LDA   XDIR,X
C410 49 01              EOR   #1
C412 9D 5A 03           STA   XDIR,X
C415 20 8A C4           JSR   RANDOMIZE
C418 20 B7 C4           JSR   BONK
C41B BD 5A 03  NOBO2    LDA   XDIR,X
C41E D0 1D              BNE   SUBIT
C420 BD 82 03           LDA   XPOSL,X
C423 18                 CLC
C424 7D 6E 03           ADC   DELTAX,X
C427 9D 82 03           STA   XPOSL,X
C42A BD 3C 03           LDA   XPOS,X
C42D 69 00              ADC   #0
C42F 9D 3C 03           STA   XPOS,X
C432 BD 50 03           LDA   XPOSH,X
C435 69 00              ADC   #0
C437 9D 50 03           STA   XPOSH,X
C43A 4C 57 C4           JMP   DOY
C43D BD 82 03  SUBIT    LDA   XPOSL,X
C440 38                 SEC
C441 FD 6E 03           SBC   DELTAX,X
C444 9D 82 03           STA   XPOSL,X
C447 BD 3C 03           LDA   XPOS,X
C44A E9 00              SBC   #0
C44C 9D 3C 03           STA   XPOS,X
C44F BD 50 03           LDA   XPOSH,X
C452 E9 00              SBC   #0
C454 9D 50 03           STA   XPOSH,X
C457 BD 64 03  DOY      LDA   YDIR,X
C45A D0 15              BNE   SUBIT2
C45C BD 8C 03           LDA   YPOSL,X
C45F 18                 CLC
C460 7D 78 03           ADC   DELTAY,X
C463 9D 8C 03           STA   YPOSL,X
C466 BD 46 03           LDA   YPOS,X
C469 69 00              ADC   #0
C46B 9D 46 03           STA   YPOS,X
C46E 4C 83 C4           JMP   DN
C471 BD 8C 03  SUBIT2   LDA   YPOSL,X
C474 38                 SEC
C475 FD 78 03           SBC   DELTAY,X
C478 9D 8C 03           STA   YPOSL,X
C47B BD 46 03           LDA   YPOS,X
C47E E9 00              SBC   #0
C480 9D 46 03           STA   YPOS,X
```

```
C483 CA        DN        DEX
C484 F0 03               BEQ   RET1
C486 4C C4 C3            JMP   BALLOOP1
C489 60        RET1      RTS
C48A 20 A7 C4  RANDOMIZE JSR   RANDOM
C48D 18                  CLC
C48E 7D 6E 03            ADC   DELTAX,X
C491 C9 64               CMP   #100
C493 90 03               BCC   DONT
C495 9D 6E 03            STA   DELTAX,X
C498 20 A7 C4  DONT      JSR   RANDOM
C49B 18                  CLC
C49C 7D 78 03            ADC   DELTAY,X
C49F C9 96               CMP   #150
C4A1 90 03               BCC   DONT2
C4A3 9D 78 03            STA   DELTAY,X
C4A6 60        DONT2     RTS
C4A7 AD 1B D4  RANDOM    LDA   $D41B
C4AA 29 1F               AND   #31
C4AC AC 1B D4            LDY   $D41B
C4AF 10 05               BPL   OK
C4B1 49 FF               EOR   #255
C4B3 18                  CLC
C4B4 69 01               ADC   #1
C4B6 60        OK        RTS
;
; ball bounce sound routine
;
C4B7           BONK      =     *
C4B7 A9 80               LDA   #128
C4B9 8D 04 D4            STA   $D404
C4BC A9 40               LDA   #64
C4BE 8D 05 D4            STA   $D405
C4C1 A9 14               LDA   #20
C4C3 8D 01 D4            STA   $D401
C4C6 A9 81               LDA   #129
C4C8 8D 04 D4            STA   $D404
C4CB 60                  RTS
;
; initializations
;
C4CC           INIT      =     *
C4CC 20 54 CC            JSR   SETSID
C4CF A9 00               LDA   #0
C4D1 85 0C               STA   FLOP
C4D3 85 0A               STA   FIREFLAG
C4D5 85 53               STA   ALREADFIRE
C4D7 A9 09               LDA   #FLOPPER
C4D9 85 40               STA   FLOPTIMER
C4DB 8D AA 02            STA   FLAG2LOW
C4DE A9 80               LDA   #128
C4E0 8D 0F D4            STA   54287
C4E3 8D 12 D4            STA   54290
C4E6 A9 08               LDA   #8
C4E8 85 41               STA   JOYL
C4EA 85 42               STA   JOYH
C4EC A9 03               LDA   #3
C4EE 8D B8 02            STA   VOUNTH
C4F1 A9 07               LDA   #7
C4F3 8D A7 02            STA   LASERL
C4F6 8D A8 02            STA   LASERH
C4F9 A0 07               LDY   #NUMBALLS
C4FB 20 2A C5  LOOP1     JSR   RAND
C4FE 99 6E 03            STA   DELTAX,Y
C501 20 2A C5            JSR   RAND
C504 99 78 03            STA   DELTAY,Y
C507 AD 1B D4            LDA   $D41B
C50A 29 01               AND   #1
C50C 99 5A 03            STA   XDIR,Y
C50F AD 1B D4            LDA   $D41B
```

```
C512 29 01                 AND   #1
C514 99 64 03              STA   YDIR,Y
C517 88                    DEY
C518 D0 E1                 BNE   LOOP1
C51A A9 09                 LDA   #9
C51C 85 A4                 STA   SPRNUM
C51E A9 00                 LDA   #0
C520 85 A8                 STA   COUNT
C522 20 FF C2              JSR   PICK
C525 A9 01                 LDA   #1
C527 85 3F                 STA   ODIR
C529 60                    RTS
C52A AD 1B D4  RAND        LDA   $D41B
C52D C9 96                 CMP   #150
C52F 90 F9                 BCC   RAND
C531 60                    RTS
C532 00 05 0A  SCORTAB     .BYT  0,5,10,20,40,80,120,
                                 160
;
; set up laser movement
;
C53A           SETUP       =     *
C53A A2 FF                 LDX   #255
C53C A9 03                 LDA   #<1027
C53E 85 02                 STA   2
C540 A9 04                 LDA   #>1027
C542 85 03                 STA   3
C544 A9 22     VOOP        LDA   #34
C546 20 9A C5              JSR   PUT
C549 E6 02                 INC   2
C54B A5 02                 LDA   2
C54D C9 28                 CMP   #40
C54F D0 F3                 BNE   VOOP
C551 C6 02                 DEC   2
C553 A9 23     VOOP2       LDA   #35
C555 20 9A C5              JSR   PUT
C558 A5 02                 LDA   2
C55A 18                    CLC
C55B 69 28                 ADC   #40
C55D 85 02                 STA   2
C55F A5 03                 LDA   3
C561 69 00                 ADC   #0
C563 85 03                 STA   3
C565 A5 02                 LDA   2
C567 C9 E7                 CMP   #231
C569 D0 E8                 BNE   VOOP2
C56B A9 BF                 LDA   #191
C56D 85 02                 STA   2
C56F A9 24     VOOP3       LDA   #36
C571 20 9A C5              JSR   PUT
C574 C6 02                 DEC   2
C576 A5 02                 LDA   2
C578 C9 9A                 CMP   #154
C57A D0 F3                 BNE   VOOP3
C57C E6 02                 INC   2
C57E A9 21     VOOP4       LDA   #33
C580 20 9A C5              JSR   PUT
C583 A5 02                 LDA   2
C585 38                    SEC
C586 E9 28                 SBC   #40
C588 85 02                 STA   2
C58A A5 03                 LDA   3
C58C E9 00                 SBC   #0
C58E 85 03                 STA   3
C590 C9 03                 CMP   #3
C592 D0 EA                 BNE   VOOP4
C594 A9 14                 LDA   #20
C596 8D A0 CE              STA   BOX
C599 60                    RTS
C59A E8        PUT         INX
```

```
C59B 9D 00 CE              STA   LASER,X
C59E A5 02                 LDA   2
C5A0 9D 00 CF              STA   LOWBYTE,X
C5A3 A5 03                 LDA   3
C5A5 9D 80 CF              STA   HIGHBYTE,X
C5A8 60                    RTS
;
; sprite movement routine
;
C5A9           SPRMOVE     =     *
C5A9 A9 00                 LDA   #0
C5AB 85 10                 STA   TEMP
C5AD A2 07                 LDX   #7
C5AF A0 0E                 LDY   #14
C5B1 BD 96 03  SPRLOOP1    LDA   SPOTTER,X
C5B4 C9 02                 CMP   #2
C5B6 D0 12                 BNE   NET
C5B8 AD 3C 03              LDA   XPOS
C5BB 9D 3C 03              STA   XPOS,X
C5BE AD 46 03              LDA   YPOS
C5C1 9D 46 03              STA   YPOS,X
C5C4 AD 50 03              LDA   XPOSH
C5C7 9D 50 03              STA   XPOSH,X
C5CA BD 50 03  NET         LDA   XPOSH,X
C5CD 4A                    LSR
C5CE 26 10                 ROL   TEMP
C5D0 BD 3C 03              LDA   XPOS,X
C5D3 99 00 D0              STA   53248,Y
C5D6 BD 46 03              LDA   YPOS,X
C5D9 99 01 D0              STA   53249,Y
C5DC 88                    DEY
C5DD 88                    DEY
C5DE CA                    DEX
C5DF 10 D0                 BPL   SPRLOOP1
C5E1 A5 10                 LDA   TEMP
C5E3 8D 10 D0              STA   53264
C5E6 60                    RTS
;
; joystick movement
;
C5E7           JOYSTICK    =     *
C5E7 A9 00                 LDA   #0
C5E9 85 06                 STA   DIR
C5EB A0 00                 LDY   #0
C5ED A5 53                 LDA   ALREADFIRE
C5EF F0 01                 BEQ   DUIT
C5F1 60                    RTS
C5F2 AD 00 DC  DUIT        LDA   $DC00
C5F5 4A                    LSR
C5F6 B0 0F                 BCS   NOTUP
C5F8 AE 46 03              LDX   YPOS
C5FB E0 37                 CPX   #UPLIMIT
C5FD F0 08                 BEQ   NOTUP
C5FF CE 46 03              DEC   YPOS
C602 A2 01                 LDX   #1
C604 86 06                 STX   DIR
C606 C8                    INY
C607 4A        NOTUP       LSR
C608 B0 0F                 BCS   NOTDOWN
C60A AE 46 03              LDX   YPOS
C60D E0 D7                 CPX   #DOWNLIMIT
C60F F0 08                 BEQ   NOTDOWN
C611 EE 46 03              INC   YPOS
C614 A2 02                 LDX   #2
C616 86 06                 STX   DIR
C618 C8                    INY
C619 4A        NOTDOWN     LSR
C61A B0 3B                 BCS   NOTLEFT
C61C 20 EC C6              JSR   CHECK
C61F B0 0F                 BCS   NOP1
```

```
C621 D0 34               BNE  NOTLEFT
C623 48                  PHA
C624 A9 20               LDA  #32
C626 20 9C C3            JSR  PUTIT
C629 20 57 C3            JSR  SUBIT3
C62C 68                  PLA
C62D 4C 57 C6            JMP  NOTLEFT
C630 AE 50 03  NOP1      LDX  XPOSH
C633 D0 07               BNE  NOCHECK
C635 AE 3C 03            LDX  XPOS
C638 E0 19               CPX  #LEFTLIMIT-30
C63A F0 1B               BEQ  NOTLEFT
C63C 48        NOCHECK   PHA
C63D AD 3C 03            LDA  XPOS
C640 38                  SEC
C641 E9 01               SBC  #1
C643 8D 3C 03            STA  XPOS
C646 AD 50 03            LDA  XPOSH
C649 E9 00               SBC  #0
C64B 8D 50 03            STA  XPOSH
C64E C0 00               CPY  #0
C650 D0 04               BNE  NOPE
C652 A9 03               LDA  #3
C654 85 06               STA  DIR
C656 68        NOPE      PLA
C657 4A        NOTLEFT   LSR
C658 B0 3B               BCS  NOTRIGHT
C65A 20 EC C6            JSR  CHECK
C65D B0 0F               BCS  NOP2
C65F D0 34               BNE  NOTRIGHT
C661 48                  PHA
C662 A9 20               LDA  #32
C664 20 9C C3            JSR  PUTIT
C667 20 46 C3            JSR  ADDIT
C66A 68                  PLA
C66B 4C 95 C6            JMP  NOTRIGHT
C66E AE 50 03  NOP2      LDX  XPOSH
C671 F0 07               BEQ  NOCHECK2
C673 AE 3C 03            LDX  XPOS
C676 E0 37               CPX  #RIGHTLIMIT
C678 F0 1B               BEQ  NOTRIGHT
C67A 48        NOCHECK2  PHA
C67B AD 3C 03            LDA  XPOS
C67E 18                  CLC
C67F 69 01               ADC  #1
C681 8D 3C 03            STA  XPOS
C684 AD 50 03            LDA  XPOSH
C687 69 00               ADC  #0
C689 8D 50 03            STA  XPOSH
C68C C0 00               CPY  #0
C68E D0 04               BNE  NOPE2
C690 A9 04               LDA  #4
C692 85 06               STA  DIR
C694 68        NOPE2     PLA
C695 4A        NOTRIGHT  LSR
C696 B0 04               BCS  DONE1
C698 A9 01               LDA  #1
C69A 85 0A               STA  FIREFLAG
C69C A5 06     DONE1     LDA  DIR
C69E D0 0A               BNE  NOTSTANDING
C6A0 A6 3F               LDX  ODIR
C6A2 BD 10 C7            LDA  STANDTAB-1,X
C6A5 8D F8 07            STA  2040
C6A8 D0 3F               BNE  DONEDIR
C6AA C6 40     NOTSTANDI DEC  FLOPTIMER
C6AC D0 30               BNE  NOFLOP
C6AE A9 09               LDA  #FLOPPER
C6B0 85 40               STA  FLOPTIMER
C6B2 A5 0C               LDA  FLOP
C6B4 49 01               EOR  #1
```

```
C6B6 85 0C               STA  FLOP
C6B8 A9 40               LDA  #64
C6BA 8D 04 D4            STA  $D404
C6BD A5 06               LDA  DIR
C6BF CD 83 CC            CMP  DIRSV
C6C2 F0 08               BEQ  OLDIR
C6C4 8D 83 CC  NUDIR     STA  DIRSV
C6C7 A9 00               LDA  #0
C6C9 8D 84 CC            STA  STEPCOUNT
C6CC EE 84 CC  OLDIR     INC  STEPCOUNT
C6CF AD 84 CC            LDA  STEPCOUNT
C6D2 18                  CLC
C6D3 69 05               ADC  #5
C6D5 0A                  ASL
C6D6 8D 01 D4            STA  $D401
C6D9 A9 41     STEPS     LDA  #65
C6DB 8D 04 D4            STA  $D404
C6DE A5 0C     NOFLOP    LDA  FLOP
C6E0 A6 06               LDX  DIR
C6E2 18                  CLC
C6E3 7D 14 C7            ADC  OFFTAB-1,X
C6E6 8D F8 07            STA  2040
C6E9 86 3F     DONEDIR   STX  ODIR
C6EB 60                  RTS
C6EC AE 3C 03  CHECK     LDX  XPOS
C6EF E0 19               CPX  #LEFTLIMIT-30
C6F1 D0 1C               BNE  NE6
C6F3 AE 46 03            LDX  YPOS
C6F6 E0 37               CPX  #UPLIMIT
C6F8 D0 15               BNE  NE6
C6FA AE 50 03            LDX  XPOSH
C6FD D0 10               BNE  NE6
C6FF CE A7 02            DEC  LASERL
C702 D0 08               BNE  DUT
C704 AE A8 02            LDX  LASERH
C707 8E A7 02            STX  LASERL
C70A A2 00               LDX  #0
C70C 18        DUT       CLC
C70D 90 01               BCC  RET9
C70F 38        NE6       SEC
C710 60        RET9      RTS
C711 CC CF D3  STANDTAB  .BYT 204,207,211,213
C715 CD D0 D2  OFFTAB    .BYT 205,208,210,212
;
; set up the screen
;
C719           SCREENSET =    *
C719 A9 93               LDA  #"{CLR}"
C71B 20 D2 FF            JSR  $FFD2
C71E A9 00               LDA  #0
C720 8D 21 D0            STA  53281
C723 A9 0B               LDA  #11
C725 8D 20 D0            STA  53280
C728 A9 18               LDA  #24
C72A 85 10               STA  TEMP
C72C A9 69     FLOOP     LDA  #<LINE9
C72E A0 C7               LDY  #>LINE9
C730 20 1E AB            JSR  $AB1E
C733 C6 10               DEC  TEMP
C735 D0 F5               BNE  FLOOP
C737 A0 25               LDY  #37
C739 98        V7        TYA
C73A 99 BF 07            STA  1983,Y
C73D A9 01               LDA  #1
C73F 99 BF DB            STA  56255,Y
C742 C8                  INY
C743 C0 28               CPY  #40
C745 D0 F2               BNE  V7
C747 A2 13               LDX  #19
C749 A9 A0     CRP2      LDA  #160
```

```
C74B 9D C0 07              STA    1984,X
C74E A9 0A                 LDA    #10
C750 9D C0 DB              STA    56256,X
C753 CA                    DEX
C754 10 F3                 BPL    CRP2
C756 A2 18                 LDX    #24
C758 A0 14                 LDY    #20
C75A 18                    CLC
C75B 20 F0 FF              JSR    $FFF0
C75E A9 70                 LDA    #<SCRE
C760 A0 C7                 LDY    #>SCRE
C762 20 1E AB              JSR    $AB1E
C765 20 78 CD              JSR    SCOROUT
C768 60        DUEN        RTS
C769 1E 12 20  LINE9       .ASC   "{GRN}{RVS} "
C76E 0D 00                 .BYT   13,0
C770 9F 12 53  SCRE        .ASC   "{CYN}{RVS}SCO
                                  RE{OFF} 0"
C77A 00                    .BYT   0
;
; set up sprites
;
C77B           SPRITE      =      *
C77B A0 00                 LDY    #0
C77D B9 22 C8  SPLOOP1     LDA    SPRDAT,Y
C780 99 00 33              STA    13056,Y
C783 B9 22 C9              LDA    SPRDAT+256,Y
C786 99 00 34              STA    13056+256,Y
C789 B9 22 CA              LDA    SPRDAT+512,Y
C78C 99 00 35              STA    13056+512,Y
C78F 88                    DEY
C790 D0 EB                 BNE    SPLOOP1
C792 A9 FF                 LDA    #255
C794 8D 15 D0              STA    53269
C797 A9 07                 LDA    #7
C799 A8                    TAY
C79A 38                    SEC
C79B ED B2 02              SBC    NUMSPR
C79E 85 10                 STA    TEMP
C7A0 B9 F9 C7  SPLOOP2     LDA    COLTAB,Y
C7A3 99 27 D0              STA    53287,Y
C7A6 B9 01 C8              LDA    PTRTAB,Y
C7A9 99 F8 07              STA    2040,Y
C7AC AD 01 C8              LDA    PTRTAB
C7AF 8D F8 07              STA    2040
C7B2 B9 09 C8              LDA    XPTAB,Y
C7B5 99 3C 03              STA    XPOS,Y
C7B8 AD 09 C8              LDA    XPTAB
C7BB 8D 3C 03              STA    XPOS
C7BE B9 11 C8              LDA    YPTAB,Y
C7C1 99 46 03              STA    YPOS,Y
C7C4 AD 11 C8              LDA    YPTAB
C7C7 8D 46 03              STA    YPOS
C7CA B9 19 C8              LDA    XPHTAB,Y
C7CD 99 50 03              STA    XPOSH,Y
C7D0 A9 00                 LDA    #0
C7D2 8D 50 03              STA    XPOSH
C7D5 A9 00                 LDA    #0
C7D7 99 96 03              STA    SPOTTER,Y
C7DA 8D 96 03              STA    SPOTTER
C7DD 88                    DEY
C7DE C4 10                 CPY    TEMP
C7E0 D0 BE                 BNE    SPLOOP2
C7E2 A9 FF                 LDA    #255
C7E4 8D 1C D0              STA    53276
C7E7 A9 07                 LDA    #7
C7E9 8D 25 D0              STA    53285
C7EC A9 06                 LDA    #6
C7EE 8D 26 D0              STA    53286
C7F1 A9 1A                 LDA    #26
```

```
C7F3 85 A9                 STA    FLAG2
C7F5 20 17 C3              JSR    MAKESAFE
C7F8 60                    RTS
C7F9 02 02 03  COLTAB      .BYT   2,2,3,4,13,6,7,11
C801 CC D6 D6  PTRTAB      .BYT   204,214,214,214,21
                                  4,214,214,214
C809 19 64 64  XPTAB       .BYT   25,100,100,100,100,
                                  100,100,100
C811 64 64 64  YPTAB       .BYT   100,100,100,100,10
                                  0,100,100,100
C819 00 00 00  XPHTAB      .BYT   0,0,0,0,0,0,0,0
C821 60                    RTS
C822 00 00 00  SPRDAT      .BYT   0,0,0,0,0,0,0,60
C82A 00 00 FF              .BYT   0,0,255,0,0,255,0,0
C832 FF 00 00              .BYT   255,0,0,60,0,2,255,1
                                  28
C83A 02 FF 80              .BYT   002,255,128,2,255,1
                                  28,0,255
C842 80 00 55              .BYT   128,0,85,128,0,255,
                                  0,0
C84A FF 00 00              .BYT   255,0,0,255,0,0,255,
                                  0
C852 00 FF 00              .BYT   0,255,0,0,100,0,0,17
                                  0
C85A 00 00 AA              .BYT   0,0,170,0,0,170,0,0
C862 00 00 00              .BYT   0,0,0,0,0,0,0,60
C86A 00 00 FF              .BYT   0,0,255,0,0,255,0,0
C872 FF 00 00              .BYT   255,0,0,60,0,2,255,1
                                  28
C87A 02 FF 80              .BYT   002,255,128,2,255,1
                                  28,0,255
C882 80 00 55              .BYT   128,0,85,128,0,255,
                                  0,0
C88A FF 00 00              .BYT   255,0,0,255,0,0,175,
                                  0
C892 00 AF 00              .BYT   0,175,0,0,10,0,0,10
C89A 00 00 0A              .BYT   0,0,10,0,0,10,0,0
C8A2 00 00 00              .BYT   0,0,0,0,0,0,0,60
C8AA 00 00 FF              .BYT   0,0,255,0,0,255,0,0
C8B2 FF 00 00              .BYT   255,0,0,60,0,2,255,1
                                  28
C8BA 02 FF 80              .BYT   002,255,128,2,255,1
                                  28,2,255
C8C2 00 02 55              .BYT   0,2,85,0,0,255,0,0
C8CA FF 00 00              .BYT   255,0,0,255,0,0,250,
                                  0
C8D2 00 FA 00              .BYT   0,250,0,0,160,0,0,16
                                  0
C8DA 00 00 A0              .BYT   0,0,160,0,0,160,0,0
C8E2 00 00 00              .BYT   0,0,0,0,0,0,0,60
C8EA 00 00 FF              .BYT   0,0,255,0,0,235,0,0
C8F2 FF 00 00              .BYT   255,0,0,60,0,2,255,1
                                  28
C8FA 02 FF 80              .BYT   002,255,128,1,255,1
                                  28,1,255
C902 80 00 55              .BYT   128,0,85,128,0,255,
                                  128,0
C90A FF 00 00              .BYT   255,0,0,255,0,0,255,
                                  0
C912 00 FF 00  SPRDAT2     .BYT   0,255,0,0,170,0,0,17
                                  0
C91A 00 00 AA              .BYT   0,0,170,0,0,170,0,0
C922 00 00 00              .BYT   0,0,0,0,0,0,0,60
C92A 00 00 FF              .BYT   0,0,255,0,0,235,0,0
C932 FF 00 00              .BYT   255,0,0,60,0,2,255,1
                                  28
C93A 02 FF 80              .BYT   002,255,128,1,255,1
                                  28,1,255
C942 80 00 55              .BYT   128,0,85,128,0,255,
                                  0,0
```

```
C94A FF  00  00              .BYT     255,0,0,255,0,0,255,
                                      0
C952 00  AF  00              .BYT     0,175,0,0,10,0,0,10
C95A 00  00  0A              .BYT     0,0,10,0,0,10,0,0
C962 00  00  00              .BYT     0,0,0,0,0,0,0,60
C96A 00  00  FF              .BYT     0,0,255,0,0,235,0,0
C972 FF  00  00              .BYT     255,0,0,60,0,2,255,1
                                      28
C97A 02  FF  80              .BYT     002,255,128,2,255,1
                                      28,2,255
C982 80  01  55              .BYT     128,1,85,0,1,255,0,0
C98A FF  00  00              .BYT     255,0,0,255,0,0,255,
                                      0
C992 00  FA  00              .BYT     0,250,0,0,160,0,0,16
                                      0
C99A 00  00  A0              .BYT     0,0,160,0,0,160,0,0
C9A2 00  00  00              .BYT     0,0,0,0,0,0,0,240
C9AA 00  00  FC              .BYT     0,0,252,0,0,188,0,0
C9B2 FC  00  00              .BYT     252,0,0,60,0,0,251,0
C9BA 14  FA  00              .BYT     020,250,0,6,254,128
                                      ,0,255
C9C2 80  00  56              .BYT     128,0,86,0,0,248,0,0
C9CA FF  00  03              .BYT     255,0,3,207,0,15,15,
                                      0
C9D2 0C  0F  00              .BYT     012,15,0,8,3,168,8,0
C9DA 08  08  00              .BYT     008,8,0,8,40,0,0,255
C9E2 00  00  00              .BYT     0,0,0,0,0,0,0,240
C9EA 00  00  FC              .BYT     0,0,252,0,0,188,0,0
C9F2 FC  00  00              .BYT     252,0,0,60,0,0,251,0
C9FA 00  FB  00              .BYT     0,251,0,2,171,0,0,25
                                      5
CA02 00  00  54  SPRDAT3     .BYT     0,0,84,0,0,252,0,0
CA0A FC  00  00              .BYT     252,0,0,60,0,0,60,0
CA12 00  3C  00              .BYT     0,60,0,0,40,0,0,40
CA1A 00  00  28              .BYT     0,0,40,0,0,168,0,255
CA22 00  00  00              .BYT     0,0,0,0,0,0,0,15
CA2A 00  00  3F              .BYT     0,0,63,0,0,62,0,0
CA32 3F  00  00              .BYT     063,0,0,60,0,0,239,0
CA3A 00  AF  20              .BYT     0,175,32,2,191,128,
                                      2,255
CA42 00  00  95              .BYT     0,0,149,0,0,39,0,0
CA4A F5  00  00              .BYT     245,0,0,243,192,0,2
                                      40,240
CA52 00  F0  30              .BYT     0,240,48,42,192,32,
                                      32,0
CA5A 20  20  00              .BYT     032,32,0,32,0,0,40,2
                                      55
CA62 00  00  00              .BYT     0,0,0,0,0,0,0,15
CA6A 00  00  3F              .BYT     0,0,63,0,0,62,0,0
CA72 3F  00  00              .BYT     063,0,0,60,0,0,239,0
CA7A 00  EF  14              .BYT     0,239,20,0,234,144,
                                      0,255
CA82 00  00  15              .BYT     0,0,21,0,0,63,0,0
CA8A 3F  00  00              .BYT     063,0,0,60,0,0,60,0
CA92 00  3C  00              .BYT     0,60,0,0,40,0,0,40
CA9A 00  00  28              .BYT     0,0,40,0,0,42,0,255
CAA2 00  00  00              .BYT     0,0,0,0,0,0,0,0
CAAA 00  00  AA              .BYT     0,0,170,0,2,170,128,
                                      10
CAB2 5A  A0  09              .BYT     090,160,9,90,160,41
                                      ,106,168
CABA 29  AA  A8              .BYT     041,170,168,42,170,
                                      168,42,170
CAC2 A8  2A  AA              .BYT     168,42,170,168,42,1
                                      70,168,42
CACA AA  A8  0A              .BYT     170,168,10,170,160,
                                      10,170,160
CAD2 02  AA  80              .BYT     002,170,128,0,170,0
                                      ,0,0
CADA 00  00  00              .BYT     0,0,0,0,0,0,0,0
```

```
;
; redefine characters
;
CAE2             CHAR        =        *
CAE2 78                      SEI
CAE3 A9  33                  LDA      #51
CAE5 85  01                  STA      1
CAE7 A9  00                  LDA      #<14336
CAE9 85  02                  STA      2
CAEB A9  38                  LDA      #>14336
CAED 85  03                  STA      3
CAEF A9  00                  LDA      #<53248
CAF1 85  04                  STA      4
CAF3 A9  D0                  LDA      #>53248
CAF5 85  05                  STA      5
CAF7 A2  08                  LDX      #8
CAF9 A0  00                  LDY      #0
CAFB B1  04      CHARLOOP    LDA      (4),Y
CAFD 91  02                  STA      (2),Y
CAFF 88                      DEY
CB00 D0  F9                  BNE      CHARLOOP
CB02 E6  05                  INC      5
CB04 E6  03                  INC      3
CB06 CA                      DEX
CB07 D0  F2                  BNE      CHARLOOP
CB09 A9  37                  LDA      #55
CB0B 85  01                  STA      1
CB0D A9  1E                  LDA      #30
CB0F 8D  18  D0              STA      53272
CB12 A0  3F                  LDY      #63
CB14 B9  1F  CB  CHARL2      LDA      CHARDATA,Y
CB17 99  08  39              STA      14336+264,Y
CB1A 88                      DEY
CB1B 10  F7                  BPL      CHARL2
CB1D 58                      CLI
CB1E 60                      RTS
CB1F 80  C0  D8  CHARDATA    .BYT     128,192,216,231,23
                                      1,216,192,128
CB27 FF  7E  18              .BYT     255,126,24,36,36,24
                                      ,24,24
CB2F 01  03  1B              .BYT     1,3,27,231,231,27,3,
                                      1
CB37 18  18  18              .BYT     24,24,24,36,36,24,1
                                      26,255
CB3F 7F  3F  2A              .BYT     127,63,42,31,21,15,
                                      10,7
CB47 FF  FF  A5              .BYT     255,255,165,255,82,
                                      255,165,255
CB4F F8  F0  50              .BYT     248,240,80,224,160,
                                      192,64,128
CB57 1C  14  1D              .BYT     28,20,29,63,92,60,2
                                      0,54
;
; collision subroutine
;
CB5F             COLLISION   =        *
CB5F AD  1E  D0              LDA      53278
CB62 29  01                  AND      #1
CB64 F0  65                  BEQ      HOME
CB66 A2  07                  LDX      #7
CB68 A9  FF                  LDA      #255
CB6A 85  10                  STA      TEMP
CB6C A9  09                  LDA      #9
CB6E 85  A4                  STA      SPRNUM
CB70 BD  96  03  COLLOOP     LDA      SPOTTER,X
CB73 D0  39                  BNE      NOTREC
CB75 AD  50  03              LDA      XPOSH
CB78 DD  50  03              CMP      XPOSH,X
CB7B D0  31                  BNE      NOTREC
CB7D AD  3C  03              LDA      XPOS
```

```
CB80 38                        SEC
CB81 FD 3C 03                  SBC   XPOS,X
CB84 10 05                     BPL   PLUS
CB86 49 FF                     EOR   #255
CB88 18                        CLC
CB89 69 01                     ADC   #1
CB8B C9 0F      PLUS           CMP   #15
CB8D B0 1F                     BCS   NOTREC
CB8F 85 A5                     STA   TEMP2
CB91 AD 46 03                  LDA   YPOS
CB94 38                        SEC
CB95 FD 46 03                  SBC   YPOS,X
CB98 10 05                     BPL   PLUS2
CB9A 49 FF                     EOR   #255
CB9C 18                        CLC
CB9D 69 01                     ADC   #1
CB9F C9 0F      PLUS2          CMP   #15
CBA1 B0 0B                     BCS   NOTREC
CBA3 18                        CLC
CBA4 65 A5                     ADC   TEMP2
CBA6 C5 10                     CMP   TEMP
CBA8 B0 04                     BCS   NOTREC
CBAA 85 10                     STA   TEMP
CBAC 86 A4                     STX   SPRNUM
CBAE CA         NOTREC         DEX
CBAF D0 BF                     BNE   COLLOOP
CBB1 A6 A4                     LDX   SPRNUM
CBB3 E0 09                     CPX   #9
CBB5 F0 14                     BEQ   HOME
CBB7 A9 02                     LDA   #2
CBB9 9D 96 03                  STA   SPOTTER,X
CBBC 86 A6                     STX   SPRNUM2
CBBE E4 A7                     CPX   GOODNUM
CBC0 D0 0A                     BNE   DEADDUCK
CBC2 BD 27 D0                  LDA   53287,X
CBC5 29 0F                     AND   #15
CBC7 C9 01                     CMP   #1
CBC9 D0 01                     BNE   DEADDUCK
CBCB 60         HOME           RTS
CBCC A9 05      DEADDUCK       LDA   #5
CBCE 85 10                     STA   TEMP
CBD0 A9 14                     LDA   #20
CBD2 8D 0B D4                  STA   $D40B
CBD5 A9 00                     LDA   #0
CBD7 8D 08 D4                  STA   $D408
CBDA A9 0A                     LDA   #10
CBDC 8D 0C D4                  STA   $D40C
CBDF A9 15                     LDA   #21
CBE1 8D 0B D4                  STA   $D40B
CBE4 A2 00                     LDX   #0
CBE6 AC 1B D4   DEADLOOP       LDY   $D41B
CBE9 A9 00                     LDA   #0
CBEB 99 00 33                  STA   13056,Y
CBEE 99 00 34                  STA   13312,Y
CBF1 8E 08 D4                  STX   $D408
CBF4 99 80 34                  STA   13696-256,Y
CBF7 20 31 CC                  JSR   DELAY
CBFA CA                        DEX
CBFB D0 E9                     BNE   DEADLOOP
CBFD C6 10                     DEC   TEMP
CBFF D0 E5                     BNE   DEADLOOP
CC01 68                        PLA
CC02 68                        PLA
CC03 20 CA CC                  JSR   LASERED
CC06 CE B3 02                  DEC   NUMMEN
CC09 F0 03                     BEQ   FINISHED
CC0B 4C B8 C0                  JMP   BUCK
CC0E A9 00      FINISHED       LDA   #0
CC10 8D 15 D0                  STA   53269
CC13 A2 0B                     LDX   #11
```

```
CC15 A0 0F                     LDY   #15
CC17 18                        CLC
CC18 20 F0 FF                  JSR   $FFF0
CC1B A9 20                     LDA   #32
CC1D 8D E0 07                  STA   2016
CC20 A9 0F                     LDA   #<MESS3
CC22 A0 CD                     LDY   #>MESS3
CC24 20 1E AB                  JSR   $AB1E
CC27 AD 00 DC   WUT            LDA   $DC00
CC2A 29 10                     AND   #16
CC2C D0 F9                     BNE   WUT
CC2E 4C 05 C0                  JMP   START
CC31 A0 64      DELAY          LDY   #100
CC33 88         V2             DEY
CC34 D0 FD                     BNE   V2
CC36 60                        RTS
;
; see if touching forbidden zone
;
CC37            FORBIDDEN      =     *
CC37 A5 A9                     LDA   FLAG2
CC39 0A                        ASL
CC3A 0A                        ASL
CC3B 0A                        ASL
CC3C 18                        CLC
CC3D 69 17                     ADC   #23
CC3F CD 46 03                  CMP   YPOS
CC42 B0 0F                     BCS   NOTHIT
CC44 AD 50 03                  LDA   XPOSH
CC47 D0 0A                     BNE   NOTHIT
CC49 AD 3C 03                  LDA   XPOS
CC4C C9 2A                     CMP   #LEFTLIMIT-13
CC4E B0 03                     BCS   NOTHIT
CC50 4C CC CB                  JMP   DEADDUCK
CC53 60         NOTHIT         RTS
; sound routines
;
CC54 A2 00      SETSID         LDX   #0
CC56 8A                        TXA
CC57 9D 00 D4   CLRSID         STA   $D400,X
CC5A E8                        INX
CC5B E0 1A                     CPX   #26
CC5D D0 F8                     BNE   CLRSID
CC5F A9 14                     LDA   #20
CC61 8D 01 D4                  STA   $D401
CC64 A9 42                     LDA   #66
CC66 8D 05 D4                  STA   $D405
CC69 A9 08                     LDA   #8
CC6B 8D 03 D4                  STA   $D403
CC6E A9 05                     LDA   #5
CC70 8D 0C D4                  STA   $D40C
CC73 A9 05                     LDA   #5
CC75 8D 0A D4                  STA   $D40A
CC78 A9 FF                     LDA   #255
CC7A 8D 0F D4                  STA   $D40F
CC7D A9 0F                     LDA   #15
CC7F 8D 18 D4                  STA   $D418
CC82 60                        RTS
CC83 00         DIRSV          .BYT  0
CC84 00         STEPCOUNT      .BYT  0
;
; do that timer thing
;
CC85            TIMER          =     *
CC85 CE AA 02                  DEC   FLAG2LOW
CC88 D0 1A                     BNE   GETOUT
CC8A A9 32                     LDA   #50
CC8C 8D AA 02                  STA   FLAG2LOW
CC8F A5 A9                     LDA   FLAG2
CC91 C9 1A                     CMP   #26
```

```
CC93 F0  0F                BEQ    GETOUT
CC95 C6  A9                DEC    FLAG2
CC97 10  06                BPL    OKEY
CC99 A9  00                LDA    #0
CC9B 85  A9                STA    FLAG2
CC9D F0  05                BEQ    GETOUT
CC9F A0  07      OKEY      LDY    #7
CCA1 20  A5  CC            JSR    FUT
CCA4 60          GETOUT    RTS
CCA5 84  10      FUT       STY    TEMP
CCA7 A8                    TAY
CCA8 A9  D8                LDA    #<(55296-40)
CCAA 85  02                STA    2
CCAC A9  D7                LDA    #>(55296-40)
CCAE 85  03                STA    3
CCB0 A5  02      TROOP     LDA    2
CCB2 18                    CLC
CCB3 69  28                ADC    #40
CCB5 85  02                STA    2
CCB7 A5  03                LDA    3
CCB9 69  00                ADC    #0
CCBB 85  03                STA    3
CCBD 88                    DEY
CCBE D0  F0                BNE    TROOP
CCC0 A5  10                LDA    TEMP
CCC2 A0  02                LDY    #2
CCC4 91  02      HHH       STA    (2),Y
CCC6 88                    DEY
CCC7 10  FB                BPL    HHH
CCC9 60                    RTS
CCCA A9  01      LASERED   LDA    #1
CCCC 8D  B4  02            STA    IGNORE
CCCF 20  48  C1            JSR    SHOOT
CCD2 A5  A2                LDA    162
CCD4 C5  A2      WTY       CMP    162
CCD6 F0  FC                BEQ    WTY
CCD8 A5  0A                LDA    FIREFLAG
CCDA D0  EE                BNE    LASERED
CCDC 60                    RTS
CCDD 9E  92  4C  MESS1     .ASC   "{YEL}{O
                                  FF}L A S E R"
CCE8 00                    .BYT   0
CCE9 42  20  45  MESS2     .ASC   "B E A M [<3>]{3 DO
                                  WN}{7 LEFT}LEVE
                                  L {3 DOWN}{10 LE
                                  FT}"
CD0F 12  9F  48  MESS3     .ASC   "{RVS}{CYN}HIT FI
                                  REBUTTON"
CD1F 00                    .BYT   0
CD20 9E  4C  45  MESS4     .ASC   "{YEL}LEVEL COM
                                  PLETED{2 DOWN}{
                                  12 LEFT}BONUS "
CD44 00                    .BYT   0
CD45 AD  B7  02  BONUS     LDA    VOUNT
CD48 F0  2D                BEQ    QUOT
CD4A CE  B6  02            DEC    VOUNTL
CD4D D0  28                BNE    QUOT
CD4F CE  B8  02            DEC    VOUNTH
CD52 D0  23                BNE    QUOT
CD54 A2  03                LDX    #3
CD56 8E  B8  02            STX    VOUNTH
CD59 CE  B7  02            DEC    VOUNT
CD5C AD  B7  02            LDA    VOUNT
CD5F D0  10                BNE    NT0
CD61 A9  18                LDA    #24
CD63 85  A9                STA    FLAG2
CD65 A9  01                LDA    #1
CD67 A2  06                LDX    #6
CD69 9D  28  D0  UY        STA    53288,X
CD6C CA                    DEX
```

```
CD6D 10  FA                BPL    UY
CD6F A9  00                LDA    #0
CD71 AA          NT0       TAX
CD72 A9  20                LDA    #32
CD74 9D  C0  07            STA    1984,X
CD77 60          QUOT      RTS
CD78 A2  18      SCOROUT   LDX    #24
CD7A A0  1A                LDY    #26
CD7C 18                    CLC
CD7D 20  F0  FF            JSR    $FFF0
CD80 AD  BA  02            LDA    SCOREH
CD83 C9  FF                CMP    #$FF
CD85 F0  07                BEQ    TP
CD87 AE  B9  02            LDX    SCOREL
CD8A 20  CD  BD            JSR    $BDCD
CD8D 60                    RTS
CD8E A9  98      TP        LDA    #<MESS8
CD90 A0  CD                LDY    #>MESS8
CD92 20  1E  AB            JSR    $AB1E
CD95 68                    PLA
CD96 68                    PLA
CD97 60                    RTS
CD98 93  11  11  MESS8     .ASC   "{CLR}{3 DOWN
                                  } CONGRATULATIO
                                  NS, YOU ARE ON
                                  E OF THE FEW "
CDC5 54  4F  20            .ASC   "TO WIN AT LASE
                                  R BEAM."
CDDA 00                    .BYT   0
```

Whirlybird Source Code

Source code by Philip I. Nelson

```
; current egg position
0801          EGG         =     251
; coord for egg color
0801          EGCOL       =     253
;
;--- load & run like basic -------
;
0801                      *=    2049
;
; following bytes are a basic line
;
; '0 sys2080 rem 5del whirlybird'
;
0801 1B 08 00             .BYT  27,8,0,0,158,50,48,5
                                6,48,58
080B 8F 22 14             .BYT  143,34,20,20,20,20,
                                20,32
0813 57 48 49             .BYT  87,72,73,82,76,89,6
                                6,73,82,68
081D 00 00 00             .BYT  0,0,0
;
;
;--- clear graphics -------------
;
0820 A9 93                LDA   #$93
0822 20 D2 FF             JSR   $FFD2
0825 A2 00                LDX   #0
0827 8E 21 D0             STX   53281
082A 8E 20 D0             STX   53280
082D E8                   INX
082E 8E 1B D0             STX   53275
0831 A9 03                LDA   #3
0833 8D 86 02             STA   646
0836 A2 10                LDX   #16
0838 A9 00                LDA   #0
083A 9D 00 D0  SPRCLR     STA   53248,X
083D CA                   DEX
083E 10 FA                BPL   SPRCLR
;
;--- clear & load SID chip -------
;
0840 A2 18                LDX   #24
0842 9D 00 D4  SIDCLR     STA   54272,X
0845 CA                   DEX
0846 10 FA                BPL   SIDCLR
0848 A2 18                LDX   #24
084A BD 3B 14  SIDLD      LDA   SID,X
084D 9D 00 D4             STA   54272,X
0850 CA                   DEX
0851 10 F7                BPL   SIDLD
;
;--- install new character set ---
;
0853 A9 7F                LDA   #127
0855 8D 0D DC             STA   56333
0858 A9 33                LDA   #51
085A 85 01                STA   1
085C A9 00                LDA   #0
085E 85 FB                STA   251
0860 A9 D0                LDA   #$D0
```

```
0862 85 FC                STA   252
0864 A9 00                LDA   #0
0866 85 FD                STA   253
0868 A9 38                LDA   #$38
086A 85 FE                STA   254
086C A0 00                LDY   #0
086E B1 FB     COPY       LDA   (251),Y
0870 91 FD                STA   (253),Y
0872 E6 FB                INC   251
0874 D0 02                BNE   NOH1
0876 E6 FC                INC   252
0878 E6 FD     NOH1       INC   253
087A D0 02                BNE   NOH2
087C E6 FE                INC   254
087E 38        NOH2       SEC
087F A5 FC                LDA   252
0881 C9 D8                CMP   #$D8
0883 90 E9                BCC   COPY
0885 A9 37                LDA   #55
0887 85 01                STA   1
0889 A9 81                LDA   #129
088B 8D 0D DC             STA   56333
088E A2 08                LDX   #8
0890 BD D0 13  NWCHR      LDA   CRDATA,X
0893 9D D0 3A             STA   15056,X
0896 BD D8 13             LDA   CRDA1,X
0899 9D 50 3B             STA   15184,X
089C BD E0 13             LDA   CRDA2,X
089F 9D 50 39             STA   14672,X
08A2 CA                   DEX
08A3 10 EB                BPL   NWCHR
08A5 A9 1F                LDA   #31
08A7 8D 18 D0             STA   53272
;
;--- install sprite shapes -------
;
08AA A9 11                LDA   #<SPRDATA
08AC 85 FB                STA   251
08AE A9 11                LDA   #>SPRDATA
08B0 85 FC                STA   252
08B2 A9 00                LDA   #<15360
08B4 85 FD                STA   253
08B6 A9 3C                LDA   #>15360
08B8 85 FE                STA   254
08BA A0 00                LDY   #0
08BC B1 FB     PTSPR      LDA   (251),Y
08BE 91 FD                STA   (253),Y
08C0 E6 FB                INC   251
08C2 D0 02                BNE   NO52
08C4 E6 FC                INC   252
08C6 E6 FD     NO52       INC   253
08C8 D0 02                BNE   NO54
08CA E6 FE                INC   254
08CC 38        NO54       SEC
08CD A5 FB                LDA   251
08CF E9 D0                SBC   #<ENDATA
08D1 A5 FC                LDA   252
08D3 E9 13                SBC   #>ENDATA
08D5 90 E5                BCC   PTSPR
```

```
;--- change IRQ vector for
; interrupt-driven bird movement --
;
08D7 78                      SEI
08D8 A9 DB                   LDA    #<INTERRUPT
08DA 8D 14 03                STA    $0314
08DD A9 0F                   LDA    #>INTERRUPT
08DF 8D 15 03                STA    $0315
08E2 58                      CLI
;
;--- some startup values ---------
;
08E3 A9 00                   LDA    #0
08E5 8D 69 03                STA    GOFLAG
08E8 A9 F0                   LDA    #240
08EA 8D 3C 03                STA    SHAPE
08ED A9 00                   LDA    #0
08EF 8D 57 03                STA    SCORE
08F2 8D 58 03                STA    SCORE+1
08F5 20 3C 0A                JSR    BOAT
08F8 A9 14                   LDA    #20
08FA 8D 0B D4                STA    54283
08FD A9 15                   LDA    #21
08FF 8D 0B D4                STA    54283
;
;----- reentry point ----------
;
0902 20 C6 0E   RESTART      JSR    STARTUP
; new game if eggs all gone.
0905 AD 62 03   NWEGG        LDA    BALLS
0908 F0 F8                   BEQ    RESTART
; missed - subtract one egg
; and reverse egg direction.
090A CE 62 03                DEC    BALLS
090D A9 01                   LDA    #1
090F 8D 51 03                STA    EGUD
; new dir'n for filter sweep.
0912 A9 00                   LDA    #0
0914 8D 5F 03                STA    FDIR
0917 AD 57 03                LDA    SCORE
091A 0D 58 03                ORA    SCORE+1
091D E9 0A                   SBC    #10
091F 90 14                   BCC    NOTAK
; if score>9 then score=score-10
0921 38                      SEC
0922 AD 57 03                LDA    SCORE
0925 E9 0A                   SBC    #10
0927 8D 57 03                STA    SCORE
092A AD 58 03                LDA    SCORE+1
092D E9 00                   SBC    #0
092F 8D 58 03                STA    SCORE+1
0932 20 2E 0E                JSR    DOSCR
; wait for player to press button
0935            NOTAK        =      *
;sparkle sidebars while waiting
0935 20 CA 10                JSR    SPARKLE
0938 AD 00 DC                LDA    56320
093B 29 10                   AND    #16
093D D0 F6                   BNE    NOTAK
; choose new egg direction
093F 20 F8 0B                JSR    EGNEW
0942 20 E7 0C                JSR    EGPUT
;
;------ main loop ----------------
;
0945 20 9F FF   GETSTICK     JSR    $FF9F
0948 20 E4 FF                JSR    $FFE4
094B C9 85                   CMP    #133
;pause if f1
094D F0 06                   BEQ    STOP
```

```
094F C9 88                   CMP    #136
;resume if f7
0951 F0 0A                   BEQ    GO
0953 D0 0D                   BNE    KINT
0955 A9 01      STOP         LDA    #1
0957 8D 69 03                STA    GOFLAG
095A 4C 45 09                JMP    GETSTICK
095D A9 00      GO           LDA    #0
095F 8D 69 03                STA    GOFLAG
0962 AD 69 03   KINT         LDA    GOFLAG
0965 D0 DE                   BNE    GETSTICK
0967 AD 46 03                LDA    BIRDSPEED
096A D0 08                   BNE    JUJ
096C A9 00                   LDA    #0
096E 8D 68 03                STA    GFLAG
0971 4C 7E 09                JMP    BOKC
0974 AD 68 03   JUJ          LDA    GFLAG
0977 D0 05                   BNE    BOKC
0979 A9 01                   LDA    #1
097B 8D 68 03                STA    GFLAG
; check for game over
097E 20 97 0E   BOKC         JSR    DONCHK
0981 AD 5B 03                LDA    DNFLAG
0984 D0 12                   BNE    PLAY
; game over, start a new one
0986 20 02 0E                JSR    UPSCORE
0989 20 02 0E                JSR    UPSCORE
098C 20 02 0E                JSR    UPSCORE
098F 20 02 0E                JSR    UPSCORE
0992 20 02 0E                JSR    UPSCORE
0995 4C 02 09                JMP    RESTART
; game not over, check for a
; collision between bird & egg.
0998 AD 55 03   PLAY         LDA    COLSAV
099B F0 32                   BEQ    NOCOL
; yes, the bird did hit the egg
099D 20 39 0D                JSR    BOUNCE
09A0 20 ED 0B                JSR    EGERASE
; decide which way to bounce egg,
; based on current joystick dir'n.
09A3 AD 00 DC                LDA    56320
09A6 29 0F                   AND    #15
09A8 C9 0B                   CMP    #11
09AA F0 07                   BEQ    LDIR
09AC C9 07                   CMP    #7
09AE F0 0B                   BEQ    RDIR
09B0 4C C0 09                JMP    NODIR
09B3 A9 00      LDIR         LDA    #0
09B5 8D 53 03                STA    EGLR
09B8 4C C0 09                JMP    NODIR
09BB A9 01      RDIR         LDA    #1
09BD 8D 53 03                STA    EGLR
; add to score, cuz we hit the egg
09C0 20 02 0E   NODIR        JSR    UPSCORE
09C3 20 51 0A                JSR    SLOWDOWN
; reverse egg u/d direction
09C6 A2 01                   LDX    #1
09C8 8E 51 03                STX    EGUD
09CB 8A                      TXA
09CC 4C D1 09                JMP    DOCOL
; entry point when no collision
09CF A9 00      NOCOL        LDA    #0
09D1 8D 56 03   DOCOL        STA    COLFLAG
09D4 20 51 0A                JSR    SLOWDOWN
; now we can move the egg
09D7 20 38 0C                JSR    EGMOVE
; if there was no collision with a
; poison deal, go move bird.
09DA AD 63 03                LDA    PFLAG
09DD F0 03                   BEQ    DOJOY
```

```
; oops, hit poison gob, so we
; need a new egg.
09DF 4C 05 09              JMP   NWEGG
; read stick to decide whether to
; accelerate, decelerate or
; reverse the bird's movement.
09E2 AD 00 DC  DOJOY       LDA   56320
09E5 29 0F                 AND   #15
09E7 C9 0B                 CMP   #11
09E9 C9 0B                 CMP   #11
09EB F0 24                 BEQ   LPULL
09ED C9 07                 CMP   #7
09EF F0 18                 BEQ   RPULL
09F1 AD 46 03              LDA   BIRDSPEED
09F4 D0 03                 BNE   CHOC
09F6 4C 45 09              JMP   GETSTICK
09F9 CE 64 03  CHOC        DEC   SLFLAG
09FC AD 64 03              LDA   SLFLAG
09FF D0 2D                 BNE   NOSDWN
0A01 A9 03                 LDA   #3
0A03 8D 64 03              STA   SLFLAG
0A06 4C 2B 0A              JMP   BSLOW
0A09 AD 41 03  RPULL       LDA   LEX
0A0C D0 08                 BNE   FASTER
0A0E 4C 26 0A              JMP   SLOWER
0A11 AD 41 03  LPULL       LDA   LEX
0A14 D0 10                 BNE   SLOWER
0A16           FASTER      =     *
0A16 AD 46 03              LDA   BIRDSPEED
0A19 C9 0C                 CMP   #12
0A1B 90 03                 BCC   FAH
0A1D 4C 45 09              JMP   GETSTICK
0A20 EE 46 03  FAH         INC   BIRDSPEED
0A23 4C 45 09              JMP   GETSTICK
0A26           SLOWER      =     *
0A26 AD 46 03              LDA   BIRDSPEED
0A29 F0 06                 BEQ   REVERSE
0A2B CE 46 03  BSLOW       DEC   BIRDSPEED
0A2E 4C 45 09  NOSDWN      JMP   GETSTICK
0A31           REVERSE     =     *
0A31 AD 41 03              LDA   LEX
0A34 49 01                 EOR   #1
0A36 8D 41 03              STA   LEX
0A39 4C 45 09              JMP   GETSTICK
;--- end of main loop ------------
;
;--- sound ----------------------
;
; reset registers for
; initial sound.
0A3C           BOAT        =     *
0A3C A9 9B                 LDA   #155
0A3E 8D 07 D4              STA   54279
0A41 A9 03                 LDA   #3
0A43 8D 08 D4              STA   54280
0A46 A9 0B                 LDA   #11
0A48 8D 00 D4              STA   54272
0A4B A9 07                 LDA   #7
0A4D 8D 01 D4              STA   54273
0A50 60                    RTS
;
;--- slow that bird -------------
;
; slows bird down when you're not
; pressing on the stick.
0A51           SLOWDOWN    =     *
0A51 EE 47 03              INC   COUNT
0A54 AD 47 03              LDA   COUNT
0A57 D0 F8                 BNE   SLOWDOWN
0A59 EE 48 03  COUNTUP     INC   COUNT+1
```

```
0A5C AD 48 03              LDA   COUNT+1
0A5F C9 0F                 CMP   #15
0A61 90 EE                 BCC   SLOWDOWN
0A63 A9 00     COUNTDONE   LDA   #0
0A65 8D 48 03              STA   COUNT+1
0A68 60                    RTS
;
;--- draw playfield screen ------
;
0A69           DRAWSCRN    =     *
0A69 A9 C0                 LDA   #<1984
0A6B 8D 49 03              STA   DLIMIT
0A6E A9 07                 LDA   #>1984
0A70 8D 4A 03              STA   DLIMIT+1
0A73 A9 50                 LDA   #<1104
0A75 85 03                 STA   3
0A77 A9 04                 LDA   #>1104
0A79 85 04                 STA   4
0A7B A9 6A                 LDA   #106
0A7D 8D 4B 03              STA   CHAR
0A80 A9 0D                 LDA   #13
0A82 8D 4C 03              STA   COL
0A85 20 90 0B              JSR   DRAWDOWN
0A88 A9 E7                 LDA   #<2023
0A8A 8D 49 03              STA   DLIMIT
0A8D A9 07                 LDA   #>2023
0A8F 8D 4A 03              STA   DLIMIT+1
0A92 A9 77                 LDA   #<1143
0A94 85 03                 STA   3
0A96 A9 04                 LDA   #>1143
0A98 85 04                 STA   4
0A9A A9 6A                 LDA   #106
0A9C 8D 4B 03              STA   CHAR
0A9F 20 90 0B              JSR   DRAWDOWN
0AA2 A9 F9                 LDA   #<1785
0AA4 85 03                 STA   3
0AA6 A9 06                 LDA   #>1785
0AA8 85 04                 STA   4
0AAA A9 1E                 LDA   #<1822
0AAC 8D 49 03              STA   DLIMIT
0AAF A9 07                 LDA   #>1822
0AB1 8D 4A 03              STA   DLIMIT+1
0AB4 A9 5A                 LDA   #90
0AB6 8D 4B 03              STA   CHAR
0AB9 A9 07                 LDA   #7
0ABB 8D 4C 03              STA   COL
0ABE 20 C2 0B              JSR   DRAWRIGHT
0AC1 A9 21                 LDA   #<1825
0AC3 85 03                 STA   3
0AC5 A9 07                 LDA   #>1825
0AC7 85 04                 STA   4
0AC9 A9 46                 LDA   #<1862
0ACB 8D 49 03              STA   DLIMIT
0ACE A9 07                 LDA   #>1862
0AD0 8D 4A 03              STA   DLIMIT+1
0AD3 A9 06                 LDA   #6
0AD5 8D 4C 03              STA   COL
0AD8 20 C2 0B              JSR   DRAWRIGHT
0ADB A9 49                 LDA   #<1865
0ADD 85 03                 STA   3
0ADF A9 07                 LDA   #>1865
0AE1 85 04                 STA   4
0AE3 A9 6E                 LDA   #<1902
0AE5 8D 49 03              STA   DLIMIT
0AE8 A9 07                 LDA   #>1902
0AEA 8D 4A 03              STA   DLIMIT+1
0AED A9 05                 LDA   #5
0AEF 8D 4C 03              STA   COL
0AF2 20 C2 0B              JSR   DRAWRIGHT
0AF5 A9 71                 LDA   #<1905
```

```
0AF7 85 03              STA    3
0AF9 A9 07              LDA    #>1905
0AFB 85 04              STA    4
0AFD A9 96              LDA    #<1942
0AFF 8D 49 03           STA    DLIMIT
0B02 A9 07              LDA    #>1942
0B04 8D 4A 03           STA    DLIMIT+1
0B07 A9 07              LDA    #7
0B09 8D 4C 03           STA    COL
0B0C 20 C2 0B           JSR    DRAWRIGHT
0B0F A9 99              LDA    #<1945
0B11 85 03              STA    3
0B13 A9 07              LDA    #>1945
0B15 85 04              STA    4
0B17 A9 BE              LDA    #<1982
0B19 8D 49 03           STA    DLIMIT
0B1C A9 07              LDA    #>1982
0B1E 8D 4A 03           STA    DLIMIT+1
0B21 A9 06              LDA    #6
0B23 8D 4C 03           STA    COL
0B26 20 C2 0B           JSR    DRAWRIGHT
0B29 A9 C1              LDA    #<1985
0B2B 85 03              STA    3
0B2D A9 07              LDA    #>1985
0B2F 85 04              STA    4
0B31 A9 E6              LDA    #<2022
0B33 8D 49 03           STA    DLIMIT
0B36 A9 07              LDA    #>2022
0B38 8D 4A 03           STA    DLIMIT+1
0B3B A9 03              LDA    #3
0B3D 8D 4C 03           STA    COL
0B40 20 C2 0B           JSR    DRAWRIGHT
0B43 A9 57              LDA    #87
0B45 8D 02 07           STA    1794
0B48 8D 15 07           STA    1813
0B4B 8D 77 07           STA    1911
0B4E 8D 90 07           STA    1936
0B51 8D C4 07           STA    1988
0B54 8D CA 07           STA    1994
0B57 8D D0 07           STA    2000
0B5A 8D 58 07           STA    1880
0B5D 8D 60 07           STA    1888
0B60 8D D7 07           STA    2007
0B63 8D DD 07           STA    2013
0B66 8D E3 07           STA    2019
0B69 A9 01              LDA    #1
0B6B 8D 02 DB           STA    1794+54272
0B6E 8D 15 DB           STA    1813+54272
0B71 8D 77 DB           STA    1911+54272
0B74 8D 90 DB           STA    1936+54272
0B77 8D C4 DB           STA    1988+54272
0B7A 8D CA DB           STA    1994+54272
0B7D 8D D0 DB           STA    2000+54272
0B80 8D D7 DB           STA    2007+54272
0B83 8D 58 DB           STA    1880+54272
0B86 8D 60 DB           STA    1888+54272
0B89 8D DD DB           STA    2013+54272
0B8C 8D E3 DB           STA    2019+54272
0B8F 60                 RTS
;
;----- rtn for screen draw ------
;
0B90           DRAWDOWN =      *
0B90 A0 00              LDY    #0
0B92 AD 4B 03  DRAWSOME LDA    CHAR
0B95 91 03              STA    (3),Y
0B97 A5 03              LDA    3
0B99 85 05              STA    5
0B9B 18                 CLC
0B9C A5 04              LDA    4
0B9E 69 D4              ADC    #212
0BA0 85 06              STA    6
0BA2 AD 4C 03           LDA    COL
0BA5 91 05              STA    (5),Y
0BA7 18                 CLC
0BA8 A5 03              LDA    3
0BAA 69 28              ADC    #40
0BAC 85 03              STA    3
0BAE A5 04              LDA    4
0BB0 69 00              ADC    #0
0BB2 85 04              STA    4
0BB4 38                 SEC
0BB5 AD 49 03           LDA    DLIMIT
0BB8 E5 03              SBC    3
0BBA AD 4A 03           LDA    DLIMIT+1
0BBD E5 04              SBC    4
0BBF B0 D1              BCS    DRAWSOME
0BC1 60                 RTS
;
;----- rtn for screen draw ------
;
0BC2           DRAWRIGHT =     *
0BC2 A0 00              LDY    #0
0BC4 AD 4B 03  DRAWR    LDA    CHAR
0BC7 91 03              STA    (3),Y
0BC9 A5 03              LDA    3
0BCB 85 05              STA    5
0BCD 18                 CLC
0BCE A5 04              LDA    4
0BD0 69 D4              ADC    #212
0BD2 85 06              STA    6
0BD4 AD 4C 03           LDA    COL
0BD7 91 05              STA    (5),Y
0BD9 E6 03              INC    3
0BDB D0 02              BNE    NO4INK
0BDD E6 04              INC    4
0BDF 38        NO4INK   SEC
0BE0 AD 49 03           LDA    DLIMIT
0BE3 E5 03              SBC    3
0BE5 AD 4A 03           LDA    DLIMIT+1
0BE8 E5 04              SBC    4
0BEA B0 D8              BCS    DRAWR
0BEC 60                 RTS
;
;----- erase egg ---------------
;
0BED           EGERASE  =      *
0BED A9 20              LDA    #32
0BEF A0 00              LDY    #0
0BF1 91 FB              STA    (EGG),Y
0BF3 A9 01              LDA    #1
0BF5 91 FD              STA    (EGCOL),Y
0BF7 60                 RTS
;
;----- new egg ----------------
;
0BF8           EGNEW    =      *
0BF8 20 34 0C           JSR    GETRAND
0BFB C9 23              CMP    #35
0BFD B0 F9              BCS    EGNEW
0BFF 8D 4F 03           STA    TEMP
0C02 18                 CLC
0C03 A9 CA              LDA    #202
0C05 6D 4F 03           ADC    TEMP
0C08 85 FB              STA    EGG
0C0A A9 04              LDA    #4
0C0C 69 00              ADC    #0
0C0E 85 FC              STA    EGG+1
0C10 18                 CLC
0C11 A5 FB              LDA    EGG
```

```
0C13 69 00                ADC   #0
0C15 85 FD                STA   EGCOL
0C17 A5 FC                LDA   EGG+1
0C19 69 D4                ADC   #212
0C1B 85 FE                STA   EGCOL+1
0C1D A9 01                LDA   #1
0C1F 8D 51 03             STA   EGUD
0C22 20 34 0C             JSR   GETRAND
0C25 30 07                BMI   EGRT
0C27 A9 00                LDA   #0
0C29 8D 53 03             STA   EGLR
0C2C F0 05                BEQ   DEXIT
0C2E A9 01      EGRT      LDA   #1
0C30 8D 53 03             STA   EGLR
0C33 60         DEXIT     RTS
;
;-- read random number generator --
;
0C34 AD 1B D 4  GETRAND   LDA   $D41B
0C37 60                   RTS
;
;----- move egg ----------------
;
0C38            EGMOVE    =     *
; clear poison flag, erase egg
0C38 A9 00                LDA   #0
0C3A 8D 63 03             STA   PFLAG
0C3D 20 ED 0B             JSR   EGERASE
; check l/r flag to see which way
0C40 AD 53 03             LDA   EGLR
0C43 D0 16                BNE   EGRIT
; move that pup to the left
0C45 20 71 0C             JSR   EGL
0C48 A0 00                LDY   #0
0C4A B1 FB                LDA   (EGG),Y
; check collision w/ sidebar
0C4C C9 6A                CMP   #106
0C4E F0 02                BEQ   LSIT
0C50 D0 3B                BNE   EGUP
0C52 20 7F 0C   LSIT      JSR   EGR
0C55 20 20 0D             JSR   REVLR
0C58 4C 38 0C             JMP   EGMOVE
; move egg to the right this time
0C5B 20 7F 0C   EGRIT     JSR   EGR
0C5E A0 00                LDY   #0
0C60 B1 FB                LDA   (EGG),Y
0C62 C9 6A                CMP   #106
0C64 F0 02                BEQ   RSIT
0C66 D0 25                BNE   EGUP
0C68 20 71 0C   RSIT      JSR   EGL
0C6B 20 20 0D             JSR   REVLR
0C6E 4C 38 0C             JMP   EGMOVE
;
0C71 38         EGL       SEC
0C72 A5 FB                LDA   EGG
0C74 E9 01                SBC   #1
0C76 85 FB                STA   EGG
0C78 A5 FC                LDA   EGG+1
0C7A E9 00                SBC   #0
0C7C 85 FC                STA   EGG+1
0C7E 60                   RTS
;
0C7F 18         EGR       CLC
0C80 A5 FB                LDA   EGG
0C82 69 01                ADC   #1
0C84 85 FB                STA   EGG
0C86 A5 FC                LDA   EGG+1
0C88 69 00                ADC   #0
0C8A 85 FC                STA   EGG+1
0C8C 60                   RTS
```

```
; egg is moving up
0C8D A5 FB      EGUP      LDA   EGG
0C8F 85 AA                STA   $AA
0C91 A5 FC                LDA   EGG+1
0C93 85 AB                STA   $AB
0C95 AD 51 03             LDA   EGUD
0C98 D0 1B                BNE   EGDN
0C9A 38                   SEC
0C9B A5 FB                LDA   EGG
0C9D E9 51                SBC   #<1105
0C9F A5 FC                LDA   EGG+1
0CA1 E9 04                SBC   #>1105
0CA3 90 5A                BCC   EGTOP
0CA5 38                   SEC
0CA6 A5 FB                LDA   EGG
0CA8 E9 28                SBC   #40
0CAA 85 FB                STA   EGG
0CAC A5 FC                LDA   EGG+1
0CAE E9 00                SBC   #0
0CB0 85 FC                STA   EGG+1
0CB2 4C CD 0C             JMP   EGCHEK
; egg moving down
0CB5 38         EGDN      SEC
0CB6 A5 FB                LDA   EGG
0CB8 E9 C0                SBC   #<1984
0CBA A5 FC                LDA   EGG+1
0CBC E9 07                SBC   #>1984
0CBE B0 4D                BCS   EGBOT
0CC0 18                   CLC
0CC1 A5 FB                LDA   EGG
0CC3 69 28                ADC   #40
0CC5 85 FB                STA   EGG
0CC7 A5 FC                LDA   EGG+1
0CC9 69 00                ADC   #0
0CCB 85 FC                STA   EGG+1
; deliberate fall-through here
;
;--- check egg/character collisions
0CCD            EGCHEK    =     *
0CCD B1 FB                LDA   (EGG),Y
; 87=poison
0CCF C9 57                CMP   #87
0CD1 D0 03                BNE   DEGH
0CD3 4C 54 0D             JMP   PENALTY
; 90=score
0CD6 C9 5A      DEGH      CMP   #90
0CD8 F0 07                BEQ   HITSOME
; 106=sidebar
0CDA C9 6A                CMP   #106
0CDC F0 42                BEQ   REVLR
; only other possibility now is
; space, so just move egg
0CDE 4C E7 0C             JMP   EGPUT
;
0CE1 20 15 0D   HITSOME   JSR   REVUD
0CE4 20 83 0E             JSR   ONEMORE
;
0CE7 A0 00      EGPUT     LDY   #0
0CE9 A9 2A                LDA   #42
0CEB 91 FB                STA   (EGG),Y
0CED 18                   CLC
0CEE A5 FB                LDA   EGG
0CF0 69 00                ADC   #0
0CF2 85 FD                STA   EGCOL
0CF4 A5 FC                LDA   EGG+1
0CF6 69 D4                ADC   #212
0CF8 85 FE                STA   EGCOL+1
0CFA A9 01                LDA   #1
0CFC 91 FD                STA   (EGCOL),Y
0CFE 60                   RTS
```

```
;
0CFF A9 01      EGTOP       LDA    #1
0D01 8D 51 03               STA    EGUD
0D04 8D 63 03               STA    PFLAG
0D07 A9 00                  LDA    #0
0D09 8D 46 03               STA    BIRDSPEED
0D0C 60                     RTS
;
0D0D A9 00      EGBOT       LDA    #0
0D0F 8D 51 03               STA    EGUD
0D12 4C 39 0D               JMP    BOUNCE
; reverse egg up/down dir'n
0D15 AD 51 03   REVUD       LDA    EGUD
0D18 49 01                  EOR    #1
0D1A 8D 51 03               STA    EGUD
0D1D 4C 39 0D               JMP    BOUNCE
; reverse egg left/right dir'n
0D20            REVLR       =      *
0D20 AD 53 03               LDA    EGLR
0D23 49 01                  EOR    #1
0D25 8D 53 03               STA    EGLR
0D28 4C 39 0D               JMP    BOUNCE
;
0D2B            UPONE       =      *
0D2B 38                     SEC
0D2C A5 FB                  LDA    EGG
0D2E E9 28                  SBC    #40
0D30 85 FB                  STA    EGG
0D32 A5 FC                  LDA    EGG+1
0D34 E9 00                  SBC    #0
0D36 85 FC                  STA    EGG+1
0D38 60                     RTS
;
;----- bounce -------------------
;
; sound effects and sidebar
; sparkle for every egg collision
0D39            BOUNCE      =      *
0D39 A9 40                  LDA    #64
0D3B 8D 04 D4               STA    54276
0D3E 20 CA 10               JSR    SPARKLE
0D41 A9 B4                  LDA    #180
0D43 8D 5E 03               STA    FCUT
0D46 A9 01                  LDA    #1
0D48 8D 5D 03               STA    FCNT
0DD4B 8D 5F 03               STA    FDIR
0D4E A9 41                  LDA    #65
0D50 8D 04 D4               STA    54276
0D53 60                     RTS
;
;-------- penalty ---------------
;
; comes when egg hits poison
; globule down below. ow!
0D54 20 3C 0A   PENALTY     JSR    BOAT
0D57 A9 03                  LDA    #3
0D59 8D 46 03               STA    BIRDSPEED
0D5C A9 02                  LDA    #2
0D5E 8D 20 D0               STA    $D020
0D61 8D 21 D0               STA    $D021
0D64 20 ED 0B               JSR    EGERASE
; move bird down, deposit sidebar
; and move back to top again
0D67            DESCEND     =      *
; sparkle while you work.....
0D67 20 CA 10               JSR    SPARKLE
0D6A EE 01 D0               INC    53249
0D6D EE 01 D0               INC    53249
0D70 AD 01 D0               LDA    53249
0D73 8D 01 D4               STA    54273
```

```
0D76 C9 E9                  CMP    #233
0D78 90 02                  BCC    DNMOR
0D7A B0 05                  BCS    BONSCR
0D7C AD 55 03   DNMOR       LDA    COLSAV
0D7F F0 E6                  BEQ    DESCEND
0D81 20 39 0D   BONSCR      JSR    BOUNCE
0D84 20 34 0C   BOTSCR      JSR    GETRAND
0D87 C9 20                  CMP    #32
0D89 B0 F9                  BCS    BOTSCR
0D8B 8D 4F 03               STA    TEMP
0D8E A9 AC                  LDA    #<1708
0D90 85 B0                  STA    $B0
0D92 A9 06                  LDA    #>1708
0D94 85 B1                  STA    $B1
0D96 A0 00                  LDY    #0
0D98 18                     CLC
0D99 A5 B0                  LDA    $B0
0D9B 6D 4F 03               ADC    TEMP
0D9E 85 B0                  STA    $B0
0DA0 A5 B1                  LDA    $B1
0DA2 69 00                  ADC    #0
0DA4 85 B1                  STA    $B1
0DA6 B1 B0      CKIT        LDA    ($B0),Y
0DA8 C9 6A                  CMP    #106
0DAA F0 D8                  BEQ    BOTSCR
0DAC C8                     INY
0DAD C0 05                  CPY    #5
0DAF D0 F5                  BNE    CKIT
;--- deposit a new sidebar
0DB1 A0 02                  LDY    #2
0DB3 A9 6A                  LDA    #106
0DB5 91 B0                  STA    ($B0),Y
0DB7 18                     CLC
0DB8 A5 B0                  LDA    $B0
0DBA 69 00                  ADC    #0
0DBC 85 B0                  STA    $B0
0DBE A5 B1                  LDA    $B1
0DC0 69 D4                  ADC    #212
0DC2 85 B1                  STA    $B1
0DC4 A9 03                  LDA    #3
0DC6 91 B0                  STA    ($B0),Y
0DC8 A0 00                  LDY    #0
0DCA A9 03                  LDA    #3
0DCC 91 B0                  STA    ($B0),Y
0DCE A0 00                  LDY    #0
;--- move bird back up
0DD0 AD 01 D0   BAKUP       LDA    53249
0DD3 C9 3C                  CMP    #60
0DD5 90 12                  BCC    DAKUP
0DD7 20 CA 10               JSR    SPARKLE
0DDA CE 01 D0               DEC    53249
0DDD CE 01 D0               DEC    53249
0DE0 AD 01 D0               LDA    53249
0DE3 8D 01 D4               STA    54273
0DE6 4C D0 0D               JMP    BAKUP
0DE9 AD 1F D0   DAKUP       LDA    53279
0DEC A9 00                  LDA    #0
0DEE 8D 20 D0               STA    $D020
0DF1 8D 21 D0               STA    $D021
0DF4 8D 55 03               STA    COLSAV
0DF7 20 39 0D               JSR    BOUNCE
; cheesy way to exit, but it
; works just fine
0DFA 68                     PLA
0DFB 68                     PLA
0DFC 20 3C 0A               JSR    BOAT
0DFF 4C 35 09               JMP    NOTAK
;
;----- score -------------------
;
```

```
; increase score and display new
; scores above the playfield
0E02                UPSCORE     =      *
0E02  18                        CLC
0E03  AD 57 03                  LDA    SCORE
0E06  69 0A                     ADC    #10
0E08  8D 57 03                  STA    SCORE
0E0B  AD 58 03                  LDA    SCORE+1
0E0E  69 00                     ADC    #0
0E10  8D 58 03                  STA    SCORE+1
0E13  38            SC2         SEC
0E14  AD 57 03                  LDA    SCORE
0E17  ED 54 14                  SBC    HIGH
0E1A  AD 58 03                  LDA    SCORE+1
0E1D  ED 55 14                  SBC    HIGH+1
0E20  90 0C                     BCC    DOSCR
0E22  AD 57 03                  LDA    SCORE
0E25  8D 54 14                  STA    HIGH
0E28  AD 58 03                  LDA    SCORE+1
0E2B  8D 55 14                  STA    HIGH+1
0E2E                DOSCR       =      *
0E2E  A9 13                     LDA    #19
0E30  20 D2 FF                  JSR    $FFD2
0E33  A2 0B                     LDX    #11
0E35  A9 20                     LDA    #32
0E37  9D 00 04      BLANK       STA    1024,X
0E3A  CA                        DEX
0E3B  10 FA                     BPL    BLANK
0E3D  A9 1D                     LDA    #29
0E3F  20 D2 FF                  JSR    $FFD2
0E42  A9 1D                     LDA    #29
0E44  20 D2 FF                  JSR    $FFD2
0E47  AE 57 03                  LDX    SCORE
0E4A  AD 58 03                  LDA    SCORE+1
0E4D  20 CD BD                  JSR    $BDCD
0E50  A9 20                     LDA    #32
0E52  20 D2 FF                  JSR    $FFD2
0E55  A9 E8                     LDA    #<HIMSG
0E57  A0 13                     LDY    #>HIMSG
0E59  20 1E AB                  JSR    $AB1E
0E5C  AE 54 14                  LDX    HIGH
0E5F  AD 55 14                  LDA    HIGH+1
0E62  20 CD BD                  JSR    $BDCD
0E65  A9 20                     LDA    #32
0E67  20 D2 FF                  JSR    $FFD2
0E6A  A9 18         BALOT       LDA    #<BLMSG
0E6C  A0 14                     LDY    #>BLMSG
0E6E  20 1E AB                  JSR    $AB1E
0E71  AE 62 03                  LDX    BALLS
0E74  A9 00                     LDA    #0
0E76  20 CD BD                  JSR    $BDCD
0E79  A9 20                     LDA    #32
0E7B  20 D2 FF                  JSR    $FFD2
0E7E  A9 20                     LDA    #32
0E80  4C D2 FF                  JMP    $FFD2
;
0E83                ONEMORE     =      *
0E83  18                        CLC
0E84  AD 57 03                  LDA    SCORE
0E87  69 01                     ADC    #1
0E89  8D 57 03                  STA    SCORE
0E8C  AD 58 03                  LDA    SCORE+1
0E8F  69 00                     ADC    #0
0E91  8D 58 03                  STA    SCORE+1
0E94  4C 13 0E                  JMP    SC2
;
;------ done yet ---------------
;
; make sure everything's cleaned
; out of the playfield
```

```
0E97                DONCHK      =      *
0E97  A9 00                     LDA    #0
0E99  8D 5B 03                  STA    DNFLAG
0E9C  85 AA                     STA    $AA
0E9E  A9 04                     LDA    #4
0EA0  85 AB                     STA    $AB
0EA2  A0 00                     LDY    #0
0EA4  B1 AA         ALL         LDA    ($AA),Y
0EA6  C9 5A                     CMP    #90
0EA8  F0 16                     BEQ    YETMOR
0EAA  C9 57                     CMP    #87
0EAC  F0 12                     BEQ    YETMOR
0EAE  E6 AA                     INC    $AA
0EB0  D0 02                     BNE    NOXAB
0EB2  E6 AB                     INC    $AB
0EB4  38            NOXAB       SEC
0EB5  A5 AA                     LDA    $AA
0EB7  E9 E6                     SBC    #<2022
0EB9  A5 AB                     LDA    $AB
0EBB  E9 07                     SBC    #>2022
0EBD  90 E5                     BCC    ALL
0EBF  60                        RTS
;
0EC0  A9 01         YETMOR      LDA    #1
0EC2  8D 5B 03                  STA    DNFLAG
0EC5  60                        RTS
;
;------ startup ----------------
;
; initialize for a new game
;
0EC6                STARTUP     =      *
0EC6  20 3C 0A                  JSR    BOAT
0EC9  AD 56 14                  LDA    BALLY
0ECC  8D 62 03                  STA    BALLS
0ECF  A9 03                     LDA    #3
0ED1  8D 64 03                  STA    SLFLAG
0ED4  A9 93                     LDA    #$93
0ED6  20 D2 FF                  JSR    $FFD2
0ED9  A9 00                     LDA    #0
0EDB  8D 15 D0                  STA    53269
0EDE  20 69 0A                  JSR    DRAWSCRN
0EE1  20 2E 0E                  JSR    DOSCR
0EE4  A9 01                     LDA    #1
0EE6  8D 1D D0                  STA    53248+29
0EE9  A9 3D                     LDA    #61
0EEB  8D 01 D0                  STA    53249
0EEE  A9 F3                     LDA    #243
0EF0  8D F8 07                  STA    2040
0EF3  8D 3C 03                  STA    SHAPE
0EF6  A9 06                     LDA    #6
0EF8  8D 3E 03                  STA    DELSPEED
0EFB  A9 02                     LDA    #2
0EFD  8D 46 03                  STA    BIRDSPEED
0F00  A9 01                     LDA    #1
0F02  8D 51 03                  STA    EGUD
0F05  8D 53 03                  STA    EGLR
0F08  A9 50                     LDA    #80
0F0A  8D 3F 03                  STA    XPOS
0F0D  8D 00 D0                  STA    53248
0F10  A9 00                     LDA    #0
0F12  8D 40 03                  STA    XPOS+1
0F15  8D 10 D0                  STA    53264
0F18  8D 41 03                  STA    LEX
;--- draw startup message
0F1B  A2 08                     LDX    #8
0F1D  BD FD 13      DRMES       LDA    MSG1,X
0F20  9D 4D 05                  STA    1357,X
0F23  A9 03                     LDA    #3
```

```
0F25 9D 4D D9                 STA    1357+54272,X
0F28 BD 06 14                 LDA    MSG2,X
0F2B 9D 75 05                 STA    1397,X
0F2E A9 03                    LDA    #3
0F30 9D 75 D9                 STA    1397+54272,X
0F33 BD 0F 14                 LDA    MSG3,X
0F36 9D 9D 05                 STA    1437,X
0F39 A9 03                    LDA    #3
0F3B 9D 9D D9                 STA    1437+54272,X
0F3E CA                       DEX
0F3F 10 DC                    BPL    DRMES
;--- default bounds for whirlybird
0F41 A9 22                    LDA    #34
0F43 8D 44 03                 STA    LLIMIT
0F46 A9 00                    LDA    #0
0F48 8D 45 03                 STA    LLIMIT+1
0F4B A9 20                    LDA    #32
0F4D 8D 42 03                 STA    RLIMIT
0F50 A9 01                    LDA    #1
0F52 8D 43 03                 STA    RLIMIT+1
0F55 8D 15 D0                 STA    53269
;--- adjust number of eggs
0F58 20 CA 10   KEYIN         JSR    SPARKLE
0F5B AD 00 DC                 LDA    56320
0F5E 29 0F                    AND    #15
0F60 C9 0E                    CMP    #14
0F62 F0 07                    BEQ    BALUP
0F64 C9 0D                    CMP    #13
0F66 F0 0C                    BEQ    BALDN
0F68 4C 7D 0F                 JMP    DEYIN
0F6B EE 62 03   BALUP         INC    BALLS
0F6E 20 6A 0E                 JSR    BALOT
0F71 4C 77 0F                 JMP    BEIN
0F74 CE 62 03   BALDN         DEC    BALLS
0F77 20 6A 0E   BEIN          JSR    BALOT
0F7A 20 51 0A                 JSR    SLOWDOWN
;--- poll keyboard for f1, f3, f7
0F7D A5 C5      DEYIN         LDA    197
0F7F C9 04                    CMP    #4
0F81 F0 0B                    BEQ    NORM
0F83 C9 05                    CMP    #5
0F85 F0 14                    BEQ    EXPRT
0F87 C9 03                    CMP    #3
0F89 F0 49                    BEQ    BASIC
0F8B 4C 58 0F                 JMP    KEYIN
;--- whirlybird is expanded
0F8E A9 00      NORM          LDA    #0
0F90 8D 5C 03                 STA    SKILL
0F93 A9 01                    LDA    #1
0F95 8D 1D D0                 STA    53248+29
0F98 4C B9 0F                 JMP    STDONE
;--- whirlybird is small
0F9B A9 01      EXPRT         LDA    #1
0F9D 8D 5C 03                 STA    SKILL
0FA0 A9 00                    LDA    #0
0FA2 8D 1D D0                 STA    53248+29
;- small bird needs wider bounds -
0FA5 A9 20                    LDA    #32
0FA7 8D 44 03                 STA    LLIMIT
0FAA A9 00                    LDA    #0
0FAC 8D 45 03                 STA    LLIMIT+1
0FAF A9 37                    LDA    #55
0FB1 8D 42 03                 STA    RLIMIT
0FB4 A9 01                    LDA    #1
0FB6 8D 43 03                 STA    RLIMIT+1
;--- erase startup message -------
0FB9 A2 08      STDONE        LDX    #8
0FBB A9 20                    LDA    #32
0FBD 9D 4D 05   ERMES         STA    1357,X
0FC0 9D 75 05                 STA    1397,X
```

```
0FC3 9D 9D 05                 STA    1437,X
0FC6 CA                       DEX
0FC7 10 F4                    BPL    ERMES
0FC9 A9 00                    LDA    #0
0FCB 8D 57 03                 STA    SCORE
0FCE 8D 58 03                 STA    SCORE+1
0FD1 4C 2E 0E                 JMP    DOSCR
;--- exit to basic if f7 pressed -
; execute a brk to restore vectors,
; clear screen, etc.
0FD4 A9 00      BASIC         LDA    #0
0FD6 85 C6                    STA    198
0FD8 4C C6 00                 JMP    198
;
;---- interrupt handler ---------
;
; operates as a background process
; during irq intervals. moves
; bird, rotates its shape and
; changes filtering for sounds
;
0FDB            INTERRUPT     =      *
0FDB AD 69 03                 LDA    GOFLAG
0FDE F0 03                    BEQ    DINT
0FE0 4C 31 EA                 JMP    $EA31
0FE3 EE 5D 03   DINT          INC    FCNT
0FE6 AD 5D 03                 LDA    FCNT
0FE9 C9 01                    CMP    #1
0FEB 90 33                    BCC    FDO
0FED A9 00                    LDA    #0
0FEF 8D 5D 03                 STA    FCNT
0FF2 AD 5F 03                 LDA    FDIR
0FF5 D0 17                    BNE    FDWN
0FF7 EE 5E 03   FUP           INC    FCUT
0FFA EE 5E 03                 INC    FCUT
0FFD AD 5E 03                 LDA    FCUT
1000 C9 FA                    CMP    #250
1002 B0 02                    BCS    FBDN
1004 90 1A                    BCC    FDO
1006 A9 01      FBDN          LDA    #1
1008 8D 5F 03                 STA    FDIR
100B 4C 20 10                 JMP    FDO
100E CE 5E 03   FDWN          DEC    FCUT
1011 CE 5E 03                 DEC    FCUT
1014 AD 5E 03                 LDA    FCUT
1017 C9 32                    CMP    #50
1019 B0 05                    BCS    FDO
101B A9 00                    LDA    #0
101D 8D 5F 03                 STA    FDIR
;--- current filter value
1020 AD 5E 03   FDO           LDA    FCUT
1023 8D 16 D4                 STA    54294
1026 AD 1F D0                 LDA    53279
1029 8D 55 03                 STA    COLSAV
102C AD 46 03                 LDA    BIRDSPEED
102F 4A                       LSR
1030 8D 3E 03                 STA    DELSPEED
1033 AD 3D 03                 LDA    DELAY
1036 CD 3E 03                 CMP    DELSPEED
1039 B0 06                    BCS    DLDONE
103B EE 3D 03                 INC    DELAY
103E 4C 60 10                 JMP    BACKFORTH
;--- rotate bird shape
1041 A9 00      DLDONE        LDA    #0
1043 8D 3D 03                 STA    DELAY
1046 AD 3C 03                 LDA    SHAPE
1049 C9 F9                    CMP    #249
104B B0 0C                    BCS    BKSHAPE
104D            UPSHAPE       =      *
104D EE 3C 03                 INC    SHAPE
```

```
1050 AD 3C 03               LDA   SHAPE
1053 8D F8 07               STA   SPRSHAPE
1056 4C 60 10               JMP   BACKFORTH
1059 A9 F0      BKSHAPE     LDA   #240
105B 8D 3C 03               STA   SHAPE
105E D0 ED                  BNE   UPSHAPE
;
;--- move bird right/left
;
1060            BACKFORTH   =     *
1060 AD 41 03               LDA   LEX
1063 F0 2B                  BEQ   LEFT
1065            RIGHT       =     *
1065 38                     SEC
1066 AD 3F 03               LDA   XPOS
1069 ED 42 03               SBC   RLIMIT
106C AD 40 03               LDA   XPOS+1
106F ED 43 03               SBC   RLIMIT+1
1072 B0 15                  BCS   BACKLEFT
1074            RITEOK      =     *
1074 18                     CLC
1075 AD 3F 03               LDA   XPOS
1078 6D 46 03               ADC   BIRDSPEED
107B 8D 3F 03               STA   XPOS
107E AD 40 03               LDA   XPOS+1
1081 69 00                  ADC   #0
1083 8D 40 03               STA   XPOS+1
1086 4C BB 10               JMP   GOMOVE
1089            BACKLEFT    =     *
1089 A9 00                  LDA   #0
108B 8D 41 03               STA   LEX
108E F0 2B                  BEQ   GOMOVE
1090            LEFT        =     *
1090 38                     SEC
1091 AD 44 03               LDA   LLIMIT
1094 ED 3F 03               SBC   XPOS
1097 AD 45 03               LDA   LLIMIT+1
109A ED 40 03               SBC   XPOS+1
109D F0 17                  BEQ   BACKRIGHT
109F B0 15                  BCS   BACKRIGHT
10A1            LEFTOK      =     *
10A1 38                     SEC
10A2 AD 3F 03               LDA   XPOS
10A5 ED 46 03               SBC   BIRDSPEED
10A8 8D 3F 03               STA   XPOS
10AB AD 40 03               LDA   XPOS+1
10AE E9 00                  SBC   #0
10B0 8D 40 03               STA   XPOS+1
10B3 4C BB 10               JMP   GOMOVE
10B6            BACKRIGHT   =     *
10B6 A9 01                  LDA   #1
10B8 8D 41 03               STA   LEX
10BB            GOMOVE      =     *
10BB AD 3F 03               LDA   XPOS
10BE 8D 00 D0               STA   53248
10C1 AD 40 03               LDA   XPOS+1
10C4 8D 10 D0               STA   53264
; must terminate with jmp to
; normal irq entry point
10C7
     4C
     31
     EA
     NOMOVE
     JMP
     $EA31
;
;------ sparkle -----------------
;
10CA            SPARKLE     =     *
```

```
10CA A2 17                  LDX   #23
10CC A0 00                  LDY   #0
10CE A9 50                  LDA   #<55296+80
10D0 85 B0                  STA   $B0
10D2 A9 D8                  LDA   #>55296+80
10D4 85 B1                  STA   $B1
10D6 A9 77                  LDA   #<55335+80
10D8 85 B4                  STA   $B4
10DA A9 D8                  LDA   #>55335+80
10DC 85 B5                  STA   $B5
10DE            SPKL        =     *
10DE AD 1B D4               LDA   $D41B
10E1 29 0F                  AND   #15
10E3 C9 00                  CMP   #0
10E5 F0 F7                  BEQ   SPKL
10E7 C9 02                  CMP   #2
10E9 F0 F3                  BEQ   SPKL
10EB C9 0B                  CMP   #11
10ED F0 EF                  BEQ   SPKL
10EF 91 B0                  STA   ($B0),Y
10F1 91 B4                  STA   ($B4),Y
10F3 18                     CLC
10F4 A5 B0                  LDA   $B0
10F6 69 28                  ADC   #40
10F8 85 B0                  STA   $B0
10FA A5 B1                  LDA   $B1
10FC 69 00                  ADC   #0
10FE 85 B1                  STA   $B1
1100 18                     CLC
1101 A5 B4                  LDA   $B4
1103 69 28                  ADC   #40
1105 85 B4                  STA   $B4
1107 A5 B5                  LDA   $B5
1109 69 00                  ADC   #0
110B 85 B5                  STA   $B5
110D CA                     DEX
110E 10 CE                  BPL   SPKL
1110 60                     RTS
;
;--- sprite shape data -----------
;
1111            SPRDATA     =     *
1111 00 00 00               .BYTE 0,0,0,0,0,0,0,0,0,0,0
111C 00 00 00               .BYTE 0,0,0,0,0,0,0,0,24,0,
                                  8
1127 18 10 1E               .BYTE 24,16,30,24,120,63,
                                  153,252,103,255,23
                                  0
1132 40 FF 02               .BYTE 64,255,2,128,60,1,0,
                                  60,0,0,24
113D 00 00 18               .BYTE 0,0,24,0,1,255,128,0
                                  ,0,0,0
1148 00 00 00               .BYTE 0,0,0,0,0,0,0,0,0,0,0
1153 00 00 00               .BYTE 0,0,0,0,0,0,0,0,0,0,0
115E 00 00 00               .BYTE 0,0,0,0,0,0,28,0,4,24
                                  ,32
1169 0E 18 70               .BYTE 14,24,112,31,153,24
                                  8,55,255,236,96,255
1174 06 40 3C               .BYTE 6,64,60,2,0,60,0,0,2
                                  4,0,0
117F 18 00 00               .BYTE 24,0,0,255,0,0,0,0,0,
                                  0,0
118A 00 00 00               .BYTE 0,0,0,0,0,0,0,0,0,0,0
1195 00 00 00               .BYTE 0,0,0,0,0,0,0,0,0,0,0
11A0 00 00 00               .BYTE 0,0,0,0,28,0,2,24,0,7
                                  ,24
11AB E0 0F D9               .BYTE 224,15,217,240,31,2
                                  55,248,56,255,28,48
11B6 3C 0C 00               .BYTE 60,12,0,60,0,0,24,0,
                                  0,24,0
```

```
11C1 00 7E 00              .BYTE  0,126,0,0,0,0,0,0,0
                                  ,0
11CC 00 00 00              .BYTE  0,0,0,0,0,0,0,0,0,0,0
11D7 00 00 00              .BYTE  0,0,0,0,0,0,0,0,0,0,0
11E2 00 00 1E              .BYTE  0,0,30,0,3,24,0,7,15
                                  2,128,15
11ED DB E0 0F              .BYTE  219,224,15,255,240,
                                  28,126,56,24,60,24
11F8 00 3C 00              .BYTE  0,60,0,0,24,0,0,24,0,
                                  0,60
1203 00 00 00              .BYTE  0,0,0,0,0,0,0,0,0,0,0
120E 00 00 00              .BYTE  0,0,0,0,0,0,0,0,0,0,0
1219 00 00 00              .BYTE  0,0,0,0,0,0,0,0,0,0,0
1224 1E 00 00              .BYTE  30,0,0,92,0,0,249,0,
                                  1,255,128
122F 03 FF C0              .BYTE  3,255,192,7,126,224
                                  ,6,60,224,0,60
123A 00 00 18              .BYTE  0,0,24,0,0,24,0,0,8,0
                                  ,0
1245 00 00 00              .BYTE  0,0,0,0,0,0,0,0,0,0,0
1250 00 00 00              .BYTE  0,0,0,0,0,0,0,0,0,0,0
125B 00 00 00              .BYTE  0,0,0,0,0,0,0,0,0,24,
                                  0
1266 00 18 00              .BYTE  0,24,0,0,60,0,0,60,0,
                                  0,126
1271 00 00 7E              .BYTE  0,0,126,0,0,126,0,0,
                                  60,0,0
127C 18 00 00              .BYTE  24,0,0,24,0,0,24,0,0,
                                  0,0
1287 00 00 00              .BYTE  0,0,0,0,0,0,0,0,0,0,0
1292 00 00 00              .BYTE  0,0,0,0,0,0,0,0,0,0,0
129D 00 00 00              .BYTE  0,0,0,0,0,0,0,120,0,0
                                  ,58
12A8 00 00 9F              .BYTE  0,0,159,0,1,255,128,
                                  3,255,192,7
12B3 7E E0 07              .BYTE  126,224,7,60,96,0,6
                                  0,0,0,24,0
12BE 00 18 00              .BYTE  0,24,0,0,16,0,0,0,0,0
                                  ,0
12C9 00 00 00              .BYTE  0,0,0,0,0,0,0,0,0,0,0
12D4 00 00 00              .BYTE  0,0,0,0,0,0,0,0,0,0,0
12DF 00 00 00              .BYTE  0,0,0,0,0,120,0,0,24,
                                  192,1
12EA 19 E0 07              .BYTE  25,224,7,219,240,15
                                  ,255,240,28,126,56
12F5 18 3C 18              .BYTE  24,60,24,0,60,0,0,24
                                  ,0,0,24
1300 00 00 3C              .BYTE  0,0,60,0,0,0,0,0,0,0,
                                  0
130B 00 00 00              .BYTE  0,0,0,0,0,0,0,0,0,0,0
1316 00 00 00              .BYTE  0,0,0,0,0,0,0,0,0,0,0
1321 00 00 00              .BYTE  0,0,0,56,0,0,24,64,7,
                                  24,224
132C 0F 9B F0              .BYTE  15,155,240,31,255,2
                                  48,56,255,28,48,60
1337 0C 00 3C              .BYTE  12,0,60,0,0,24,0,0,2
                                  4,0,0
1342 7E 00 00              .BYTE  126,0,0,0,0,0,0,0,0,0
                                  ,0
134D 00 00 00              .BYTE  0,0,0,0,0,0,0,0,0,0,0
1358 00 00 00              .BYTE  0,0,0,0,0,0,0,0,0,0,0
1363 00 38 00              .BYTE  0,56,0,4,24,32,14,24
                                  ,112,31,153
136E F8 37 FF              .BYTE  248,55,255,236,96,2
                                  55,6,64,60,2,0
1379 3C 00 00              .BYTE  60,0,0,24,0,0,24,0,0,
                                  255,0
1384 00 00 00              .BYTE  0,0,0,0,0,0,0,0,0,0,0
138F 00 00 00              .BYTE  0,0,0,0,0,0,0,0,0,0,0
139A 00 00 00              .BYTE  0,0,0,0,0,0,0,0,0,0,
                                  24
```

```
13A5 00 08 18              .BYTE  0,8,24,16,30,24,120,
                                  63,153,252,103
13B0 FF E6 40              .BYTE  255,230,64,255,2,12
                                  8,60,1,0,60,0
13BB 00 18 00              .BYTE  0,24,0,0,24,0,1,255,
                                  128,0,0
13C6 00 00 00              .BYTE  0,0,0,0,0,0,0,0,0,0
13D0          ENDATA       =      *
;
;------ character data ---------
;
13D0          CRDATA       =      *
13D0 FF FF C3              .BYTE  255,255,195,195,19
                                  5,195,255,255
13D8          CRDA1        =      *
13D8 FF 3C 18              .BYTE  255,60,24,24,24,24,
                                  60,255
13E0          CRDA2        =      *
13E0 00 3C 7E              .BYTE  0,60,126,255,255,12
                                  6,60,0
;
;----- variables & etc ----------
;
13E8 13 1D 1D HIMSG        .BYT   19,29,29,29,29,29,2
                                  9
13EF 1D 1D 1D              .BYT   29,29,29,29,29,29,2
                                  9
13F6 48 49 47              .ASC   "HIGH: "
13FC 00                    .BYT   0
13FD 06 31 20 MSG1         .BYT   6,49,32,14,15,18,13,
                                  1,12
1406 06 33 20 MSG2         .BYT   6,51,32,5,24,16,5,18
                                  ,20
140F 06 37 20 MSG3         .BYT   6,55,32,17,21,9,20,3
                                  2,32
;
1418          BLMSG        =      *
1418 13 1D 1D              .BYT   19,29,29,29,29,29,2
                                  9,29,29,29,29,29
1424 1D 1D 1D              .BYT   29,29,29,29,29,29,2
                                  9,29,29,29,29
142F 1D 1D 1D              .BYT   29,29,29,29,29
1434 45 47 47              .ASC   "EGGS: "
143A 00                    .BYT   0
;
;--- SID chip values -------------
;
143B          SID          =      *
143B 0B 07 05              .BYT   11,7,5,200,64,10,0
1442 9B 03 05              .BYT   155,3,5,200,20,15,2
                                  40
1449 FF FF 00              .BYT   255,255,0,0,128,0,0
1450 00 64 F3              .BYT   0,100,243,15+16
                                  +32
;
;--- assorted variables ----------
;
; high score
1454 00 00    HIGH         .BYT   0,0
; default num of eggs
1456 0A       BALLY        .BYT   10
; sprite register
1457          SPRSHAPE     =      2040
; sprite shape
1457          SHAPE        =      828
1457          DELAY        =      829
1457          DELSPEED     =      830
; bird x position
1457          XPOS         =      831
```

```
; bird l/r flag
1457          LEX         =    833
; r bird boundary
1457          RLIMIT      =    834
; l bird boundary
1457          LLIMIT      =    836
1457          BIRDSPEED   =    838
1457          COUNT       =    839
; for drawing screen
1457          DLIMIT      =    841
; ditto
1457          CHAR        =    843
; ditto
1457          COL         =    844
1457          TEMP        =    847
; egg up/down flag
1457          EGUD        =    849
; egg l/r flag
1457          EGLR        =    851
; collision info
1457          COLSAV      =    853
; collision flag
1457          COLFLAG     =    854
; current score
1457          SCORE       =    855
; done/not done
1457          DNFLAG      =    859
; skill level
1457          SKILL       =    860
; filter counter
1457          FCNT        =    861
; filter cutoff
1457          FCUT        =    862
; filter u/d flag
1457          FDIR        =    863
1457          FCN1        =    864
1457          RING        =    865
; num of eggs left
1457          BALLS       =    866
1457          PFLAG       =    867
1457          SLFLAG      =    868
1457          BNOTE       =    869
1457          GFLAG       =    872
; pause flag
1457          GOFLAG      =    873
```

Miami Ice Source Code

Source code by Kevin Mykytyn

```
0801                            *=     2049
;
; miami ice
;
0801            SCR             =      5
0801            STARTX          =      38
0801            STARTY          =      60
0801            WALLS           =      4
0801            MAX             =      70
0801            MIN             =      185
0801            TEMP            =      2
0801            TEMP0           =      3
0801            FREE0           =      251
0801            TIMER           =      248
0801            TICK            =      247
0801            SCORE           =      195
0801            LIVES           =      189
0801            LEVEL           =      180
;
; BASIC line to start program
;
0801 0C 08 0A                   .BYT   12,8,10,0,158
0806 20 32 30                   .ASC   " 2062"
080B 00 00 00                   .BYT   0,0,0
;
080E 20 E0 0E                   JSR    SETSPRITE
0811 20 BC 0D                   JSR    RASTSET
0814 A9 00                      LDA    #0
0816 A0 18                      LDY    #24
0818 B9 A2 08   D               LDA    SID,Y
081B 99 00 D4                   STA    54272,Y
081E 88                         DEY
081F 10 F7                      BPL    D
0821 20 C4 08   NEWGAME         JSR    STARTGAME
0824 20 75 0A   STARTROUN       JSR    DRAWSCREEN
0827 AD 01 DC   LETGO           LDA    $DC01
082A 29 10                      AND    #16
082C F0 F9                      BEQ    LETGO
082E A2 00                      LDX    #0
0830 A0 10                      LDY    #16
0832 18                         CLC
0833 20 F0 FF                   JSR    $FFF0
0836 A9 E2                      LDA    #<MESS4
0838 A0 0B                      LDY    #>MESS4
083A 20 1E AB                   JSR    $AB1E
083D A6 B4                      LDX    LEVEL
083F E8                         INX
0840 A9 00                      LDA    #0
0842 20 CD BD                   JSR    $BDCD
0845 A9 C8                      LDA    #200
0847 85 F8                      STA    TIMER
0849 A9 00                      LDA    #0
084B 85 F9                      STA    TIMER+1
084D 20 D6 0D   RESET           JSR    INITROUND
0850 20 52 0A                   JSR    SCROUT
0853 20 95 0E   WAITBUT         JSR    JOYSTICK
0856 A2 64                      LDX    #100
0858 88         WTER            DEY
0859 D0 FD                      BNE    WTER
085B CA                         DEX
```

```
085C D0 FA                      BNE    WTER
085E AD 1F D0                   LDA    53279
0861 AD 1E D0                   LDA    53278
0864 AD 01 DC                   LDA    $DC01
0867 29 10                      AND    #16
0869 D0 E8                      BNE    WAITBUT
086B A9 40                      LDA    #64
086D 8D 04 D4                   STA    54276
0870 A9 41                      LDA    #65
0872 8D 04 D4                   STA    54276
0875            MAIN            =      *
0875 20 1D 09                   JSR    TOCK
0878 20 48 09                   JSR    CRASH
087B CE 21 13                   DEC    JOYL
087E D0 09                      BNE    NOJOY
0880 AD 22 13                   LDA    JOYH
0883 8D 21 13                   STA    JOYL
0886 20 95 0E                   JSR    JOYSTICK
0889            NOJOY           =      *
0889 CE 25 13                   DEC    MOVL
088C D0 0C                      BNE    NOMOVE
088E AD 26 13                   LDA    MOVH
0891 8D 25 13                   STA    MOVL
0894 20 16 0E                   JSR    MOVCAR
0897 20 4F 0D                   JSR    CALC
089A            NOMOVE          =      *
089A A0 00                      LDY    #0
089C 88         WT              DEY
089D D0 FD                      BNE    WT
089F 4C 75 08                   JMP    MAIN
08A2 00 05 00   SID             .BYT   0,5,0,1,0,25,240
08A9 00 1E 00                   .BYT   0,30,0,0,0,137,0
08B0 00 0A 00                   .BYT   0,10,0,0,0,43,0
08B7 00 05 F1                   .BYT   0,5,241,79
;
; output a number
;
08BB            RBDCD           =      *
08BB 20 CD BD                   JSR    $BDCD
08BE A9 20                      LDA    #" "
08C0 20 D2 FF                   JSR    $FFD2
08C3 60                         RTS
;
; start game
;
08C4            STARTGAME       =      *
08C4 A9 00                      LDA    #0
08C6 85 C3                      STA    SCORE
08C8 85 C4                      STA    SCORE+1
08CA A9 00                      LDA    #0
08CC 85 B4                      STA    LEVEL
08CE A9 03                      LDA    #3
08D0 85 BD                      STA    LIVES
08D2 A9 93                      LDA    #"{CLR}"
08D4 20 D2 FF                   JSR    $FFD2
08D7 A2 03                      LDX    #3
08D9 8E 21 D0                   STX    53281
08DC E8                         INX
08DD 8E 20 D0                   STX    53280
08E0 A2 07                      LDX    #7
```

```
08E2 A0 0B              LDY   #11
08E4 18                 CLC
08E5 20 F0 FF           JSR   $FFF0
08E8 A9 EC              LDA   #<MESS5
08EA A0 0B              LDY   #>MESS5
08EC 20 1E AB           JSR   $AB1E
08EF A5 B4     VP       LDA   LEVEL
08F1 18                 CLC
08F2 69 31              ADC   #49
08F4 8D A7 05           STA   1447
08F7 A9 0A              LDA   #10
08F9 20 45 0A           JSR   BIGDELAY
08FC. AD 01 DC          LDA   $DC01
08FF 4A                 LSR
0900 B0 0A              BCS   NUP
0902 A5 B4              LDA   LEVEL
0904 C9 06              CMP   #6
0906 F0 E7              BEQ   VP
0908 E6 B4              INC   LEVEL
090A 10 E3              BPL   VP
090C 4A        NUP      LSR
090D B0 08              BCS   NDOWN
090F A5 B4              LDA   LEVEL
0911 F0 DC              BEQ   VP
0913 C6 B4              DEC   LEVEL
0915 10 D8              BPL   VP
0917 4A        NDOWN    LSR
0918 4A                 LSR
0919 4A                 LSR
091A B0 D3              BCS   VP
091C 60                 RTS
;
; update timer
;
091D           TOCK     =     *
091D C6 F7              DEC   TICK
091F D0 26              BNE   NOTDOWN
0921 A9 C8              LDA   #200
0923 85 F7              STA   TICK
0925 A5 F8              LDA   TIMER
0927 05 F9              ORA   TIMER+1
0929 F0 1C              BEQ   NOTDOWN
092B A5 F8              LDA   TIMER
092D 38                 SEC
092E E9 01              SBC   #1
0930 85 F8              STA   TIMER
0932 A5 F9              LDA   TIMER+1
0934 E9 00              SBC   #0
0936 85 F9              STA   TIMER+1
0938 A2 18              LDX   #24
093A A0 07              LDY   #7
093C 18                 CLC
093D 20 F0 FF           JSR   $FFF0
0940 A6 F8              LDX   TIMER
0942 A5 F9              LDA   TIMER+1
0944 20 BB 08           JSR   RBDCD
0947 60        NOTDOWN  RTS
;
; check on a crash
;
0948           CRASH    =     *
0948 AD 23 13           LDA   VELX
094B 10 05              BPL   CHIT
094D 49 FF              EOR   #255
094F 18                 CLC
0950 69 01              ADC   #1
0952 85 02     CHIT     STA   TEMP
0954 AD 24 13           LDA   VELY
0957 10 05              BPL   CHIT2
0959 49 FF              EOR   #255
```

```
095B 18                 CLC
095C 69 01              ADC   #1
095E 18        CHIT2    CLC
095F 65 02              ADC   TEMP
0961 8D 00 D4           STA   54272
0964 AD 1F D0           LDA   53279
0967 29 08              AND   #8
0969 D0 14              BNE   DIDCRASH
096B AD 1E D0           LDA   53278
096E 85 02              STA   TEMP
0970 29 0A              AND   #10
0972 C9 0A              CMP   #10
0974 F0 09              BEQ   DIDCRASH
0976 A5 02              LDA   TEMP
0978 29 0C              AND   #12
097A C9 0C              CMP   #12
097C F0 3A              BEQ   MADIT
097E 60                 RTS
097F A9 40     DIDCRASH LDA   #64
0981 8D 04 D4           STA   54276
0984 A9 80              LDA   #128
0986 8D 12 D4           STA   54290
0989 A9 81              LDA   #129
098B 8D 12 D4           STA   54290
098E A9 08              LDA   #8
0990 8D 20 13           STA   SPRPTR
0993 A9 05     EXPL     LDA   #5
0995 20 45 0A           JSR   BIGDELAY
0998 AD 20 13           LDA   SPRPTR
099B C9 0D              CMP   #13
099D F0 06              BEQ   FIN
099F EE 20 13           INC   SPRPTR
09A2 4C 93 09           JMP   EXPL
09A5 A9 07     FIN      LDA   #7
09A7 8D 15 D0           STA   53269
09AA A9 64              LDA   #100
09AC 20 45 0A           JSR   BIGDELAY
09AF C6 BD              DEC   LIVES
09B1 F0 5A              BEQ   FINGAME
09B3 68                 PLA
09B4 68                 PLA
09B5 4C 4D 08           JMP   RESET
09B8 A9 40     MADIT    LDA   #64
09BA 8D 04 D4           STA   54276
09BD A9 00              LDA   #0
09BF 85 02              STA   TEMP
09C1 A5 F8     MADE     LDA   TIMER
09C3 05 F9              ORA   TIMER+1
09C5 F0 32              BEQ   FINBONUS
09C7 A5 F8              LDA   TIMER
09C9 38                 SEC
09CA E9 01              SBC   #1
09CC 85 F8              STA   TIMER
09CE A5 F9              LDA   TIMER+1
09D0 E9 00              SBC   #0
09D2 85 F9              STA   TIMER+1
09D4 A5 B4              LDA   LEVEL
09D6 85 03              STA   TEMP0
09D8           REPEAT   =     *
09D8 E6 C3              INC   SCORE
09DA D0 02              BNE   NOTH
09DC E6 C4              INC   SCORE+1
09DE 20 52 0A  NOTH     JSR   SCROUT
09E1 E6 02              INC   TEMP
09E3 A5 02              LDA   TEMP
09E5 8D 01 D4           STA   54273
09E8 A9 40              LDA   #64
09EA 8D 04 D4           STA   54276
09ED A9 41              LDA   #65
09EF 8D 04 D4           STA   54276
```

```
09F2 C6 03             DEC   TEMP0
09F4 10 E2             BPL   REPEAT
09F6 4C C1 09          JMP   MADE
09F9 A9 40   FINBONUS  LDA   #64
09FB 8D 04 D4          STA   54276
09FE E6 B4             INC   LEVEL
0A00 A9 64             LDA   #100
0A02 20 45 0A          JSR   BIGDELAY
0A05 A9 05             LDA   #5
0A07 8D 01 D4          STA   54273
0A0A 4C 24 08          JMP   STARTROUND
0A0D 20 52 0A FINGAME  JSR   SCROUT
0A10 A9 00             LDA   #0
0A12 8D 15 D0          STA   53269
0A15 A2 0C             LDX   #12
0A17 A0 05             LDY   #5
0A19 18                CLC
0A1A 20 F0 FF          JSR   $FFF0
0A1D A9 9E             LDA   #<MESS2
0A1F A0 0B             LDY   #>MESS2
0A21 20 1E AB          JSR   $AB1E
0A24 A2 0B             LDX   #11
0A26 20 38 0A          JSR   LINOUT
0A29 A2 0D             LDX   #13
0A2B 20 38 0A          JSR   LINOUT
0A2E AD 01 DC WTBU     LDA   $DC01
0A31 29 10             AND   #16
0A33 D0 F9             BNE   WTBU
0A35 4C 21 08          JMP   NEWGAME
0A38 A0 05    LINOUT   LDY   #5
0A3A 18                CLC
0A3B 20 F0 FF          JSR   $FFF0
0A3E A9 C1             LDA   #<MESS3
0A40 A0 0B             LDY   #>MESS3
0A42 4C 1E AB          JMP   $AB1E
0A45 85 02    BIGDELAY STA   TEMP
0A47 A9 00             LDA   #0
0A49 85 A2             STA   162
0A4B A5 A2    DEL      LDA   162
0A4D C5 02             CMP   TEMP
0A4F D0 FA             BNE   DEL
0A51 60                RTS
;
; output the score
;
0A52 A9 9F    SCROUT   LDA   #"{CYN}"
0A54 20 D2 FF          JSR   $FFD2
0A57 A2 18             LDX   #24
0A59 A0 17             LDY   #23
0A5B 18                CLC
0A5C 20 F0 FF          JSR   $FFF0
0A5F A6 C3             LDX   SCORE
0A61 A5 C4             LDA   SCORE+1
0A63 20 BB 08          JSR   RBDCD
0A66 A2 18             LDX   #24
0A68 A0 26             LDY   #38
0A6A 18                CLC
0A6B 20 F0 FF          JSR   $FFF0
0A6E A6 BD             LDX   LIVES
0A70 A9 00             LDA   #0
0A72 4C CD BD          JMP   $BDCD
;
; draw a screen
;
0A75          DRAWSCREE =    *
0A75 A9 93             LDA   #"{CLR}"
0A77 20 D2 FF          JSR   $FFD2
0A7A A9 01             LDA   #1
0A7C 8D 21 D0          STA   53281
0A7F A9 03             LDA   #3
```

```
0A81 8D 20 D0          STA   53280
0A84 A2 18             LDX   #24
0A86 A0 00             LDY   #0
0A88 18                CLC
0A89 20 F0 FF          JSR   $FFF0
0A8C A9 71             LDA   #<MESS1
0A8E A0 0B             LDY   #>MESS1
0A90 20 1E AB          JSR   $AB1E
0A93 A0 27             LDY   #39
0A95 A9 A0    TOPBOT   LDA   #160
0A97 99 00 04          STA   1024,Y
0A9A 99 98 07          STA   1944,Y
0A9D A9 04             LDA   #WALLS
0A9F 99 00 D8          STA   55296,Y
0AA2 99 98 DB          STA   56216,Y
0AA5 88                DEY
0AA6 10 ED             BPL   TOPBOT
0AA8 A9 00             LDA   #0
0AAA 85 FB             STA   FREE0
0AAC 85 FD             STA   FREE0+2
0AAE A9 04             LDA   #4
0AB0 85 FC             STA   FREE0+1
0AB2 A9 D8             LDA   #216
0AB4 85 FE             STA   FREE0+3
0AB6 A2 18             LDX   #24
0AB8 A0 00    SIDES    LDY   #0
0ABA A9 A0             LDA   #160
0ABC 91 FB             STA   (FREE0),Y
0ABE A0 27             LDY   #39
0AC0 91 FB             STA   (FREE0),Y
0AC2 A9 04             LDA   #WALLS
0AC4 91 FD             STA   (FREE0+2),Y
0AC6 A0 00             LDY   #0
0AC8 91 FD             STA   (FREE0+2),Y
0ACA A5 FB             LDA   FREE0
0ACC 18                CLC
0ACD 69 28             ADC   #40
0ACF 85 FB             STA   FREE0
0AD1 A5 FC             LDA   FREE0+1
0AD3 69 00             ADC   #0
0AD5 85 FC             STA   FREE0+1
0AD7 A5 FD             LDA   FREE0+2
0AD9 18                CLC
0ADA 69 28             ADC   #40
0ADC 85 FD             STA   FREE0+2
0ADE A5 FE             LDA   FREE0+3
0AE0 69 00             ADC   #0
0AE2 85 FE             STA   FREE0+3
0AE4 CA                DEX
0AE5 D0 D1             BNE   SIDES
0AE7 A9 00             LDA   #0
0AE9 85 FB             STA   FREE0
0AEB A9 04             LDA   #4
0AED 85 FC             STA   FREE0+1
0AEF A6 B4             LDX   LEVEL
0AF1 E0 07             CPX   #7
0AF3 90 02             BCC   FINE
0AF5 A2 06             LDX   #6
0AF7 BD 09 0D FINE     LDA   ORDER,X
0AFA 0A                ASL
0AFB AA                TAX
0AFC BD FB 0C          LDA   INDEX,X
0AFF 85 03             STA   TEMP0
0B01 BD FC 0C          LDA   INDEX+1,X
0B04 85 04             STA   TEMP0+1
0B06 A0 FF             LDY   #255
0B08 C8       DRAWLOOP INY
0B09 B1 03             LDA   (TEMP0),Y
0B0B D0 19             BNE   DOIT
0B0D C8                INY
```

```
0B0E B1 03                LDA   (TEMP0),Y
0B10 8D 02 D0             STA   53250
0B13 8D 04 D0             STA   53252
0B16 8D 00 D0             STA   53248
0B19 C8                   INY
0B1A B1 03                LDA   (TEMP0),Y
0B1C 8D 03 D0             STA   53251
0B1F 8D 05 D0             STA   53253
0B22 8D 01 D0             STA   53249
0B25 60                   RTS
0B26 84 02     DOIT       STY   TEMP
0B28 48                   PHA
0B29 48                   PHA
0B2A 29 1F                AND   #31
0B2C 8D 28 13             STA   NUM
0B2F 68                   PLA
0B30 4A                   LSR
0B31 4A                   LSR
0B32 4A                   LSR
0B33 4A                   LSR
0B34 29 06                AND   #6
0B36 AA                   TAX
0B37 A0 00                LDY   #0
0B39 68                   PLA
0B3A 10 02                BPL   NODRAW
0B3C A0 A0                LDY   #160
0B3E 8C 29 13  NODRAW     STY   BLOCK
0B41 A5 FB     DRAW2      LDA   FREE0
0B43 18                   CLC
0B44 7D 37 0C             ADC   DRAWDIR,X
0B47 85 FB                STA   FREE0
0B49 A5 FC                LDA   FREE0+1
0B4B 7D 38 0C             ADC   DRAWDIR+1,X
0B4E 85 FC                STA   FREE0+1
0B50 A0 00                LDY   #0
0B52 AD 29 13             LDA   BLOCK
0B55 F0 11                BEQ   NOBL
0B57 91 FB                STA   (FREE0),Y
0B59 A5 FB                LDA   FREE0
0B5B 85 FD                STA   FREE0+2
0B5D A5 FC                LDA   FREE0+1
0B5F 18                   CLC
0B60 69 D4                ADC   #212
0B62 85 FE                STA   FREE0+3
0B64 A9 04                LDA   #WALLS
0B66 91 FD                STA   (FREE0+2),Y
0B68 A4 02     NOBL       LDY   TEMP
0B6A CE 28 13             DEC   NUM
0B6D D0 D2                BNE   DRAW2
0B6F F0 97                BEQ   DRAWLOOP
0B71 12 9F 54  MESS1      .ASC  "{RVS}{CYN}TIME
                                R: SCORE: CARS: {L
                                EFT}"
0B9B 94 20 00             .BYT  148,32,0
0B9E 12 9F 20  MESS2      .ASC  "{RVS}{CYN}{OFF
                                }GAME OVER - PRE
                                SS FIREBUTTON{RV
                                S} "
0BC0 00                   .BYT  0
0BC1 12 9F 20  MESS3      .ASC  "{RVS}{CYN} "
0BE1 00                   .BYT  0
0BE2 9C 12 53  MESS4      .ASC  "{PUR}{RVS}SCRE
                                EN "
0BEB 00                   .BYT  0
                                3 DOWN}{12 LEFT
                                }{BLU} SCREEN 1"
0C17 11 11 11             .ASC  "{3 DOWN}{12 LEF
                                T}PRESS FIREBUTT
                                ON"
0C36 00                   .BYT  0
```

```
0C37 D8 FF FF  DRAWDIR    .WORD -40,-1,40,1
0C3F 4B FF 86  SCREEN1    .BYT  75,255,134,205,0,55
                                ,180
0C46 6A D1 A3  SCREEN2    .BYT  106,209,163,230,10
                                2,198,144,163,230
0C4F 65 86 C5             .BYT  101,134,197,227,20
                                4,163,230,0,251,66
0C59 45 EC 47  SCREEN3    .BYT  69,236,71,37,238,13
                                9,106,67,205,170
0C63 C2 02 B4             .BYT  194,2,180,69,101
0C68 00 2D C8             .BYT  0,45,200
0C6B 68 C3 45  SCREEN4    .BYT  104,195,69,167,235,
                                195,131,228,131
0C74 C3 E6 C3             .BYT  195,230,195,131,22
                                6,161,4,131,71,230,
                                131
0C7F C3 E3 C3             .BYT  195,227,195,69,229,
                                171,131,198,131
0C88 AA 83 C6             .BYT  170,131,198,131,17
                                0,131,198,69,101,13
                                1,0,250,188
0C95 66 C6 A1  SCREEN5    .BYT  102,198,161,227,16
                                2,200,225,39,226,16
                                1
0C9F 65 C4 45             .BYT  101,196,69,36,225,9
                                9,5,234,194,130,234
0CAA 81 C1 E5             .BYT  129,193,229,70,129,
                                5,227,6,228,161,194
0CB5 82 AD 81             .BYT  130,173,129,193,16
                                6,194,130,164,132,2
                                25
0CBF A1 82 A1             .BYT  161,130,161,225,12
                                9,104,5,195,161,225
                                ,195
0CCA EA C1 81             .BYT  234,193,129,226,6,1
                                93,0,220,70
0CD3 65 D3 E9  SCREEN6    .BYT  101,211,233,100,22
                                6,129,193,238
0CDB 8E A2 24             .BYT  142,162,36,170,36,1
                                64,201,232,100,236
0CE5 00 73 89             .BYT  0,115,137
0CE8 46 F2 82  SCREEN7    .BYT  70,242,130,225,194,
                                236,197,226,193
0CF1 A2 C4 AC             .BYT  162,196,172,194,16
                                1,130,177,0,45,200
0CFB 3F 0C 46  INDEX      .WORD SCREEN1,SCREEN2,
                                SCREEN3,SCREEN4,
                                SCREEN5,SCREEN6
0D07 E8 0C                .WORD SCREEN7
0D09 00 06 01  ORDER      .BYT  0,6,1,5,2,4,3
;
; friction
;
0D10           FRICTION   =     *
0D10 CE 27 13             DEC   FRICT
0D13 D0 39                BNE   NOFRICT
0D15 A9 07                LDA   #7
0D17 8D 27 13             STA   FRICT
0D1A CE 2A 13             DEC   SLOWL
0D1D D0 0F                BNE   NOSLOW
0D1F A9 4B                LDA   #75
0D21 8D 2A 13             STA   SLOWL
0D24 AD 26 13             LDA   MOVH
0D27 C9 07                CMP   #7
0D29 F0 03                BEQ   NOSLOW
0D2B EE 26 13             INC   MOVH
0D2E AD 23 13  NOSLOW     LDA   VELX
0D31 F0 0B                BEQ   DY
0D33 10 06                BPL   POS1
0D35 EE 23 13             INC   VELX
```

```
0D38 4C 3E 0D           JMP   DY
0D3B CE 23 13  POS1     DEC   VELX
0D3E AD 24 13  DY       LDA   VELY
0D41 F0 0B              BEQ   NOFRICT
0D43 10 06              BPL   POS2
0D45 EE 24 13           INC   VELY
0D48 4C 4E 0D           JMP   NOFRICT
0D4B CE 24 13  POS2     DEC   VELY
0D4E 60        NOFRICT  RTS
;
; calculate the speed
;
0D4F           CALC     =     *
0D4F AE 20 13           LDX   SPRPTR
0D52 AD 23 13           LDA   VELX
0D55 18                 CLC
0D56 7D 77 0D           ADC   TABX,X
0D59 C9 46              CMP   #MAX
0D5B 90 04              BCC   YESX
0D5D C9 B9              CMP   #MIN
0D5F 90 03              BCC   NOTX
0D61 8D 23 13  YESX     STA   VELX
0D64 AD 24 13  NOTX     LDA   VELY
0D67 18                 CLC
0D68 7D 7F 0D           ADC   TABY,X
0D6B C9 46              CMP   #MAX
0D6D 90 04              BCC   YESY
0D6F C9 B9              CMP   #MIN
0D71 90 03              BCC   NOTY
0D73 8D 24 13  YESY     STA   VELY
0D76 60        NOTY     RTS
0D77 01 00 FF  TABX     .BYT  1,0,255,255,255,0,1,
                              1
0D7F FF FF FF  TABY     .BYT  255,255,255,0,1,1,1,
                              0
;
; move the sprite
;
0D87           SPRMOV   =     *
0D87 A9 01              LDA   #01
0D89 8D 19 D0           STA   $D019
0D8C AD 1B 13           LDA   SPRX+1
0D8F 8D 06 D0           STA   53254
0D92 AD 1E 13           LDA   SPRY+1
0D95 8D 07 D0           STA   53255
0D98 AD 1C 13           LDA   SPRX+2
0D9B 0A                 ASL
0D9C 0A                 ASL
0D9D 0A                 ASL
0D9E 8D 10 D0           STA   53264
0DA1 AD 20 13           LDA   SPRPTR
0DA4 18                 CLC
0DA5 69 D4              ADC   #212
0DA7 8D FB 07           STA   2043
0DAA A9 FA              LDA   #250
0DAC 8D 12 D0           STA   $D012
0DAF AD 0D DC           LDA   $DC0D
0DB2 29 01              AND   #1
0DB4 F0 03              BEQ   SKIP
0DB6 4C 31 EA           JMP   $EA31
0DB9 4C BC FE  SKIP     JMP   $FEBC
;
; set up raster interrupt
;
0DBC           RASTSET  =     *
0DBC A9 1B              LDA   #27
0DBE 8D 11 D0           STA   53265
0DC1 A9 7F              LDA   #127
0DC3 8D 0D DC           STA   56333
0DC6 A9 87              LDA   #<SPRMOV
```

```
0DC8 8D 14 03             STA   $0314
0DCB A9 0D                LDA   #>SPRMOV
0DCD 8D 15 03             STA   $0315
0DD0 A9 81                LDA   #129
0DD2 8D 1A D0             STA   53274
0DD5 60                   RTS
;
; initialize start of round
;
0DD6           INITROUND  =     *
0DD6 A9 00                LDA   #0
0DD8 8D 1C 13             STA   SPRX+2
0DDB 8D 23 13             STA   VELX
0DDE 8D 24 13             STA   VELY
0DE1 A9 26                LDA   #STARTX
0DE3 8D 1B 13             STA   SPRX+1
0DE6 A9 3C                LDA   #STARTY
0DE8 8D 1E 13             STA   SPRY+1
0DEB A9 07                LDA   #7
0DED 8D 20 13             STA   SPRPTR
0DF0 A5 A2                LDA   162
0DF2 C5 A2     WTSYNC     CMP   162
0DF4 F0 FC                BEQ   WTSYNC
0DF6 A9 0F                LDA   #15
0DF8 8D 15 D0             STA   53269
0DFB A9 64                LDA   #100
0DFD 8D 21 13             STA   JOYL
0E00 8D 22 13             STA   JOYH
0E03 A9 07                LDA   #7
0E05 8D 25 13             STA   MOVL
0E08 8D 26 13             STA   MOVH
0E0B A9 07                LDA   #7
0E0D 8D 27 13             STA   FRICT
0E10 A9 37                LDA   #55
0E12 8D 2A 13             STA   SLOWL
0E15 60                   RTS
;
; move the car
;
0E16           MOVCAR     =     *
0E16 20 10 0D             JSR   FRICTION
0E19 AD 23 13             LDA   VELX
0E1C 30 1A                BMI   SUBX
0E1E 18                   CLC
0E1F 6D 1A 13             ADC   SPRX
0E22 8D 1A 13             ST    SPRX
0E25 AD 1B 13             LDA   SPRX+1
0E28 69 00                ADC   #0
0E2A 8D 1B 13             STA   SPRX+1
0E2D AD 1C 13             LDA   SPRX+2
0E30 69 00                ADC   #0
0E32 8D 1C 13             STA   SPRX+2
0E35 4C 57 0E             JMP   DOY
0E38 49 FF     SUBX       EOR   #255
0E3A 69 01                ADC   #1
0E3C 85 02                STA   TEMP
0E3E AD 1A 13             LDA   SPRX
0E41 38                   SEC
0E42 E5 02                SBC   TEMP
0E44 8D 1A 13             STA   SPRX
0E47 AD 1B 13             LDA   SPRX+1
0E4A E9 00                SBC   #0
0E4C 8D 1B 13             STA   SPRX+1
0E4F AD 1C 13             LDA   SPRX+2
0E52 E9 00                SBC   #0
0E54 8D 1C 13             STA   SPRX+2
0E57 AD 24 13  DOY        LDA   VELY
0E5A 30 18                BMI   SUBY
0E5C 18                   CLC
0E5D 6D 1D 13             ADC   SPRY
```

```
0E60 8D 1D 13               STA    SPRY
0E63 AD 1E 13               LDA    SPRY+1
0E66 69 00                  ADC    #0
0E68 8D 1E 13               STA    SPRY+1
0E6B AD 1F 13               LDA    SPRY+2
0E6E 69 00                  ADC    #0
0E70 8D 1F 13               STA    SPRY+2
0E73 60                     RTS
0E74 49 FF     SUBY         EOR    #255
0E76 18                     CLC
0E77 69 01                  ADC    #1
0E79 85 02                  STA    TEMP
0E7B AD 1D 13               LDA    SPRY
0E7E 38                     SEC
0E7F E5 02                  SBC    TEMP
0E81 8D 1D 13               STA    SPRY
0E84 AD 1E 13               LDA    SPRY+1
0E87 E9 00                  SBC    #0
0E89 8D 1E 13               STA    SPRY+1
0E8C AD 1F 13               LDA    SPRY+2
0E8F E9 00                  SBC    #0
0E91 8D 1F 13               STA    SPRY+2
0E94 60                     RTS
;
; joystick routine
;
0E95           JOYSTICK     =      *
0E95 AD 01 DC               LDA    $DC01
0E98 4A                     LSR
0E99 4A                     LSR
0E9A 4A                     LSR
0E9B B0 12                  BCS    NOTLEFT
0E9D 20 D3 0E               JSR    SCREECH
0EA0 EE 20 13               INC    SPRPTR
0EA3 AE 20 13               LDX    SPRPTR
0EA6 E0 08                  CPX    #8
0EA8 D0 05                  BNE    NOTLEFT
0EAA A2 00                  LDX    #0
0EAC 8E 20 13               STX    SPRPTR
0EAF 4A        NOTLEFT      LSR
0EB0 B0 0D                  BCS    NOTRIGHT
0EB2 20 D3 0E               JSR    SCREECH
0EB5 CE 20 13               DEC    SPRPTR
0EB8 10 05                  BPL    NOTRIGHT
0EBA A2 07                  LDX    #7
0EBC 8E 20 13               STX    SPRPTR
0EBF 4A        NOTRIGHT     LSR
0EC0 B0 10                  BCS    NOTFIRE
0EC2 AD 26 13               LDA    MOVH
0EC5 C9 02                  CMP    #2
0EC7 F0 09                  BEQ    NOTFIRE
0EC9 CE 26 13               DEC    MOVH
0ECC AD 26 13               LDA    MOVH
0ECF 8D 25 13               STA    MOVL
0ED2 60        NOTFIRE      RTS
0ED3 48        SCREECH      PHA
0ED4 A9 80                  LDA    #128
0ED6 8D 0B D4               STA    54283
0ED9 A9 81                  LDA    #129
0EDB 8D 0B D4               STA    54283
0EDE 68                     PLA
0EDF 60                     RTS
;
; set up sprites
;
0EE0           SETSPRITE    =      *
0EE0 A0 00                  LDY    #0
0EE2 B9 5A 0F  COPLOOP      LDA    SPRDEF,Y
0EE5 99 00 35               STA    13568,Y
0EE8 B9 5A 10               LDA    SPRDEF+256,Y
```

```
0EEB 99 00 36               STA    13568+256,Y
0EEE B9 5A 11               LDA    SPRDEF+512,Y
0EF1 99 00 37               STA    13568+512,Y
0EF4 B9 5A 12               LDA    SPRDEF+768,Y
0EF7 99 00 38               STA    13568+768,Y
0EFA A9 00                  LDA    #0
0EFC 99 00 39               STA    13568+1024,Y
0EFF 88                     DEY
0F00 D0 E0                  BNE    COPLOOP
0F02 A0 3F                  LDY    #63
0F04 B9 DA 12  MORE         LDA    GARAGE,Y
0F07 99 80 39               STA    13568+1024+128
                                   ,Y
0F0A 88                     DEY
0F0B 10 F7                  BPL    MORE
0F0D A0 02                  LDY    #2
0F0F A9 FF                  LDA    #255
0F11 99 00 39  VOOP1        STA    13568+1024,Y
0F14 99 3C 39               STA    13568+1084,Y
0F17 88                     DEY
0F18 10 F7                  BPL    VOOP1
0F1A A0 36                  LDY    #54
0F1C A9 80                  LDA    #128
0F1E 99 03 39  VOOP2        STA    13568+1027,Y
0F21 88                     DEY
0F22 88                     DEY
0F23 88                     DEY
0F24 10 F8                  BPL    VOOP2
0F26 A9 0C                  LDA    #12
0F28 8D 5C 39               STA    13568+1024+92
0F2B 8D 62 39               STA    13568+1024+98
0F2E A2 E4                  LDX    #228
0F30 8E F9 07               STX    2041
0F33 E8                     INX
0F34 8E FA 07               STX    2042
0F37 E8                     INX
0F38 8E F8 07               STX    2040
0F3B A0 03                  LDY    #3
0F3D B9 56 0F  ZOOP         LDA    SPRCOL,Y
0F40 99 27 D0               STA    53287,Y
0F43 88                     DEY
0F44 10 F7                  BPL    ZOOP
0F46 A9 08                  LDA    #8
0F48 8D 1C D0               STA    53276
0F4B A9 00                  LDA    #0
0F4D 8D 25 D0               STA    53285
0F50 A9 07                  LDA    #7
0F52 8D 26 D0               STA    53286
0F55 60                     RTS
0F56 02 00 02  SPRCOL       .BYT   2,0,2,2
0F5A 00 00 00  SPRDEF       .BYT   0,0,0,0,0,0,0,0
0F62 00 00 00               .BYT   0,0,0,016,0,0,096,0
0F6A 00 28 00               .BYT   0,040,0,0,169,4,2,13
                                   2
0F72 14 8A 00               .BYT   020,138,0,082,168,0
                                   ,075,224
0F7A 00 2F A0               .BYT   0,047,160,0,042,132
                                   ,0,010
0F82 14 00 02               .BYT   020,0,2,080,0,0,064,
                                   0
0F8A 00 00 00               .BYT   0,0,0,0,0,0,0,0
0F92 00 00 00               .BYT   0,0,0,0,0,0,0,0
0F9A 00 00 00               .BYT   0,0,0,1,033,0,1,169
0FA2 00 01 21               .BYT   0,1,033,0,0,032,0,0
0FAA 20 00 00               .BYT   032,0,0,032,0,0,032,
                                   0
0FB2 00 20 00               .BYT   0,032,0,0,168,0,0,18
                                   4
0FBA 00 00 B8               .BYT   0,0,184,0,4,184,064,
                                   5
```

```
0FC2 A9 40 05          .BYT   169,064,5,169,064,4
                              ,168,064
0FCA 00 00 00          .BYT   0,0,0,0,0,0,0,0
0FD2 00 00 00          .BYT   0,0,0,0,0,0,0,0
0FDA 00 00 00          .BYT   0,0,0,0,0,0,0,0
0FE2 00 04 00          .BYT   0,4,0,0,9,0,0,040
0FEA 00 00 6A          .BYT   0,0,106,0,0,018,128,
                              016
0FF2 00 A2 14          .BYT   0,162,020,0,043,133
                              ,0,011
0FFA E1 00 0A          .BYT   225,0,010,232,0,018
                              ,168,0
1002 14 A0 00          .BYT   020,160,0,5,128,0,1,
                              0
100A 00 00 00          .BYT   0,0,0,0,0,0,0,0
1012 00 00 00          .BYT   0,0,0,0,0,0,0,0
101A 00 00 00          .BYT   0,0,0,0,0,0,0,0
1022 00 00 00          .BYT   0,0,0,0,0,0,0,0
102A 00 00 00          .BYT   0,0,0,0,084,0,0,084
1032 40 00 10          .BYT   064,0,016,064,010,1
                              68,170,171
103A E8 AA AB          .BYT   232,170,171,232,06
                              4,010,168,064
1042 00 10 00          .BYT   0,016,0,0,084,0,0,08
                              4
104A 00 00 00          .BYT   0,0,0,0,0,0,0,0
1052 00 00 00          .BYT   0,0,0,0,0,0,0,0
105A 00 00 00          .BYT   0,0,0,0,0,0,0,0
1062 00 00 00          .BYT   0,0,0,0,0,0,0,0
106A 00 00 00          .BYT   0,0,0,1,0,0,5,128
1072 00 14 A0          .BYT   0,020,160,0,018,232
                              ,0,011
107A E8 00 0B          .BYT   232,0,011,161,0,042
                              ,133,0
1082 A2 14 12          .BYT   162,020,018,128,01
                              6,106,0,0
108A 28 00 00          .BYT   040,0,0,9,0,0,4,0
1092 00 00 00          .BYT   0,0,0,0,0,0,0,0
109A 00 00 00          .BYT   0,0,0,0,0,0,0,0
10A2 00 00 00          .BYT   0,0,0,0,0,0,0,0
10AA 00 00 04          .BYT   0,0,4,168,064,5,169,
                              064
10B2 05 A9 40          .BYT   005,169,064,4,184,0
                              64,0,184
10BA 00 00 B8          .BYT   0,0,184,0,0,168,0,0
10C2 A8 00 00          .BYT   168,0,0,032,0,0,032,
                              0
10CA 00 20 00          .BYT   0,032,0,0,032,0,1,03
                              3
10D2 00 01 A9          .BYT   0,1,169,0,1,033,0,0
10DA 00 00 00          .BYT   0,0,0,0,0,0,0,0
10E2 00 00 00          .BYT   0,0,0,0,0,0,0,0
10EA 00 00 00          .BYT   0,0,0,064,0,2,080,0
10F2 0A 14 00          .BYT   010,020,0,043,132,0
                              ,043,224
10FA 00 4A E0          .BYT   0,074,224,0,082,168
                              ,0,020
1102 8A 00 04          .BYT   138,0,4,2,132,0,0,16
                              9
110A 00 00 28          .BYT   0,0,040,0,0,096,0,0
1112 10 00 00          .BYT   016,0,0,0,0,0,0,0
111A 00 00 00          .BYT   0,0,0,0,0,0,0,0
1122 00 00 00          .BYT   0,0,0,0,0,0,0,0
112A 00 00 15          .BYT   0,0,021,0,0,021,0,0
1132 04 00 01          .BYT   004,0,1,042,160,1,0
                              43,234
113A AA 2B EA          .BYT   170,043,234,170,04
                              2,160,1,4
1142 00 01 15          .BYT   0,1,021,0,0,021,0,0
114A 00 00 00          .BYT   0,0,0,0,0,0,0,0
```

```
1152 00 00 00          .BYT   0,0,0,0,0,0,0,0
115A 00 00 00          .BYT   0,0,0,0,0,0,0,1
1162 00 00 07          .BYT   0,0,7,064,016,7,064,
                              016
116A 01 00 01          .BYT   001,0,1,048,040,8,1
                              28,056
1172 02 80 C0          .BYT   002,128,192,4,162,3
                              ,0,152
117A 01 0C AB          .BYT   001,012,171,1,2,168
                              ,128,4
1182 22 B0 03          .BYT   034,176,3,0,096,012
                              ,016,0
118A 28 C0 40          .BYT   040,192,064,032,1,2
                              08,192,1
1192 D0 00 00          .BYT   208,0,0,064,0,0,0,0
119A 00 00 00          .BYT   0,0,0,0,1,0,064,7
11A2 40 40 07          .BYT   064,064,7,064,0,049
                              ,0,4
11AA 00 0A 20          .BYT   0,010,032,0,014,0,1
                              28,0
11B2 02 88 C0          .BYT   002,136,192,066,04
                              2,131,0,168
11BA 00 30 2B          .BYT   0,048,043,032,2,168
                              ,012,016
11C2 20 80 00          .BYT   032,128,0,0,040,3,0,
                              064
11CA 30 00 00          .BYT   048,0,0,0,016,016,1
                              31,0
11D2 74 A0 00          .BYT   116,160,0,116,0,0,0
                              16,0
11DA 00 07 40          .BYT   0,7,064,064,1,0,0,0
11E2 00 04 30          .BYT   0,4,048,0,0,0,0,128
11EA 00 00 02          .BYT   0,0,2,128,0,0,138,12
                              8
11F2 00 2A 0C          .BYT   0,042,012,066,8,3,2,
                              139
11FA 00 C0 20          .BYT   0,192,032,224,202,0
                              40,3,208
1202 A8 00 02          .BYT   168,0,2,0,040,048,0
                              32,2
120A 0C 00 40          .BYT   012,0,064,192,0,0,0,
                              016
1212 01 03 00          .BYT   001,3,0,7,128,0,7,0
121A 00 01 00          .BYT   0,1,0,016,048,0,0,0
1222 00 00 00          .BYT   0,0,0,0,2,136,0,2
122A 20 00 00          .BYT   032,0,0,050,160,0,0
                              10,3
1232 00 00 80          .BYT   0,0,128,0,130,0,2,16
                              8
123A 00 20 08          .BYT   0,032,8,224,058,0,0,
                              040
1242 02 08 40          .BYT   002,8,064,2,2,0,032,
                              0
124A 00 00 00          .BYT   0,0,0,048,0,016,0,0
1252 00 00 10          .BYT   0,0,016,0,3,0,1,0
125A 40 30 00          .BYT   064,048,0,0,0,0,8,0
1262 00 0A 00          .BYT   0,010,0,0,0,8,0,0
126A 30 2A 00          .BYT   048,042,0,0,160,0,0,
                              0
1272 0A 00 20          .BYT   010,0,032,2,0,128,0,
                              8
127A A0 20 00          .BYT   160,032,0,184,204,0
                              ,0,040
1282 00 00 20          .BYT   0,0,032,0,2,0,2,0
128A 00 22 00          .BYT   0,034,0,0,0,0,192,0
1292 00 00 00          .BYT   0,0,0,1,0,016,0,0
129A 08 30 00          .BYT   008,048,0,040,0,0,0,
                              0
12A2 00 00 02          .BYT   0,0,2,0,0,012,0,0
12AA 00 0A A0          .BYT   0,010,160,0,0,128,0,0
```

```
12B2 00 00 02             .BYT   0,0,2,0,0,8,0,0
12BA 0A 80 00             .BYT   010,128,0,011,048,0
                                 ,032,0
12C2 00 00 00             .BYT   0,0,0,0,0,0,0,0
12CA 00 00 00             .BYT   0,0,0,0,0,0,0,0
12D2 00 08 02             .BYT   0,8,2,1,0,018,0,0
12DA 00 00 00   GARAGE    .BYT   000,000,000,063,25
                                 5,252,095,255
12E2 FA 5F FF             .BYT   250,095,255,250,11
                                 1,255,246,111
12EA FF F6 77             .BYT   255,246,119,255,23
                                 8,119,255,238
12F2 7B FF DE             .BYT   123,255,222,123,24
                                 3,222,124,000
12FA 3E 7B F3             .BYT   062,123,243,222,12
                                 3,255,222,119
1302 FF EE 77             .BYT   255,238,119,255,23
                                 8,111,255,246
130A 6F FF F6             .BYT   111,255,246,095,25
                                 5,250,095,255
1312 FA 3F FF             .BYT   250,063,255,252,00
                                 0,000,000,255
131A            SPRX      =      *
131A            SPRY      =      *+3
131A            SPRPTR    =      *+6
131A            JOYL      =      *+7
131A            JOYH      =      *+8
131A            VELX      =      *+9
131A            VELY      =      *+10
131A            MOVL      =      *+11
131A            MOVH      =      *+12
131A            FRICT     =      *+13
131A            NUM       =      *+14
131A            BLOCK     =      *+15
131A            SLOWL     =      *+16
```

Appendices

How to Type In Programs

In order to make it as easy as possible to type in the programs in this book, we've included two program entry aids written in BASIC: "The Automatic Proofreader" and "MLX." To assist you in understanding how to enter these programs, COMPUTE! has established the following listing conventions.

Generally, BASIC program listings like the one for MLX contain words within braces which spell out any special characters: {DOWN} means to press the cursor-down key; {5 SPACES} means to press the space bar five times.

To indicate that a key should be *shifted* (press the key while holding down the SHIFT key), the key is underlined in our listings. For example, S means to type the S key while holding down the SHIFT key. This will appear on your screen as a heart symbol. If you find an underlined key enclosed in braces, for example, {10 N}, you should type the key as many times as indicated. In that case, you would enter ten shifted N's.

If a key is enclosed in special brackets, [<>], you should hold down the *Commodore key* while pressing the key inside the special brackets. (The Commodore key is the key in the lower-left corner of the keyboard.) Again, if the key is preceded by a number, you should press the key as many times as indicated; for example, [<9@>] means to type Commodore-@ nine times.

Refer to the following table when entering cursor and color control keys:

Keyboard Conventions

When You Read:	Press:		See:
{CLR}	SHIFT	CLR/HOME	
{HOME}		CLR/HOME	
{UP}	SHIFT	↑ CRSR ↓	
{DOWN}		↑ CRSR ↓	
{LEFT}	SHIFT	← CRSR →	
{RIGHT}		← CRSR →	
{RVS}	CTRL	9	
{OFF}	CTRL	0	
{BLK}	CTRL	1	
{WHT}	CTRL	2	
{RED}	CTRL	3	
{CYN}	CTRL	4	
{PUR}	CTRL	5	
{GRN}	CTRL	6	
{BLU}	CTRL	7	
{YEL}	CTRL	8	

When You Read:	Press:		See:
{ 1 }	COMMODORE	1	
{ 2 }	COMMODORE	2	
{ 3 }	COMMODORE	3	
{ 4 }	COMMODORE	4	
{ 5 }	COMMODORE	5	
{ 6 }	COMMODORE	6	
{ 7 }	COMMODORE	7	
{ 8 }	COMMODORE	8	
{ F1 }		f1	
{ F2 }	SHIFT	f1	
{ F3 }		f3	
{ F4 }	SHIFT	f3	
{ F5 }		f5	
{ F6 }	SHIFT	f5	
{ F7 }		f7	
{ F8 }	SHIFT	f7	
←	←		
↑	SHIFT	↑	

The Automatic Proofreader

Philip I. Nelson

"The Automatic Proofreader" helps you type in program listings for the Commodore 64 and 128, and prevents nearly every kind of typing mistake.

Type in the "Proofreader" *exactly* as listed. Since the program can't check itself, type carefully to avoid mistakes. Don't omit any lines, even if they contain unfamiliar commands. After finishing, save a copy or two on disk or tape before running it. This is important because the Proofreader erases the BASIC portion of itself when you run it, leaving only the machine language portion in memory.

Next, type **RUN** and press RETURN. After announcing which computer it's running on, the Proofreader displays the message *Proofreader Active*. Now you're ready to type in a BASIC program.

Two Letters

Every time you finish typing a line and press RETURN, the Proofreader displays a two-letter checksum in the upper-left corner of the screen. Compare this result with the two-letter checksum printed to the left of the line in the program listing.

> The one program in this book that can be used in conjunction with the Proofreader is "MLX: Machine Language Entry Program," which is found in Chapter 2. Make sure you use the Proofreader to type in MLX, since it is vital that you have the latter program entered correctly. If there is an error in MLX, you will not be able to type in any of the seven machine language games in this book.

If the letters match, it's almost certain the line was typed correctly. If the letters don't match, check for your mistake and correct the line.

What It Does, What It Doesn't

The Proofreader ignores spaces not enclosed in quotation marks, so you can omit or add spaces between keywords and still see a matching checksum. However, since spaces *inside* quotation marks are almost always significant, the Proofreader pays attention to them. For example, **10 PRINT"THIS IS BASIC"** generates a different checksum than does **10 PRINT"THIS ISBA SIC"**.

A common typing error is transposition—typing two successive characters in the wrong order, like PIRNT instead of PRINT or 64378 instead of 64738. The Proofreader is sensitive to the *position* of each character within the line and thus catches transposition errors.

The Proofreader does *not* accept keyword abbreviations (for example, ? instead of PRINT). If you prefer to use abbreviations, you can still check the line by LISTing it after typing it in, moving the cursor back to the line, and pressing RETURN. LISTing the line substitutes the full keyword for the abbreviation and allows the Proofreader to work properly. The same technique works for rechecking programs you've already typed in.

If you're using the Proofreader on the Commodore 128, *do not perform any GRAPHIC commands while the Proofreader is active.* When you perform a command like GRAPHIC 1, the computer moves everything at the start of BASIC program space—including the Proofreader—to another memory area, causing the Proofreader to crash. The same thing happens if you *run* any program with a GRAPHIC command while the Proofreader is in memory.

Though the Proofreader doesn't interfere with other BASIC operations, it's a good idea to disable it before running another program. However, the Proofreader is purposely difficult to dislodge: It's not affected by tape or disk operations, or by pressing RUN/STOP–RESTORE. The simplest way to disable it is to turn the computer off, then on. A gentler method is to SYS to the computer's built-in reset routine (SYS 65341 for the 128; 64738 for the 64). These reset routines erase any program in memory, so be sure to save the program you're typing in *before* entering the SYS command.

If you own a Commodore 64, you may already have wondered whether the Proofreader works with other programming utilities like "MetaBASIC." The answer is, generally, yes, *if you're using a 64 and activate the Proofreader after installing the other utility.* For example, first load and activate MetaBASIC, then load and run the Proofreader.

When using the Proofreader with another utility, you should disable *both* programs before running a BASIC program. While the Proofreader seems unaffected by most utilities, there's no way to promise that it will work with any and every combination of utilities you might want to use. The more utilities activated, the more fragile the system becomes.

The Automatic Proofreader

Read Appendix A before entering this program.

```
10 VEC=PEEK(772)+256*PEEK(773):LO=43:HI=44
20 PRINT"{CLR}{WHT}AUTOMATIC PROOFREADER FOR ";:IF
    VEC=42364 THEN PRINT "C-64"
30 IF VEC=50556 THEN PRINT "VIC-20{BLU}"
40 IF VEC=35158 THEN WAIT CLR:PRINT "PLUS/4 & 16"
50 IF VEC=17165 THEN LO=45:HI=46:WAIT CLR:PRINT"12
   8{WHT}"
60 SA=(PEEK(LO)+256*PEEK(HI))+6:ADR=SA
70 FOR J=0 TO 166:READ BYT:POKE ADR,BYT:ADR=ADR+1:
   CHK=CHK+BYT:NEXT
80 IF CHK<>20570 THEN PRINT "*ERROR* CHECK TYPING
   {SPACE}IN DATA STATEMENTS":END
90 FOR J=1 TO 5:READ RF,LF,HF:RS=SA+RF:HB=INT(RS/2
   56):LB=RS-(256*HB)
100 CHK=CHK+RF+LF+HF:POKE SA+LF,LB:POKE SA+HF,HB:N
   EXT
110 IF CHK<>22054 THEN PRINT "*ERROR* RELOAD PROGR
   AM AND CHECK FINAL LINE":END
120 POKE SA+149,PEEK(772):POKE SA+150,PEEK(773)
130 IF VEC=17165 THEN POKE SA+14,22:POKE SA+18,23:
   POKESA+29,224:POKESA+139,224
140 PRINT CHR$(147);CHR$(17);"PROOFREADER ACTIVE":
   SYS SA
150 POKE HI,PEEK(HI)+1:POKE (PEEK(LO)+256*PEEK(HI)
   )-1,0:NEW
160 DATA 120,169,73,141,4,3,169,3,141,5,3
170 DATA 88,96,165,20,133,167,165,21,133,168,169
180 DATA 0,141,0,255,162,31,181,199,157,227,3
190 DATA 202,16,248,169,19,32,210,255,169,18,32
200 DATA 210,255,160,0,132,180,132,176,136,230,180
210 DATA 200,185,0,2,240,46,201,34,208,8,72
```

```
220 DATA 165,176,73,255,133,176,104,72,201,32,208
230 DATA 7,165,176,208,3,104,208,226,104,166,180
240 DATA 24,165,167,121,0,2,133,167,165,168,105
250 DATA 0,133,168,202,208,239,240,202,165,167,69
260 DATA 168,72,41,15,168,185,211,3,32,210,255
270 DATA 104,74,74,74,74,168,185,211,3,32,210
280 DATA 255,162,31,189,227,3,149,199,202,16,248
290 DATA 169,146,32,210,255,76,86,137,65,66,67
300 DATA 68,69,70,71,72,74,75,77,80,81,82,83,88
310 DATA 13,2,7,167,31,32,151,116,117,151,128,129,
   167,136,137
```

To order your copy of *COMPUTE!'s Machine Language Games Disk,* call our toll-free US order line: 1-800-346-6767 (in NY 212-887-8525) or send your prepaid order to:

COMPUTE!'s Machine Language Games Disk
COMPUTE! Publications
P.O. Box 5038
F.D.R. Station
New York, NY 10150

All orders must be prepaid (check, charge, or money order). NC residents add 4.5% sales tax. NY residents add 8.25% sales tax.

Send ____ copies of *COMPUTE!'s Machine Language Games Disk* at $12.95 per copy. (610BDSK)

Subtotal $__________

Shipping and Handling: $2.00/disk $__________

Sales tax (if applicable) $__________

Total payment enclosed $__________

□ Payment enclosed
□ Charge □ Visa □ MasterCard □ American Express

Acct. No. ______________________________ Exp. Date ________
(Required)

Name ____

Address ____

City ________________________ State ________ Zip ________

Please allow 4-5 weeks for delivery.

COMPUTE! Books

Ask your retailer for these **COMPUTE! Books** or order directly from **COMPUTE!**.

Call toll free (in US) **800-346-6767** (in NY 212-887-8525) or write COMPUTE! Books, P.O. Box 5038, F.D.R. Station, New York, NY 10150.

Quantity	Title	Price*	Total
______	SpeedScript: The Word Processor for the Commodore 64 and VIC-20 (94-9)	**$ 9.95**	______
______	Commodore SpeedScript Book Disk	**$12.95**	______
______	128 Machine Language for Beginners (033-5)	**$16.95**	______
______	COMPUTE!'s First Book of the Commodore 128 (059-9)	**$14.95**	______
______	COMPUTE!'s Commodore 64/128 Collection (97-3)	**$12.95**	______
______	All About the Commodore 64, Volume Two (45-0)	**$16.95**	______
______	All About the Commodore 64, Volume One (40-X)	**$12.95**	______
______	Programming the Commodore 64: The Definitive Guide (50-7)	**$24.95**	______
______	COMPUTE!'s Data File Handler for the Commodore 64 (86-8)	**$12.95**	______
______	COMPUTE!'s Kids and the Commodore 128 (032)	**$14.95**	______
______	Kids and the Commodore 64 (77-9)	**$12.95**	______
______	COMPUTE!'s Commodore Collection, Volume 1 (55-8)	**$12.95**	______
______	COMPUTE!'s Commodore Collection, Volume 2 (70-1)	**$12.95**	______
______	COMPUTE!'s Personal Accounting Manager for the Commodore 64 (014-9)	**$12.95**	______
______	COMPUTE!'s VIC-20 and Commodore 64 Tool Kit: BASIC (32-9)	**$16.95**	______
______	COMPUTE!'s VIC-20 and Commodore 64 Tool Kit: Kernal (33-7)	**$16.95**	______
______	COMPUTE!'s Telecomputing on the Commodore 64 (009)	**$12.95**	______
______	COMPUTE!'s VIC-20 Collection (007)	**$12.95**	______
______	Programming the VIC (52-3)	**$24.95**	______
______	VIC Games for Kids (35-3)	**$12.95**	______
______	Mapping the VIC (24-8)	**$14.95**	______
______	COMPUTE!'s 128 Programmer's Guide (0319)	**$16.95**	______
______	The Complete 64 (Book/Disk Combination) (062-9)	**$29.95**	______
______	COMPUTE!'s Machine Language Games for the Commodore 64 (0610)	**$16.95**	______

*Add $2.00 per book for shipping and handling.
Outside US add $5.00 air mail or $2.00 surface mail.

NC residents add 4.5% sales tax __________
NY residents add 8.25% sales tax __________
Shipping & handling: $2.00/book __________
Total payment __________

All orders must be prepaid (check, charge, or money order).
All payments must be in US funds.
□ Payment enclosed.
Charge □ Visa □ MasterCard □ American Express

Acct. No.______________________________ Exp. Date__________
(Required)

Name____

Address____

City________________________ State __________ Zip__________

*Allow 4–5 weeks for delivery.
Prices and availability subject to change.
Current catalog available upon request.

Say YES now to COMPUTE!'s GAZETTE

You can search far and wide and you simply *won't* find a better magazine...a better source of insightful, stimulating, usable information for your Commodore computer than *COMPUTE!'s Gazette.*

COMPUTE's Gazette works harder...digs deeper...researches further—all to help guarantee that you get the absolute most from your Commodore.

Subscribe today and each and every month you'll receive up to 20 all-new action-packed programs. That's up to 200 programs each year—game programs...education programs...home applications programs...personal and budgeting programs...sorting and filing programs.

Add it up for yourself. Where else can you get exciting programs each month...expert advice...insightful analysis...up-to-the-minute software reviews and so much more—all for just $18.

So why wait. Subscribe now to *COMPUTE!'s Gazette*—and get the most from your Commodore computer. Return card below—or call 1-800-247-5470 (in Iowa, 1-800-532-1272).

COMPUTE!'s
GAZETTE
FOR COMMODORE PERSONAL COMPUTER USERS
Shootout
Power BASIC:
64 RAM Disk
Now you can have instant access to an extra 8K of free memory. This short utility is an excellent programming development tool

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 7551 DES MOINES, IA

POSTAGE WILL BE PAID BY ADDRESSEE

COMPUTE!'s Gazette
P.O. Box 10775
Des Moines, IA 50347-0775

Fast, Furious, Fascinating

Nothing beats machine language (ML) for computer game speed and power. The native language of your Commodore 64 or 128, ML makes possible fast-action arcade games and sophisticated simulations.

The seven games in this book are written entirely in machine language.

- **CUT-OFF!** hurtles two lines of light across the screen, with each player forcing the other into a wall or previous path.
- **Nessie** puts you behind a camera in search of the famed Loch Ness monster.
- **Campaign Manager** simulates the making of a president—from nomination to election day.
- **Heat Seeker** sets you in the cockpit of a jet chased by deadly homing missiles.
- **Laser Beam** sends your character scurrying across a chaotic field of bouncing balls.
- **Whirlybird** rebounds eggs from the bricks below. Let one get by you and the trouble begins.
- **Miami Ice** is a nightmare of slick ice, a sliding car, and guardrails that fill the windshield.

All seven games are ready to type in with our machine language entry program, "MLX." You don't have to know anything about machine language.

But if you do program in machine language or want to learn more, *Machine Language Games* includes the complete source code for each game. Printed in their entirety as written by the original programmers, the source codes give you an invaluable opportunity. Study them, learn from them, or even modify them to suit yourself.

Fast games, furious action, fascinating source code. If you like games, like to program in machine language, or both, you'll love *COMPUTE!'s Machine Language Games for the Commodore 64*.

All games run on the Commodore 64 and Commodore 128 (in 64 mode).

All the programs in this book are also available on a companion disk. See coupon in the back for details.

ISBN 0-87455-061-0

COMPUTE!'s Machine Language Games for the Commodore 64

COMPUTE! Books