APX ATARI® PROGRAM EXCHANGE

Alan Griesemer and Stephen Bradshaw

BLOCK BUSTER

APX-10110 APX-20110

March 1982

User-Written Software for ATARI Home Computers

Alan Griesemer and Stephen Bradshaw

BLOCK BUSTER

APX-10110 APX-20110

March 1982

BLOCK BUSTER

by

H. Alan Griesemer and Stephen C. Bradshaw

TRADEMARKS OF ATARI

The following are trademarks of Atari, Inc.

ATARI®
ATARI 400™ Home Computer
ATARI 800™ Home Computer
ATARI 410™ Program Recorder
ATARI 810™ Disk Drive
ATARI 820™ 40-Column Printer
ATARI 822™ Thermal Printer
ATARI 825™ 80-Column Printer
ATARI 830™ Acoustic Modem
ATARI 850™ Interface Module

****
Distributed by

The ATARI Program Exchange
P. O. Box 427
155 Moffett Park Drive, B-1
Sunnyvale, CA 94086

To request an APX Software Catalog, write to the address above, or call toll-free:

800/538-1862 (outside California)
800/672-1850 (within California)

Or call our Sales number, 408/745-5535.
****

TABLE OF CONTENTS

FIGURES

FIGURES

INTRODUCTION

OVERVIEW

BLOCK BUSTER is a program that can help you solve the infuriating cube puzzle you got last Christmas. BLOCK BUSTER significantly extends the capabilities of physical cubes by letting you

(1) view all sides of the cube at once,

(2) display your moves in a special notation that allows you to document and repeat a successful series of moves,

(3) program sequences of moves that can then be invoked with a single command, and

(4) instantly reset the cube to its solved state at any time.

Additional features include manipulation of the cube using the keyboard, yellow console keys, or an ATARI Paddle Controller; ability to solve the cube to three different patterns using a programmed solution procedure; ability to enter a cube pattern and have the computer solve it; and a timer and move counter.

REQUIRED ACCESSORIES

Cassette version

32K RAM
ATARI 410 Program Recorder

Diskette version

32K RAM
ATARI 810 Disk Drive

OPTIONAL ACCESSORIES

One set of ATARI Paddle Controllers
ATARI 825 80-Column Printer or ATARI 822 Thermal Printer

CONTACTING THE AUTHORS

Users wishing to contact the authors about BLOCK BUSTER may write them at:

P.O. Box 2335
Monterey, CA 93940

GETTING STARTED

LOADING BLOCK BUSTER INTO COMPUTER MEMORY

1. Remove any program cartridges from the cartridge slot of your computer.

2. Plug your Paddle Controllers into the first (the leftmost) controller jack at the front of your computer console.

3. If you have the cassette version of BLOCK BUSTER:

a. Have your computer turned OFF.

b. Insert the BLOCK BUSTER cassette into the program recorder's cassette holder and press REWIND on the recorder until the tape rewinds completely. Then press PLAY to prepare the program recorder for loading the program.

c. Turn on the computer while holding down the START key.

d. When you hear a beep, release the START key and press the RETURN key. The program will load into computer memory and start automatically.

If you have the diskette version of BLOCK BUSTER:

a. Have your computer turned OFF.

b. Turn on your disk drive.

c. When the BUSY light goes out, open the disk drive door and insert the BLOCK BUSTER diskette with the label in the lower right-hand corner nearest to you. (Use disk drive one if you have more than one drive.)

d. Turn on your computer and your TV set. The program will load into computer memory and start automatically.

THE DISPLAY

The first display is the cube in the middle of the screen. The cube is red, blue, green, white, violet, and gold. You may want to adjust your TV or monitor for brighter and more pleasing colors. The image of the cube on the screen has been exploded so that you can see all six sides of the cube at once. The right, bottom and back sides of the cube have been "pulled" away from the rest of the cube. This view is similar to what you would see if there were mirrors to the right, behind, and below a real cube.

In addition to the image of the cube, a timer displays at the top left side of the screen and a move counter displays on the top right. At the bottom of the screen, below the yellow line , you see the text area where typed commands and cube moves appear.

USING BLOCK BUSTER

MANIPULATING THE CUBE

The purpose of BLOCK BUSTER is to find solutions to your puzzle cube. You can manipulate the cube on the screen in three different ways. Use:

(1) an ATARI paddle controller,
(2) the yellow console keys, or
(3) the keyboard.

Using the ATARI Paddle Controller

Using a paddle controller is the easiest way to manipulate the cube. Plug a set of paddle controllers into the leftmost controller jack and turn the paddle. The arrow in the display will move around the edge of the cube. If it doesn't, check the plug and/or try the other paddle. (Only one paddle of the set will work. The other has no effect.)

To rotate the cube, use the paddle to position the arrow over the section of the cube you want to move and briefly press the trigger. The colors on the section of the cube under the arrow will rotate 90 degrees in the same direction as the arrow. The name of the move in cube notation (See the section below) will display at the bottom of the screen.

> Note: You can position the arrow off the edge of the cube causing the entire cube, rather then just one section, to rotate.

Change the direction of the arrow by holding down the trigger instead of releasing it immediately. As you hold down the trigger, the arrow will begin to change direction every half second. When the arrow is pointed in the direction you want to move, release the trigger and the cube will rotate in that direction.

Using the Yellow Console Keys

The Console keys, OPTION, SELECT and START, work much the same as the paddle controller. The OPTION and SELECT keys replace the paddle and the START key replaces the trigger. Pressing the OPTION key moves the arrow around the cube to the right in a fixed sequence, one step at a time. Holding down the OPTION key produces repeated arrow movements to the right. The SELECT key works the in the same way as the OPTION key except that it moves the arrow to the left instead of to the right.

When you have the arrow over the section of the cube you want to move, press the START key briefly and the cube section will rotate in the direction of the arrow. To change the direction of the arrow hold down the START key and wait until the arrow points in the direction you want. Then release the START key to rotate the cube.

Using the Keyboard

Manipulating the cube with keyboard commands is most convenient when you're trying out a sequence of moves or using the programming feature. To use the keyboard, you must first understand the cube notation used by the program. Each rotatable segment of the cube is given a one-letter name. These names are:

L for the left hand face,
V for the vertical band between the left and right face,
R for the right hand face,

F for the front face,
M for the middle band between the front and back faces,
B for the back face,

U for the upper face,
H for the horizontal band between the upper and lower face,
D for the lower (down) face,

Figure 1 shows cube segment names and possible rotation directions.

You can note any cube move with one of these letter names, followed with a plus or minus sign. A plus sign indicates a clockwise movement and a minus sign indicates a counterclockwise rotation.

For example: R+ means a clockwise movement of the right face. R- is a counterclockwise movement of the right face.

To note a rotation of the entire cube, combine the names of the three individual segments followed with a plus or minus sign. The three possible rotations of the entire cube are LVR, FMB, and UHD followed by a plus or minus sign (e.g., LVR+ will move the entire cube).

To rotate the cube from the keyboard, type the cube notation for the move or moves you want to make followed by pressing the RETURN key. For example, to move the right face clockwise type "R+" and press the RETURN key. You can make a series of moves by typing several commands (separated by spaces) before pressing the RETURN key. For example, if you type:

```
R+ L- F- R+
```

and press the RETURN key, the program will make four moves.

When typing commands from the keyboard, you have full use of the ATARI screen editor. You can use the cursor movement keys and the INSERT and DELETE keys to enter your lines. Also, if you want to repeat a command or a slight variation of it, don't type the whole command over again as long as it's still on the screen. Move the cursor up to the command line, make your changes, and press the RETURN key. The program executes the command line just as it appears on the screen.

BLOCK BUSTER distinguishes between logical and physical command lines. Logical command lines can be up to two physical command lines long. If you want to execute a sequence of commands more then one physical line long, continue typing commands past the end of the physical line without pressing the RETURN key. This causes the logical line to extend into the next physical line. When you've typed the last command in the sequence, press the RETURN key. The program will execute all the commands on the logical line. If you try to extend the logical line beyond two physical lines, and press the RETURN key,

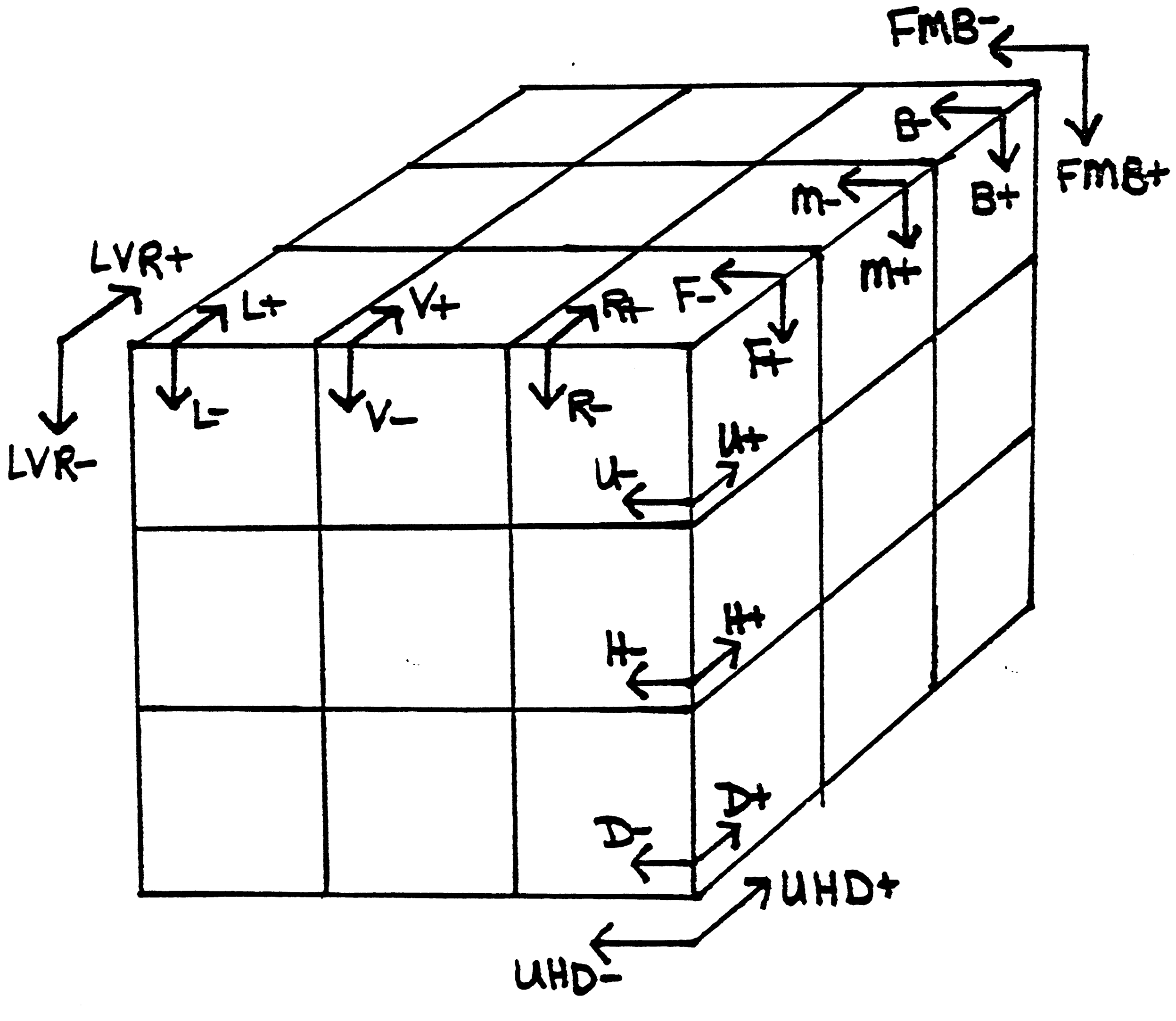

Figure 1. Cube segment names and rotation directions.

BLOCK BUSTER gives you the error message

LOGICAL LINE LONGER THAN 2 PHYSICAL LINES

and won't execute the commands.

SCRAMBLING AND RESETTING THE CUBE

When BLOCK BUSTER starts, the cube is in its solved state. To generate a random cube configuration, type SCRAMBLE and press the RETURN key.

To reset the cube to its solved state press the CLEAR key while holding down the CTRL key. Pressing CTRL-CLEAR also resets the timer and the move counter.

CONTROLLING THE TIMER AND MOVE COUNTER

Two additional commands effect the timer and move counter.

(1) Pressing the CTRL and letter "R" keys resets the timer and counter to zero without resetting the cube to the solved state.

(2) Pressing the CTRL and letter "C" keys creates a switch that turns the timer and move counter on and off.

SOLVING THE CUBE WITH THE COMPUTER

BLOCK BUSTER can solve any scrambled cube. Just type the word, "SOLVE" and press the RETURN key. BLOCK BUSTER will take control of the cube on the screen and solve it, displaying the moves as it goes along.

BLOCK BUSTER can solve any cube in approximately four minutes and 200 moves. Most of the time involves displaying the moves. If BLOCK BUSTER didn't have to spend time communicating with human beings, it could generate the solution in about 15 seconds.

To solve the cube, BLOCK BUSTER performs six basic tasks sequentially. They are:

(1) Position and orient the four top edge cubies (Refer to the section entitled Solving the Cube for the definition of a cubie).
(2) Bring the top corners into position.
(3) Bring the bottom corners into position.
(4) Position and orient the middle band.
(5) Position and orient the four bottom edge cubies.
(6) Orient the corners.

You can watch these stages as BLOCK BUSTER solves the cube. (Refer to the section entitled Advanced Technical Information, for more information on the solution procedure.)

Alternate Solutions

BLOCK BUSTER has two alternate solutions for the cube, the Checkers pattern and the Christmas Cross. The checkers pattern makes every side of the cube look like a checker board. The Christmas cross makes every side of the cube look like a cross. To see these patterns, type CHECKERS or CROSS instead of SOLVE.

The Pause Feature

If you want to interrupt the computer's solution process temporarily, press any key on the keyboard other than the ESC (escape) key. The program will pause after it finishes displaying the current move. The program remains stopped until you press a second key. It then resumes the solution where it left off.

Another use for the "pause feature" is single stepping through all or part of a solution. To single step,

(1) Press any key as described above.
(2) After the program pauses, press any key twice in rapid succession.

The program will generate one move and stop. You can continue single stepping for as long as you desire. To go back to continuous operation, use a single keypress.

To permanently stop the solution process and regain control of the cube, press the ESC key while the program is running (not when it's temporarily halted). BLOCK BUSTER will stop solving the cube and return control to you.

The Attract Feature

By typing ATTRACT you can see BLOCK BUSTER scramble and solve the cube repeatedly, with a 15-second pause between each solution and the next scramble. ATTRACT will first solve the cube to the Checkers pattern, then the Christmas cross, and finally to the standard solid color solution.

To leave the Attract Feature, press the ESC key during the solution process. Control will return to you.

PROGRAMMING SEQUENCES OF MOVES

PROG: and :END

BLOCK BUSTER includes two keyboard commands, PROG: and :END. These commands allow you to program frequently used sequences of moves and execute them with a single command. The general usage of PROG: and :END is:

```
PROG: <command name>
<sequence of moves> :END
```

"Command name" is the name you use to invoke the program. "Sequence of moves" is a series of keyboard notation commands, such as "R+", "U-", etc.

Pay special attention to the necessary spaces between the commands and the command name or sequence of moves.

One useful sequence of commands is

```
B+ D+ B- D+ R+ D- R-
```

This sequence will swap the position of the two bottom right corners, moving the front corner to the back and moving the back corner to the front (the orientation of the corners and the adjacent edges is not maintained however). To program a command named "CSWAP" that executes this sequence you would type:

```
PROG: CSWAP B+ D+ B- D+ R+ D- R- :END
```

and press the RETURN key. After entering this program, type CSWAP and you'll execute the entire sequence of commands in the program.

Once you enter a new command, you can use it just like BLOCK BUSTER's built-in commands. This new command includes using the new commands to program additional commands. For example, if after programming "CSWAP", you want a command, "DIAGSWAP", to swap the right front corner with the left rear corner, you program it as follows:

```
PROG: DIAGSWAP CSWAP UHD+ CSWAP UHD- CSWAP :END
```

Command names used with "PROG:" can be from 1 to 31 characters long. You can use them over again in later "PROG:" statements. If you do use a command-name again, the following warning message displays

```
(command-name) NAME IS NOT UNIQUE OK
```

and the new usage of the command name supersedes all previous uses. This feature is one way to redefine a command. If you find that you've accidently redefined a command by using a command name over again, type "FORGET <command name>" immediately. You've now recovered use of the previous command. Now type another PROG: command with a different command name. Use of the FORGET command is described more fully on the next page.

BLOCK BUSTER uses three additional commands to support the programming features. They are the "SPACE", "LIST", and "FORGET" commands.

SPACE

Typing the word SPACE followed by pressing the RETURN key causes a display of the number of bytes (units of computer memory) left in the programming area. Each command definition requires an amount of space equal to the number of characters in the command name, plus 2 times the number of commands in the sequence of moves, plus 11 bytes.

For example the command:

```
PROG: CSWAP B+ D+ B- D+ R+ D- R- :END
```

requires five bytes (CSWAP), plus fourteen bytes (2 times 7 moves), plus 11 bytes. The total for this command is thirty bytes.

If you attempt to program a command without enough space available, BLOCK BUSTER will print an error message.

LIST

Type the word LIST followed by pressing the RETURN key to display all the user-defined and built-in command names that BLOCK BUSTER knows. The last name printed is always a heart, indicating the end of the list. If the display is longer then one screen, you can use the CTRL and "1" keys to start and stop the listing temporarily. The commands remain on the screen until you press the ESC key which returns you to the cube display. Pressing one of the yellow console keys before the end of the display (i.e. before the heart appears) will stop the listing and immediately return you to the cube.

FORGET

Use the FORGET command to remove commands from the command display in order to make room for new commands, or to erase an unwanted duplicate command name. Only user- programmed commands can be forgotten.To activate FORGET, type FORGET and the command name and press the RETURN key.

****WARNING****

When you use FORGET and the command-name, all previous commands it in the command display will also be forgotten!!

ENTERING A CUBE

BLOCK BUSTER lets you enter any cube pattern. This feature is useful if you have a real cube you want the computer to solve, or if you have a particular cube configuration you want to try to solve yourself, or if you want to enter a standard cube for several different people to solve in a competition.

To enter a cube, press the CTRL and letter "X" keys. A cross hair now appears on the screen. You can move the cross hair from block to block on the front face of the cube by turning the paddle controller or pressing the OPTION and SELECT keys. Change the color of the block under the cross hair by pressing the paddle trigger or the START key. Repeated pressing or holding down of the trigger or START key will cycle the block through all six possible colors, allowing you to select the color for that block.

Since the cross hair can cover only the front face of the cube, you must rotate the cube to bring each face to the front. To do so, either enter a rotation command from the keyboard, or press CTRL-A to redisplay the arrow. Now you can use the paddle or SELECT/START keys to perform the rotation. In either case, you'll have to press CTRL-X to redisplay the cross hair after the rotation.

The CHECK Command

When entering a cube, you must be careful that the cube is legal. A cube must have the correct number of cubies each color. It must also have the proper edge and corner pieces. One way to enter a cube is by using a real cube as a model and checking the BLOCK BUSTER cube against the real cube. The CHECK command can also help find an illegal cube by checking to see that all the required edge, corner, and face pieces exist somewhere on the cube. This check won't find all possible illegal cubes, but it will detect the most common cube entry mistakes.

To use the CHECK command, type CHECK and press the RETURN key. The CHECK command will print an error message if it finds anything wrong with the cube. Otherwise, it will print OK.

THE PRINT FEATURE

If you have a printer, BLOCK BUSTER can print the moves that you or the solution routines make. To start printing your moves, attach your printer to the computer, turn it and the interface module, if applicable, on, make sure it's in online mode, and press the CTRL and letter "P" keys (no RETURN is required). The text area will turn green as an indication that the print feature is working. Each move you or the computer makes (except those generated by SCRAMBLE) is sent to the printer. However, the printer will not actually print the moves until it has accumulated a whole line. To turn the print feature off, press CTRL-P again. The command window will turn from green to black when the print feature is off.

Certain types of printer errors (e.g., "printer offline") will cause problems for BLOCK BUSTER. If you have problems, the command window will turn bright green and all the text will disappear. If this condition occurs, check your printer, correct any problems, and wait. Within a minute BLOCK BUSTER will recover from the error. The text will reappear, and the text window will be dark green, indicating the print feature is "on". If you didn't correct the printer problem, BLOCK BUSTER will turn the print feature off and display an error message.

ADVANCED TECHNICAL INFORMATION

SOLVING THE CUBE

In the March 1981 issue of Scientific American Douglas Hofstedter wrote of the 43,252,003,274,489,856,000 possible positions and orientations for a solvable cube. Obviously, the number of possibilities makes it impossible to solve the cube by random twisting. A form of strategy is necessary. The strategy that BLOCK BUSTER uses is to break the total solution problem into a number of smaller, more manageable problems. Instead of looking at the whole cube at once, BLOCK BUSTER deals with components of the cube called "cubies" and "cubicles".

"Cubie" is the name for the small cubes that make up the larger cube. Figure 2 shows the three different types of cubies: corner cubies, edge cubies, and face cubies. A complete cube has a total of eight corner cubies, twelve edge cubies, and six face cubies. "Cubicle" is the name for the location where you can place a cubie on the cube.

Different types of cubies have different numbers of exposed surfaces. Corner cubies have three, edge cubies have two, and face cubies have one exposed surface. A particular cubie can only be in a certain cubicle. For example, you can't move corner cubies, with three exposed surfaces, to an edge cubicle having room for only two exposed surfaces. You can't move face cubies at all! No matter how you twist the cube, the six face cubies always have the same color face cubie opposite them and the same four face cubies around them. Because the relative position of the face cubies is unchanging, the solution pattern (algorithm) uses the face cubies to determine which color belongs on each face of the cube.

To talk about the cube, it's convenient to give names to each of the cubicles. Naming lets you identify them without having to refer to color names (which change all the time as we twist the cube). BLOCK BUSTER uses a notation developed by David Singmaster, a professor of mathematical sciences and computing at the Polytechnic Institute of the South Bank, London.

The six face names presented in Figure 1 (i.e. Front, Right, Back, Left, Up, and Down) create the basis for Singmaster's notation. In this system, each cubical has a name consisting of the initials of all the faces of its exposed surfaces. Cubical names use lowercase letters to avoid confusion with the face names. For example "ul" is the name of the edge cubicle straddling the Up and Left faces of the cube. The left, front, up corner cubicle is "lfu" (or "ufl", or "flu" because all combinations of "l", "f", and "u" refer to the corner cubicle that's located in the upper left front corner). Naming face cubicles requires only one letter. For exmple, "f" is the front face cubicle. Figure 3 shows the names of each of the cubicles.

The distinction between the position and orientation of the cubies is important for the solution algorithm. Correct position means that a cubie is in the correct cubical. However, a cubie in the correct cubical may still be incorrectly oriented. It may not have the colors on the proper surfaces. For example, when BLOCK BUSTER's cube is in the solved state, the cubie in the "ur" cubicle is blue on the top and white on the side. This is the correct orientation for this cubie. If you have white on the top and blue on the side you have the correct position and incorrect orientation.

Breaking the cube down into cubies and cubicles lets us evaluate partial solutions instead of having to consider the entire solution at once. For example, even in a completlely random cube the cubie that belongs in the "uf" cubicle can only be in one of

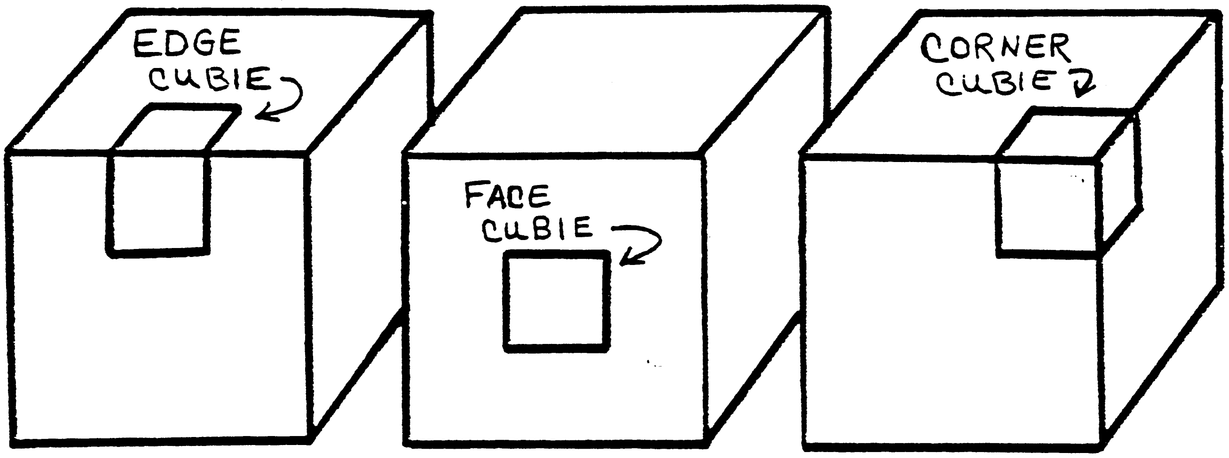

Figure 2. The three types of cubies.

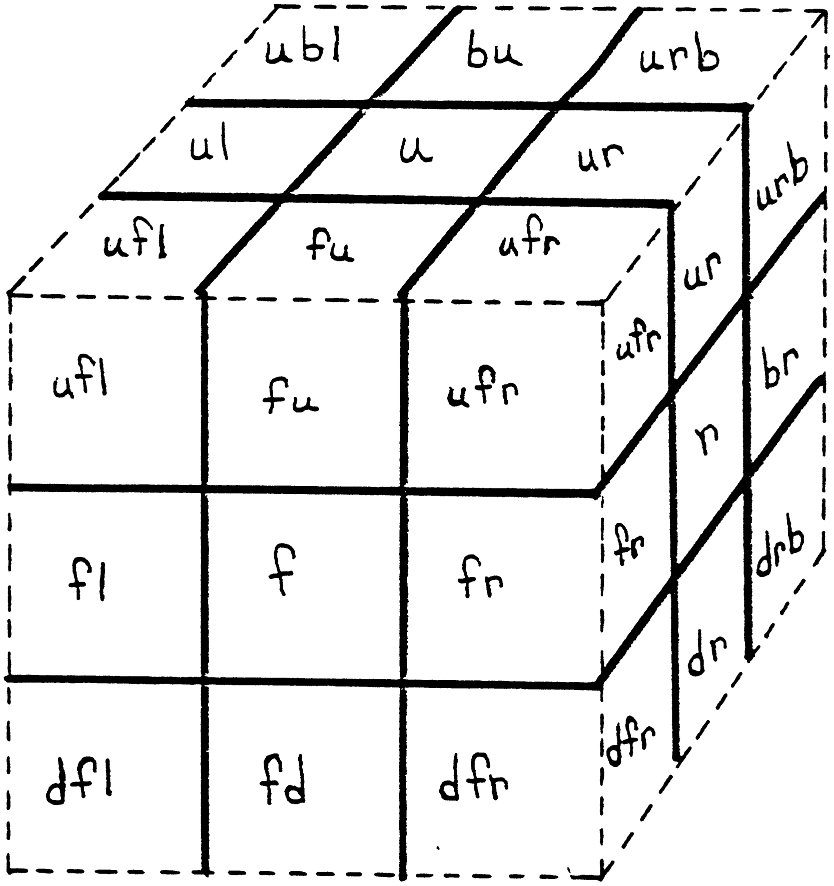

Figure 3. Cubicle names.

the twelve edge cubies, and can be oriented in only one of two ways. This results in only 24 possible positions for that cubie. You can easily handle these possibilities on a microcomputer (even a human could do it!). BLOCK BUSTER solves the cube by breaking it down into a series of relatively simple problems and solving them one by one. BLOCK BUSTER uses an eleven-step solution process which is described below.

STEP 1

> Find the cubie that belongs in edge cubicle "uf". Move it to cubicle "uf" so that it's oriented correctly.

In Step 1, BLOCK BUSTER searches for the cubie that belongs in the "uf" cubical and uses its position as a pointer into a table of twists for edge cubies. This table has 24 different twist sequences corresponding to the 24 positions and orientations that an edge cubie can have in the cube. For example, if the cubie that belongs in the "uf" cubicle is located in the "rd" cubicle, the solution algorithm executes a sequence of twists from the table causing the cubie to move from "rd" to "uf".

Because of the problem of orientation, BLOCK BUSTER actually has two sequences of moves in the table which move a cubie from "rd" to "uf". The one used depends on the orientation of the cubie. In the above example, the two sequences in the table are:

> V- D- V+ and D- V- D+ D+ V+

The solving algorithm uses the first sequence if the up face color (as determined by the "u" face cubie) is the same as the color on the right face of the cubie in "rd". If the "U" face color is on the "d" face of the cubie in "rd", the solving algorithm uses the second sequence of moves. You may want to try out these two sequences of moves on BLOCK BUSTER to see how they move a cubie from "rd" to "uf".

STEP 2

> Find the cubie that belongs in the "ufr" corner cubicle. Move it to cubicle "ufr" ignoring the cubie's orientation.

Step 2 is easier than Step 1. The table of moves for corner cubies has only eight move routines corresponding to the eight corners. Also, at this point the algorithm doesen't need to worry about the orientations of the corner cubies because Step 11 will orient all the corners. An example of a corner moving sequence used in Step 2 is

> R- D+ R+

BLOCK BUSTER uses this sequence if the cubie belonging in the "ufr" cubicle is in the "dfl" cubicle.

STEP 3

> Repeat Steps 1 and 2 for the "ur" edge and the "urb" corner, the "ub" edge and the "ubl" corner, and the "ul" edge and the "ulf" corner.

We can carry out step 3 in several possible ways. The first way is to have separate tables of twists for each cubicle positioned in Step 3. This method takes a lot of memory. A second possibility is to use successive rotations of the whole cube to move, in turn, each of the additional cubicles to be positioned into "uf" and "ufr". This method allows us to use the same tables over and over again, but requires BLOCK BUSTER to make many extra moves.

BLOCK BUSTER uses a third approach. It uses a special routine to transpose any sequence of moves keyed to cubicles on one face so that the sequence will work with cubicles on another face. This technique rotates the command sequences instead of the cube. Although the initial implementation of this technique requires some difficult programming and a bit of memory space, it provides a general tool used extensively throughout the rest of the solution pattern.

To see how the move translater works, use the three-twist sequence

R- D+ R+

This twist sequence moves a cubie from "dfl" to "ufr". If the solving algorithm is working on the "urb" cubicle and finds the correct cubie in the "frd" cubicle, it could use the same three-twist sequence preceded by a "UHD+" command (i.e., "UHD+ R- D+ R+"). Instead, the solving algorithm generates the moves it needs directly by transposing "R- D+ R+" to "B+ D+ B-". This transposition moves the "frd" cubie to "urb" without rotating the cube. Try it yourself and see!

Steps 1, 2, and 3 all handle the "up" face of the cube. They assume that the cube is totally scrambled when the solution pattern begins so when the algorithm performs steps 1-3, it doesn't preserve the cubie positions or orientations in the lower two levels of the assumption cube. This aspect of the algorithm makes the first three steps easier for the solution process because the steps require only a few twists (usually only 3-5). The assumption also explains why the algorithm may take a slightly scrambled cube (e.g., only one twist) and makes it appear much more random before it's solved. If you've scrambled any part of the "up" face, the algorithm will disturb the bottom of the cube while trying to solve the top. If you use the SOLVE command when a partally solved cube is on the screen, it's best to have the solved portion of the cube at the top. This process reduces the number of moves required to solve the cube.

STEP 4

Find the cubie that belongs in the "frd" corner. Move it to "frd" ignoring the cubie's orientation.

By now all of the "U" plane corners have found their place. Therefore, the cubie belonging in the "frd" cubicle must be located somewhere in the bottom (D) plane. The algorithm finds the cubie and does "D+" or "D-" until the cubie is in the "frd" cubicle.

STEP 5

Repeat Step 4 for the "rbd" and the "bld corners".

Step 5 places the rest of the corners through a sequence of twists that swap adjacent corner cubies. (Note! Once you've placed the first seven corners the final "lfd" corner must be in its place because there's no where else it can be). An example of a sequence of twists that will swap the cubie in the "frd" cubicle with the cubie in the "rbd" cubicle is as follows:

```
B+ D+ B- D+ R+ D- R-
```

The corner swapping sequences used in Step 4 can't disturb the "Up" face cubies placed in Steps 1 through 3. However, these sequences disregard the orientation of any of the corners. Step 11 will take care of all corner orientations. These sequences will also move around the edge cubies of the lower two levels of the cube since they haven't been placed yet.

STEP 6

Find the cubie that belongs in the edge cubicle "fl". Move it to cubicle "fl" so that it's oriented correctly.

STEP 7

Repeat Step 6 for the edges "fr", "rb", and "bl".

STEP 8

Find the cubie that belongs in the edge cubicle "fd". Move it to cubicle "fd" ignoring the orientation.

STEP 9

Find the cubie that belongs in the edge cubicle "ld". Move it to cubicle "ld" ignoring the orientation. Due to a specific constraint on the cube, edges "bd" and "rd" will now automatically be in their correct cubicles. (If the first 18 cubies are in place the last two cubies must also be in place.)

Step 6, 7, 8, and 9, need a sequence of twists that move the edges to new cubicles, leaving the corners in their proper places. The sequence of twists the solving algorithm uses to move edge cubies is:

```
F+ D- F- D+ F+ D- F- D+
```

This sequence moves the "fr" edge to the "rd" edge, the "rd" edge to the "fd" edge, and the "fd" edge to the "fr" edge. It rotates the edges clockwise around the "frd" corner. This sequence is known as TWIST+ because of its clockwise movement. It rotates the edges without moving any of the corners out of their correct cubicles but doesn't maintain the orientation of the corners.

TWIST- is another sequence. It moves the same three edges ("fr","fd","fd") in a counterclockwise direction. This sequence is:

```
R- D+ R+ D- R- D+ R+ D-
```

Steps 6 through 9 use combinitions of TWIST+ and TWIST- interspersed with a few additional commands to place the rest of the edges into their proper cubicles. For example, you can move a cubie from "ld" to "fr" with "D- TWIST-".

STEP 10

If edge cubie "fd" is oriented incorrectly, flip it and edge cubie "ld". If edge cubie "ld" is now incorrectly oriented, flip it and flip edge cubie "bd". If edge cubie "bd" is now incorrectly, oriented flip it and flip edge cubie "rd". Now all edges on the cube will be in the solved position.

Step 10 flips the orientation of the incorrectly oriented edges. To flip the orientation of two adjacent edge cubies, BLOCK BUSTER uses "TWIST-" followed by "TWIST+". "TWIST-" rotates the "fr", "rd", and "fd" edges counterclockwise. "TWIST+" rotates the same edges clockwise putting them back into position. It has the additional effect of flipping the orientation of the "fr" and "rd" edges.

STEP 11

Orient the corner cubies. Step 11 uses TWIST+ and TWIST- to orient the corners. While TWIST+ rotates the "fr", "rd", and "fd" edge cubies, it also reorients the cubie in corner "ufr". Every time BLOCK BUSTER uses TWIST+, the cubie in "ufr" rotates in a counterclockwise direction. Three moves of TWIST+ in a row will bring the cube back to the orientaion that it had before you gave the three TWIST+ commands. Try it by resetting the cube (press the CTRL and CLEAR keys) and entering the TWIST+ sequence three times. Use BLOCK BUSTER's programming feature to make this easier. Type:

```
PROG: TWIST+ F+ D- F- D+ F+ D- F- D+ :END
```

Then type TWIST+ TWIST+ TWIST+ and press the RETURN key. The cube will end up as it started, in the solved state. TWIST- also works the same way. Three TWIST- sequences will leave the cube in the same state. Step 11 uses TWIST+ and TWIST- in groups of three to reorient the corners without moving edges around. For example, reset the cube (press the CTRL and CLEAR keys) and type

```
U+ TWIST+ U+ TWIST+ U+ TWIST+ U+
```

Now you have three corners oriented differently, but the rest of the cube is the same. BLOCK BUSTER uses similar sequences of twists to orient the corners as required. Complete this step and you have a solved cube.

Although more elegant and efficient approaches exist, the cube can be solved using only the combinations of the two twist sequences, TWIST+ and TWIST-, and the corner swapping sequence presented in Step 5.

The solution algorithm in BLOCK BUSTER doesn't produce the most efficient and elegant solution possible. The primary failing of the step-by-step approach is its inability to analyze the overall structure of the cube before performing each step. For example, if you make one twist in a solved cube, re-positioning any of the "Up" face cubies, the solution algorithm won't detect that the cube is only one twist away from a solution. Instead, it begins at Step 1 where it detects that the top of the cube is out of position and procedes as if the cube were completely random. As a result, the algorithm may take 200 moves to solve a cube that a human, with infinitely more sophisticated pattern recognition abilities, might be able to solve in one move!

The step-by-step approach does work well on random cubes, which will be interesting to most users (except, of course, those few devious ones who will immediately attempt to see "how smart" the program is by giving it cubes that are one or two twists away from the solution). The step-by-step approach allows a relatively compact implementation (requiring roughly 5k of computer memory). This approach saves users the cost of buying another memory module.

ILLEGAL CUBES

Using CTRL-X, it's possible to create cubes that don't have the standard configuration of colors on the sides. We call these alternate configurations illegal cubes. As you will see, some mutations don't really make the cube unsolvable. Rather, the new configuration shifts it into another dimension or orbit where the solution is different from the standard six solid colors. Illegal cubes are of three basic types.

(1) The first type occurs when any particular edge, corner, or face cubie is missing. This situation can happen if you redefine the colors of the cubies in any of the following ways:

(1) Two face cubies have the same colors.
(2) Two edge cubies have the same colors.
(3) An edge cubie has the same colors as an opposing face cubie.
(4) An edge cubie has one color on both sides.
(5) Two corner cubies have identical colors`
(6) A corner cubie has two of its colors from opposite faces.
(7) A corner cubie has two or more of the same color on its surfaces.

These seven possibilities cover the most common mistakes made when entering a cube in BLOCK BUSTER (such as using too many cubies of a particular color). The CHECK command covers these conditions. Since the solution algorithm evaluates individual cubies, the program will "crash" if one of the cubies is missing. To keep from crashing when you have missing cubies, the SOLVE command calls CHECK before it begins. SOLVE will stop the solution if CHECK finds any of the conditions listed above.

(2) The second type of illegal cube occurs when you have the correct colors on the corner cubies incorrectly oriented. For example, start with a solved cube and switch two of the colors on the "ufr" cubie using CTRL-X. You can never solve the cube because you can never orient the "ufr" corner correctly. If you scramble the cube, the solving algorithm will solve the cube as far as it can. When the algorithm finishes, one or more corners will be oriented incorrectly, but all cubies will be in their correct cubicles.

(3) The third type happens because of constraints on the positions and orientations of the cubies. You can place the first seven corner cubies in any of eight corner cubicals. The eighth corner must be placed in the last corner cubicle with a forced orientation to get back to the solved state. For example, if you remove a corner cubie from a solved cube and put it back in, twisted from its origional orientation, the cube can't be solved. Instead, you now have a new set of 43,252,003,274,489,856,000 possible positions and orientations that are different from the original set! Group theorists call this new set of positions and orientations an "orbit".

In the above example, the we removed the corner cubie and put it back in, twisted to get the cube into a different orbit. If we had twisted the corner cubie in the other direction, it would be in another orbit. At first it seems that you could go on forever twisting corners and creating new and different orbits but this is not the case. If you twist one corner one way and a different corner the other way, the cube is back into its solvable orbit.

Try creating new orbits by rotating the "ufr" corner. Reset the cube (press the CTRL and CLEAR keys) and put the program into the change color mode (press the CTRL and letter "X" keys). Move the cross hair to the upper right corner and change the color from red to white. Type an "LVR-" command and press the CTRL and letter "X" keys to get the cross hair back. Change the color of the lower right corner face from blue to red. Now perform a "UHD-" command and change the color of the lower left corner face from white to blue. Type "UHD+ LVR+" to move the cube back to its original position. Notice that the "ufr" corner now has a 1/3 twist that we call a +1/3 twist because the direction of rotation is clockwise.

Now twist the "ufl" corner with a -1/3 twist. Change the upper left corner face from red to gold. Do an "LVR-" and change the color of the lower left hand corner from blue to red. Do a "UHD+" and change the lower right hand corner from gold to blue. Do a "UHD- LVR+" to bring the cube back to its origional position. Notice the twist on the "ufl" cubie is opposite in direction from the twist on the "ufr" cubie. Now type SOLVE. Twenty-six moves later, BLOCK BUSTER will correctly solve the cube. Since the twist on the two corners was +1/3 and -1/3, it makes the total twist on the cube zero. If the corner twists don't add up to zero, the cube isn't in a solvable orbit.

Experiment with corner twists using the above technique. Enter a +1/3 corner twist on the cube and type SOLVE. The solving algorithm will transfer the twist to different cubies. When it finishes, the cube will still have a corner with a +1/3 twist. (Note that a +2/3 twist is really the same thing as a -1/3 twist and a -2/3 twist is the same as a +1/3 twist.) Based only on corner twists, the cube can have at least three different orbits: 0 twist; +1/3 twist; and -1/3 twist.

When you consider edge twists, more orbits become possible. An edge cubie can have two rather than three orientations. An edge cubie is either oriented correctly or it's flipped over and out of orientation. The cube is not in its solvable orbit if it has an odd number of the edge cubies flipped out of orientation. Try flipping one edge and solving. Try flipping two edges and solving. Try three. No matter how many flipped edges you start with, when the solving algorithm finishes, the solved cube has no more than one filpped edge. Any odd number edge twists will result in a solution with one edge twist. Notice also that the solving routine transfers an edge flip to the bottom of the cube the same way it did with a corner twist. This is because the program places the bottom cubies last.

Another orbit possibility occurs when the program swaps the position (not orientation) of either two adjacent edge cubies or two adjacent corner cubies. Regardless of which edges or corners are swapped, the solving algorithm will transfer the swapped condition to the "bd" edge and the "rd" edge. Again, this facet of the algorithm occurs because it places these cubicles last.

Are there more orbits? Yes, but in fact they are combinations of the three basic orbits already discussed. Using all possible combinitions of the three basic orbits, you have a total of twelve different non-overlapping orbits. They are:

(1) The solvable orbit, no corner twists, no edge filps, no adjacent cubies swapped;
(2) +1/3 corner twist, no edge filps, no adjacent cubies swapped;
(3) -1/3 corner twist, no edge flips, no adjacent cubies swapped;
(4) No corner twists, an odd number of edge twists, no adjacent cubies swapped;
(5) +1/3 corner twist, an odd number of edge flips, no adjacent cubies swapped;
(6) -1/3 corner twist, an odd number of edge flips, and no adjacent cubies swapped;
(7) No corner twists, no edge flips, adjacent cubies swapped;
(8) +1/3 corner twist, no edge flips, adjacent cubies swapped;
(9) -1/3 corner twist, no edge flips, adjacent cubies swapped;
(10) No corner twists, an odd number of edge flips, and adjacent cubies swapped;
(11) +1/3 corner twist, an odd number of edge flips, and adjacent cubies swapped;
(12) -1/3 corner twist, an odd number of edge filps, and adjacent cubies swapped;

Each of these orbits has 43,252,003,274,489,856,000 combinations of positions and orientations. Therefore the cube, counting all twelve orbits, can have 519,024,039,243,878,272,000 possible combinations of positions and orientations. This number is more than enough to challenge any solving algorithm, whether implemented by a human or on a computer!

DUPLICATING YOUR DISKETTE

****WARNING***

The diskette version of BLOCK BUSTER isn't fully compatible with the file structure used by the ATARI Disk Operating System. This incompatibility occurs because BLOCK BUSTER is written in FORTH, which has its own Disk Operating System. The BLOCK BUSTER diskette contains a "pseudo file" named BOOTFILE.4TH. This file contains an auto-loading version of the program. BOOTFILE.4TH has a "Disk Directory" (OPTION A from the DOS menu) and sectors marked as allocated in the Volume Table of Contents (VTOC). However, BOOTFILE.4TH is called a "pseudo file" because it doesn't have sector link pointers in its records. As a result, you can't delete the file (OPTION D. Delete File from the DOS menu), or use the Disk Operating System's copy (OPTION C. Copy File) and duplicate (OPTION O. Duplicate File) file commands. You can use the duplicate disk (OPTION J. Duplicate Disk) command to make a backup copy of the program diskette (for your personal use only). This copy will contain BOOTFILE.4TH and the auto-loading version of BLOCK BUSTER. You can also copy, duplicate, or save most of the other ATARI Disk Operating System files onto a diskette containing BOOTFILE.4TH and use them normally. The one DOS file that can't reside on a diskette with BOOTFILE.4TH is DOS.SYS. Don't use the DOS "Write DOS Files" (OPTION H.) command or the "Rename File" (OPTION E.) command to put a file named DOS.SYS on your BLOCK BUSTER program diskette. If you do, BLOCK BUSTER won't work.

QUICK REFERENCE GUIDE FOR BLOCK BUSTER COMMANDS

ROTATION COMMANDS

PADDLE CONTROLER

Use paddle to select segment.
Press trigger to move segment.
Hold trigger to change direction.

YELLOW CONSOLE KEYS

Use the OPTION & SELECT keys to select segment.
Press the START key to move segment.
Hold the START key to change direction.

KEYBOARD

L+ L-   V+ V-   R+ R-   LVR+ LVR-
F+ F-   M+ M-   B+ B-   FMB+ FMB-
U+ U-   H+ H-   D- D-   UHD+ UHD-

OTHER KEYBOARD COMMANDS (followed by pressing the RETURN key)

SCRAMBLE   Randomly scrambles the cube.

SOLVE, CHECKERS, or CROSS   Solves the cube. Press any key but ESC for temporary halt. Press the ESC key to abort.

ATTRACT   Continuously scrambles and solves the cube. Press ESC to exit.

PROG: <command name> <sequence of moves> :END   Use to program new commands.(Note: You must indicate a space between PROG: and <command name> as well as between and <sequence of moves> and :END.)

LIST   Lists all command names. Press the ESC key to return to the cube.

SPACE   Displays remaining program space.

FORGET <command name>   Forgets comman name and all command names above.

CONTROL COMMANDS (Pressing RETURN not required).

CTRL-CLEAR   Resets cube, timer, and counter.

CTRL-R   Resets timer and counter.

CTRL-C   Turns the timer and counter on and off.

CTRL-X   Calls cross hair for entering cubes. Use your paddles or the SELECT key to move cross hair. Use the trigger on your paddles or the START key to change color.

CTRL-A   Calls the arrow for cube rotation after the cross hair.

CTRL-P   Turns the printer on and off.

LIMITED WARRANTY ON MEDIA AND HARDWARE ACCESSORIES.

We, Atari, Inc., guarantee to you, the original retail purchaser, that the medium on which the APX program is recorded and any hardware accessories sold by APX are free from defects for thirty days from the date of purchase. Any applicable implied warranties, including warranties of merchantability and fitness for a particular purpose, are also limited to thirty days from the date of purchase. Some states don't allow limitations on a warranty's period, so this limitation might not apply to you. If you discover such a defect within the thirty-day period, call APX for a Return Authorization Number, and then return the product along with proof of purchase date to APX. We will repair or replace the product at our option.

You void this warranty if the APX product: (1) has been misused or shows signs of excessive wear; (2) has been damaged by use with non-ATARI products; or (3) has been serviced or modified by anyone other than an Authorized ATARI Service Center. Incidental and consequential damages are not covered by this warranty or by any implied warranty. Some states don't allow exclusion of incidental or consequential damages, so this exclusion might not apply to you.

DISCLAIMER OF WARRANTY AND LIABILITY ON COMPUTER PROGRAMS.

Most APX programs have been written by people not employed by Atari, Inc. The programs we select for APX offer something of value that we want to make available to ATARI Home Computer owners. To offer these programs to the widest number of people economically, we don't put APX products through rigorous testing. Therefore, APX produts are sold "as is", and we do not guarantee them in any way. In particular, we make no warranty, express or implied, including warranties of merchantability and fitness for a particular purpose. We are not liable for any losses or damages of any kind that result from use of an APX product.

ATARI PROGRAM EXCHANGE

REVIEW FORM

We're interested in your experiences with APX programs and documentation, both favorable and unfavorable. Many software authors are willing and eager to improve their programs if they know what users want. And, of course, we want to know about any bugs that slipped by us, so that the software author can fix them. We also want to know whether our documentation is meeting your needs. You are our best source for suggesting improvements! Please help us by taking a moment to fill in this review sheet. Fold the sheet in thirds and seal it so that the address on the bottom of the back becomes the envelope front. Thank you for helping us!

1. Name and APX number of program ______________________________

2. If you have problems using the program, please describe them here.

____

____

____

3. What do you especially like about this program?

____

____

____

4. What do you think the program's weaknesses are?

____

____

____

5. How can the catalog description be more accurate and/or comprehensive?

____

____

6. On a scale of 1 to 10, 1 being "poor" and 10 being "excellent", please rate the following aspects of this program?

_____ Easy to use
_____ User-oriented (e.g., menus, prompts, clear language)
_____ Enjoyable
_____ Self-instructive
_____ Useful (non-game software)
_____ Imaginative graphics and sound

7. Describe any technical errors you found in the user instructions (please give page numbers).

8. What did you especially like about the user instructions?

9. What revisions or additions would improve these instructions?

10. On a scale of 1 to 10, 1 representing "poor" and 10 representing "excellent", how would you rate the user instructions and why?

11. Other comments about the software or user instructions:

```
 -----
|     |
|STAMP|
|     |
 -----
```

ATARI Program Exchange
Attn: Publications Dept.
P.O. Box 50047
60 E. Plumeria Drive
San Jose, CA 95150

[seal here]