

Antic[®]

THE ATARI RESOURCE
PRESENTS

Envision

Designed and written by Darren Schebek

Copyright ©1986 by ARTIFEX

Envision

Designed & Written by Darren Schebek

Copyright ©1986 by Artifax

This documentation may not be copied, photocopied, reproduced, translated, or telecommunicated in any form, in whole or in part, without the prior written consent of Antic Publishing, Inc.

The accompanying program material may not be duplicated, in whole or in part, for any purpose. No copies of the floppy disk(s) may be sold or given to any person or other entity.

Notwithstanding the above, the documentation and accompanying disk(s) may be duplicated for the sole use of the original purchaser.

Antic is a trademark of Antic Publishing, Inc.

This program was created entirely in assembly language using Synapse Software's SynAssembler.

Hardware Requirements:

-) Atari 800 (with 48K)/800XL/130XE
-) One Atari-compatible disk drive
-) One Joystick

Starting Up:

Turn on your disk drive. Remove all cartridges from your computer. Insert the ENVISION diskette into the disk drive and turn on the computer while holding down the <OPTION> key. In a few seconds, the ENVISION title screen will appear. Press <START> to begin loading the ENVISION program. In a few more seconds, ENVISION's Map Editor display will appear, and you may begin.

Introduction:

Congratulations on your purchase of the ENVISION package. You now have the most versatile and powerful text editor available for the Atari computer. ENVISION consists of two programs in one. The first is a font editor, which allows you to redefine the ATARI character set to your liking. The second program is a map editor, which allows you to design your own screens and maps using the font(s) you redefine. You can also use the map editor to design, set up, and view animation sequences of up to 128 frames! These frames can be as small as 1 character, or as large as 20 characters wide by 10 characters high! The size of your maps is not bound by the limits of the TV display but can, in fact, be many times larger, containing as many as 4096 characters (a normal text screen has only 960 characters).

The second disk has three utility programs which print out the maps, display them as smooth-scrolling screens, and convert MicroPainter format pictures to character graphics. Boot the front of the disk for instructions on how to use them. The back of the disk has sample maps and fonts. Investigate the maps to get a feel for the capabilities of ENVISION.

The way ENVISION works is best described as creating your own jigsaw puzzle. The font editor allows you to make the pieces of the "jigsaw puzzle", while the map editor lets you put the pieces in place, using any character, or "puzzle piece", as many times as you wish.

The font editor and the map editor are both in memory at the same time, but behave as two independent programs, so that you can load or save many different fonts with the font editor, and not disturb the map currently in memory.

About the Font Editor:

When ENVISION is all loaded up and running, you will see the map editor display on the screen. Well, we're going to begin this little tour with the font editor, so press the <OPTION> key to get to the font editor (to get back to the map editor, just press <OPTION> again - simple, eh?). At the top of the font editor display, there are two status lines. At the left side of the first status line, there will be one of two messages. If you are running ENVISION on an Atari 800 or 800XL computer, you should see "<inverse F> Font" printed there. This indicator is toggled by the <SELECT> key. If you press <SELECT>, this indicator will change to "ROM Set", telling you that you are now looking at the Atari ROM character set (don't worry, you don't wipe out the font you're working with by doing this). This simply tells you what you're looking at in the font display area. If you are running ENVISION on an Atari 130XE, then this indicator will read "Font A" with an inverse "A" preceding it.

Atari 130XE Note

With the 130XE, you get a bonus feature: 16 fonts in memory at the same time! And you can use any or all of these fonts on your map wherever you want them! Each of the fonts are labeled with a letter. The first font is labeled "A", the second is "B", and so on, all the way to Font "P". To select any of these fonts to edit, use the FONT SELECT command (activated by typing the <F> key from the font editor), and type the appropriate letter for that font. On an 800XL or 800 computer, however, there is no FONT SELECT command. This is the only major difference between the 800/800XL version and the 130XE version. But fret not! You can still print maps that were created on a 130XE to a printer, and you can view ANY map by running the Map Scroll Program (a separate utility included with the ENVISION package - more on this later).

At the middle of the first status line, it should read "ATASCII= 000 <heart>". This tells you that you are currently editing the graphic symbol <heart>, and that its ATASCII (ATari ASCII) value is 000. The second status line gives you values for CR0, CR1, CR2, CR3, and CB. These are values for the color registers 0,1,2,3, and the background, respectively.

Now look at the text on either side of the large green editing grid. These are most of the commands supported by the font editor, with the first letter of each command being highlighted. The highlighted character is the letter you would type to activate that command. Not all of the font editor's functions are listed here; a list of those not appearing will be given later.

The large green editing grid is where you can make changes to any given character with the joystick. The character you are editing will be enlarged and displayed on the grid. The default character is <heart>, and there will be an enlarged <heart> on the editing grid when the program is first run.

Beneath the editing grid is the font itself. It defaults to the ROM images for each character (something has to be there to begin with), but each and every character can be changed and modified to your own liking. This area is a normal Antic 2 (Graphics 0) text mode, and any changes made to any character will be displayed here. There is a gray-green flashing cursor sitting on the <heart> character (the power-up default). This cursor denotes the character you are currently editing and shall be known from now on as the "font cursor" (mighty original name at that!)

Lastly, beneath the font display, there lies the text mode display area. It shows the character you are currently editing (<heart> on power-up) in each of the Atari's six text modes. They are, from top to bottom, as follows:

<u>ANTIC Mode</u>	<u>Graphics Mode</u>	
	<u>800</u>	<u>XL/XE</u>
2	0	0
3	N/A	N/A
4	N/A	12
6	1	1
5	N/A	13
7	2	2

Font Editor Commands:

Here are the commands for the font editor, along with an explanation of each. You can escape from virtually every command by pressing the <ESC> key (of course). The commands are listed in alphabetical order for easy reference.

<A> - Atari

This command will restore the character you are currently editing back to its original ROM version.

 - Blank

Will blank the current character that you are editing.

<CLEAR> - Clear Out Entire Font

This command is useful when you are designing a graphic font or some form of large, multi-character letters. When you type <CLEAR> (it's actually the "less than" symbol, but <<> looks rather silly), you will be asked if you're sure you want to clear out all the characters in the font you are editing. This is a rather necessary confirmation, since you don't want to slaughter two hours of work with the touch of a key, do you? If you confirm this by typing the <Y> key, the whole font area will go blank, since all of the characters in the font will be erased (they're still there, you just can't see 'em).

<C> - Copy To

Allows you to send the current character image to another character. When activated, this command will first prompt you to type (on the keyboard) the character you wish to COPY TO. If you wish to copy the current character to an upper-case character such as "A" or "\$", then you must type a <SHIFT A>, or a <SHIFT 4>, as the <A> key by itself will be interpreted as a lower-case "a". The same goes with control characters, too. This is because the CAPS/LOWR key is (what is commonly known as in the key-masking business) a "dead" key. The current character will be copied over to the character chosen.

<D> - Disk Directory

This command will perform a directory of the disk in drive 1, and display all font files (those with an extension of ".FNT"). The editing grid will be temporarily removed from the screen, and the filenames will be listed in its place. If there isn't enough room on the screen for all the files, a "Press Any Key" message will appear, prompting you to (what else?) press any key. The filenames on the screen will then scroll up one line and the next file will be displayed underneath. All of the filenames will have been displayed when the "free sectors" message appears. Pressing any key at this point will clear the filenames from the screen, restore the editing grid, and exit the directory command. However, you can exit the directory command at any point by pressing the <ESC> key.

<E> - Edit Character

Will prompt you to select a new character to edit. Character input is the same as in the COPY TO command. After making a selection, the chosen character will be enlarged to the editing grid, and will also be displayed in each text mode at the bottom of the screen.

<F> - Font Select (130XE Only)

This command is only available when ENVISION is running on an Atari 130XE. It is disabled on an 800 or 800XL machine. FONT SELECT allows you to select a font to edit from the 16 that are supported on the 130XE. When asked, you simply type the letter of the font you wish to edit, or you can press <ESC> to exit. The range is <A> through to <P>. After making your selection, the font will appear in the font display area and its name will appear on the left side of the top status line.

<G> - GTIA Toggle for Font Area

If you wish to design a font that uses the color power of the Atari's GTIA modes, this little key can make the going much easier. When you type the <G> key, the font display and the character display beneath it will be affected by GTIA mode 9, a 1 color, 16 shade mode. By typing <G> again, the same area will now be affected by GTIA mode 11, a 16 color, 1 shade mode. Typing <G> a third time will get you back to the normal display.

<I> - Inverse

This command essentially makes a "negative" of the current character by turning on all pixels in the current character that are off, and (conversely) turning off all pixels that are on. Here is an example using the "A" character (a "-" denotes pixels that are off, and a "+" denotes a pixel that is on):

-----	+++++++
---+---	+++--+++
--++++--	+------+
-+---+-	+---+---+
-+---+-	+---+---+
-+++++-	+-----+
-+---+-	+---+---+
-----	+++++++
Before	After
Inverse	Inverse

<L> - Load Font from Disk

The LOAD command will prompt you for the filename of the font that you wish to load. Since numbers are not allowed as the first character in a filename, the number keys will be "dead" when the prompt cursor is in the first position. Use the <DELETE/BACK S> key to correct mistakes, or press <ESC> to abort the LOAD command altogether. Wildcards ("*") and question marks ("?") are not allowed in the filename. Press <RETURN> after you have typed in the filename. The screen will then blank while the font is loaded, and will be restored when the load is complete.

Atari 130XE Note:

On the 130XE, the font will load into the font you were last editing. For example, if you are editing font "E", and you load another font, it will replace whatever is in font "E" with the new font.

<M> - Mirror

This command will flip the current character horizontally as demonstrated below with the ROM version of the "C" character:

-----	-----
--++++--	--++++--
-++--++-	-++--++-
-++-----	-----++-
-++-----	-----++-
-++--++-	-++--++-
--++++--	--++++--
-----	-----
Before	After
Mirror	Mirror

<O> - Overlay

Allows you to take the image of another character and place it over top of the current character's image. The current image is, however, not totally replaced by the chosen character's image, as no pixels are turned off as a result of the overlay. Character input is the same as in the COPY TO command.

<P> - Poke Color Register

This command allows you to select colors to be used with your font. When activated, you will be prompted to select which color register you wish to change. This is a single key input (<0>, <1>, <2>, <3>, or) and pressing the <RETURN> key is not necessary. Next, you will be asked for a new value for the chosen color register (The range is 0 to 255). You can enter a value in decimal, or, if you like, in hexadecimal by first typing a "\$", and then a two digit hexadecimal number (from 00 to FF). Since this input requires more than one keypress, you must press <RETURN> after typing your number. You may back up the prompt cursor to correct mistakes by pressing the <DELETE/BACK S> key. Once you press <RETURN>, the color register at the bottom of the font editor display will change to the color you specified, and the second status line will reflect your chosen color, as well. The map editor will also use these color values when you go to the map editor.

<R> - Rotate Character

This command will rotate the current character 90 degrees clockwise, as demonstrated below with the "A" character:

-----	-----
---+---	-++++---
--++++--	-+++++---
-++--++-	--+---++-
-++--++-	--+---++-
-+++++---	-+++++---
-++--++-	-++++---
-----	-----
Before	After
Rotate	Rotate

<S> - Save Font to Disk

This command allows you to save the font you have designed to disk. You will be prompted to enter the filename under which the font is to be saved. See the LOAD command for filename input. The screen will blank while your font is saved, and will be restored when the save is complete. Bear in mind that you need not always save fonts by themselves, as they are automatically saved with the map-editor's SAVE MAP command.

Atari 130XE Note:

On the 130XE, the font you are currently editing is the only one that will be saved when you use the SAVE FONT command. This way, compatibility is maintained with virtually all other font editors.

<T> - Type

The TYPE command will allow you to enter a byte value for each row of the current character. A prompt will appear beside the first row of the editing grid, to which you type a decimal value in the range of 0 to 255, or a hexadecimal value in the range of 00 to FF, preceded by a dollar sign ("\$"). When you press <RETURN>, the first row on the editing grid will display the graphic equivalent of the value you entered, and the prompt will move down to the next row of the grid. You can just press <RETURN> without entering a value to skip to the next row. Press the <ESC> key to back up to the previous row. Pressing <ESC> when the prompt is on the first row of the editing grid will exit the TYPE command. Pressing the <RETURN> key while the prompt is on the last row of the grid (regardless of whether you entered a value or not) will also exit the TYPE command.

<U> - Undo

This command is a great convenience, as activating it will cause the current character to revert back to what it had looked like when you first started editing it (e.g., you really messed up a character trying to make it look better, so rather than start editing the character over from scratch, you just press "U", and you're back to where you were before you started messing it up! (As I'm not one of the world's greatest artists myself, I tend to use this command a lot.)

<V> - Values

When you type "V", there will be a set of numbers printed on either side of the editing grid. The left column of numbers are the byte values in decimal for each row of the current character, while the right column contains the same values, only in hexadecimal. These values will remain on the screen until a key is pressed. If the key pressed is a command key, then that command will be activated (I just use the <ESC> key, myself).

<W> - Write Font to Disk as DATA Statements

This command is the link between the fonts you design, and the programs that will use them. This command will write your font to disk in the form of data statements. The three formats available are BASIC, SynAssembler, or MAC/65. You will first be prompted as to which language you would like the font data to be generated for. This is a single key input (<S>, <M>, or) and pressing <RETURN> is not necessary. You will then be asked for the filename under which the data statements are to be written. See the LOAD FONT command (<L> from the font editor) concerning filename input. Next you will see four headings, the first of which is "Start Char.". This heading has a prompt beside it, to which you would type on the keyboard the character at which the data statements will start. The character you type will replace the prompt, and the second heading, "End Char." will be prompted. The WRITE FONT routine will stop writing data statements after it has written this character to disk (to write out data statements for the entire font, simply type the <SPACE BAR> as the start character, and the <TAB> key as the end character). The next heading will require you to enter the starting line number for the data statements. Not that this prompt has a default value of 2000: if you just press <RETURN> at this prompt, then a starting line number of 2000 will be used.

However, if you have a better number in mind, then just type it in and press return. The range is 0 to 32000, and this number must be entered in decimal only. The next prompt will require you to enter an increment for the line numbers. This prompt also has a default number which is 10: just press <RETURN> here if this value is adequate for you. Otherwise, the range for this input is 1 to 9999, and must also be entered in decimal. Now, finally, you will be asked if the parameters you have entered are all correct. The <ESC> key will abort the whole thing and exit the WRITE FONT command. If you type <N>, you will go back to the "Start Char." parameter entry for corrections. Typing <Y> means everything's just peachy and proceed to create the data statements file on drive 1. As usual, the screen will blank while writing to the disk, and will be restored when the write is complete. You can now reboot the system, load up that program you've been working on, and ENTER the font data into your program. If you look at the data statements, you will notice that each character has its own line number all to itself, consisting of the eight bytes that comprise the character. These are the same values that you see when you use the BYTE VALUES command (the <V> key from the font editor).

Atari 130XE Note:

On the 130XE, you can only write out data statements for one font at a time, and each font should be written under a different filename.

<X> - X-Copy a Character or Range of Characters

The X-COPY command allows you to duplicate other characters' images (put simply). You will first be asked whether you would like to X-COPY a single character or a range of characters. Respond by typing <C> or <R>. If you choose to copy a single character (by typing <C>), then you will simply be asked to type the character that you wish to XCOPY on the keyboard. The character you choose will then be copied into the character you are editing, and that's it.

If you choose to XCOPY a range of characters (the <R> key), then you will be asked to type on the keyboard the start character in the desired range, the end character in the range, and the start character of the area that you would like this range of characters to be copied to. If you are running ENVISION on an 800XL or 800 computer, then you will be asked if your inputs were correct, and the entire range of characters will be copied into the area you specified if you answer yes (by typing <Y>). If you are running ENVISION on a 130XE computer, then you will be asked for a destination font for this range to be copied to. The range for this prompt is <A> through to <P>. To copy within the same font, simply type the letter corresponding to the font you are currently editing. Now you will be asked if your inputs are correct. If you answer with <Y>, then the range is copied to the desired area in the desired font, and that's the end of the XCOPY command.

Non-Alphabetical Commands:

The following commands cannot be listed alphabetically, so here they are (not in alphabetical order):

<SPACE BAR> - Graphic Toggle

Note that when you press the <SPACE BAR>, the word "Graphic" appears on the left side of the first status line. This toggle is for the benefit of the map editor, and the reason is as follows: Antic modes 6 and 7 (GRAPHICS 1 and 2 in BASIC) are two-color text modes, the first color being the background color, and the second color being one of the four other color registers. The color register used is determined from the high 2 bits of the ATASCII value of the character, as shown below with the ATASCII value of the "a" character in its binary representation:

```
01100001
Normal lower case 'a'
(ATASCII=97)
```

```
01100001
Interpreted by Antic
Modes 6 & 7 as
Upper-Case 'A' Using
Color Register 1
```

So, what we are left with is an upper-case "A" using color register 1, and not a lower case "a". What this means is that only half of the character set can be used in Antic modes 6 and 7 (GRAPHICS 1 and 2). This is where the GRAPHIC TOGGLE comes in. When "Graphic" is shown on the first status line, the map editor will use the lower-case and graphics characters in these two character modes, rather than the upper-case letters and the numbers. Pressing the <SPACE BAR> again will toggle back to upper-case.

<CTRL -> - Scroll Up

This will scroll the current character image up by one row. Any part of the image that scrolls off the top of the editing grid will reappear at the bottom.

Pressing the <-> key by itself will move the font cursor up one row with wraparound.

<CTRL => - Scroll Down

Will scroll the character image down by one row. As above, any part of the image that scrolls off the bottom of the grid will reappear at the top.

Pressing the <=> key by itself will move the font cursor down one row with wraparound.

<CTRL +> - Scroll Left

Scrolls the character image left by one column. Almost like before, any part of the image that scrolls off the left side of the editing grid will reappear on the right.

Pressing the <+> key all by itself will move the font cursor left by one character with wraparound to the last character of the previous row.

<CTRL *> - Scroll Right

Scrolls the character image right by one column. And you guessed it! Any part of the image that scrolls off the right side of the editing grid will reappear on the left.

And what a coincidence! Pressing the <*> key all by itself will move the font cursor to the right by one character with wraparound to the first character of the next row.

<OPTION> - Enter Screen Editor

Pressing the <OPTION> key will take you to ENVISION's map editor, but don't worry. Pressing <OPTION> again will take you back to the font editor and nothing will have been touched. You'll even be on the same edit character as when you left!

<SELECT> - ROM Set Toggle

When you press the <SELECT> key, notice that "ROM Set" appears on the left side of the first status line. You are now looking at the Atari ROM character set. However, you may still make changes to your redefined font, as the current character will still be displayed on the editing grid and at the bottom of the display in its redefined form. Press the <SELECT> key again to go back to your redefined font. By the way, if you are in the XCOPY command, not knowing what the character you wish to XCOPY actually is because it's now a tree, the <SELECT> key will help. You can switch to the ROM set temporarily to see what character you want. This also works in the EDIT, COPY TO, and OVERLAY commands.

<START> - Restart Font Editor

This key will restart the font editor. It will wipe out the redefined font in memory (it'll wipe out all 16 fonts on the 130XE) and replace it (them) with the ROM set. It will also reset all the colors to the Atari's power-up defaults. However, you need not worry about hitting this key accidentally, as you are first prompted to confirm the devastation described above.

Map Editor Display:

This brings us to the map editor. The map editor doesn't have as many commands as the font editor, but the commands it supports are powerful indeed. Let's begin with the map editor display.

To get to the map editor from the font editor, press the <OPTION> key. The map editor display is broken into three parts. The first part at the top of the display is the map editor's status line. It tells you the following pieces of information.

The first two status elements on this line are the "X=" and "Y=" stats. They tell you the current "X" and "Y" coordinates of the map cursor (simple, so far). The coordinates of the upper-leftmost character of any map are "X=1" and "Y=1".

The next status line element is the "Char=" status. This displays the ROM image of the character that the map cursor is currently positioned over.

Next is the "Draw=" stat. This tells you which character is currently selected as the draw character. The draw character is the character that will be displayed on your map when the joystick trigger is pressed. Basically, when you move the joystick around the map pressing the button, this will be the character that will be left behind by the button.

The next stat is "KB=". This tells you the current status of the keyboard. Unlike the font editor, the keyboard is fully functional in the map editor, in that the <CAPS/LOWR> and <ATARI LOGO> (inverse toggle) key, etc. are functional. The default setting of this stat is "KB=UC". The "UC" means "Upper Case". If you press the <CAPS/LOWR> key all by itself, the stat will change to "KB=LC", meaning "Lower Case". If you type <CONTROL> <CAPS/LOWR>, it will read "KB=CC", denoting "Control Case". If you press the <ATARI LOGO> key, toggling the inverse video of characters that you type, the status reading will be displayed in inverse. Press the <ATARI LOGO> key again, and the reading will be displayed normally.

The last status element is "Mode=" . This stat simply tells you which Antic mode you are currently working with. Remember that the value displayed here represents the ANTIC mode, and not the Basic GRAPHICS mode value, since some text modes have no BASIC GRAPHICS mode number, such as ANTIC mode 3. Here is a table of the Antic modes:

<u>ANTIC Mode</u>	<u>Character Size & Description</u>
2	(8x8) normal GRAPHICS 0 text
3	(8x10) for lowercase descenders
4	(4x8) four color text
5	(4x8) same as 4 but 2x height
6	(8x8) two-color, double wide
7	(8x8) same as 6 but 2x height

The second part of the map editor display lies directly beneath the status line. This is the command window. Any command that requires keyboard input from you will use this window for prompts, etc. Upon power-up, and whenever you are not using a command, a message will be printed in this window showing you the command keys for the map editor. The commands listed are not the only functions that the map editor performs, however, but more on that later. Also in this area at the bottom is a stat that gives a "Font Assignment", and also says "GTIA:OFF". The font assignment should read "Font" on the 800/800XL machines, and "A) Font A" on the 130XE. This stat tells you which font is being used on the row of your map that the cursor is on. On the 800/800XL, you can choose either your font or the Atari ROM set for each row of your map. On the 130XE, you can use any of the 16 fonts in memory on any row of your map. On either machine, to find out which font is being used on a given row, move the cursor anywhere along that row, and it will be displayed in the "Font Assignment" stat. To change a font assignment, use the ASSIGN FONT command (available on all machines, activated by typing the <SELECT> and <A> keys simultaneously while in the map editor). See the GTIA TOGGLE command (<SELECT G>) concerning the "GTIA:OFF" status.

The third and last part of the map editor display is the actual work area for your maps. You will see a flashing cursor in this area. This is the map cursor. When you type a letter on the keyboard, it will be printed on the map right where the cursor lies, and the cursor will move to the next character position to the right. If the cursor is at the end of a line when you type a character, it will go to the beginning of the next line after it prints the character.

Map Editor Commands:

And now, here is a rundown of the commands supported by the map editor:

<SELECT A> - Assign Font

This command allows you to have more than one font on your map. On the 800/800XL machines, you can select from either your redefined font, or the Atari ROM set. On the 130XE, you can choose any of the 16 fonts, or the ROM set. When you select the ASSIGN FONT command, you will be asked for a font to assign to the row that the cursor is on. Press <R> to assign the ROM set. On the 800/800XL, press <F> to use your redefined font. On the 130XE, press a letter in the range <A> through to <P> for one of the 16 fonts. After selecting the desired font, it will instantly be used by the row of your map that the map cursor is on.

<SELECT C> - Change Parameters Menu

This command allows you to change some of the parameters under which the map editor operates. When this command is activated, you will see a command sub-menu displayed in the input area. To exit the CHANGE PARAMETERS menu, press the <ESC> key. There are four parameters that you can change:

<R> - Change Number of Map Rows

This sub-command allows you to increase or decrease the number of rows on your map. When selected, you will be informed of the current number of rows in your map. At the bottom of the input area, you are informed of the "Safe Range". The safe range is described like this:

The memory for a display cannot exceed 4K, or 4096 bytes of memory. This is because of the way the Antic chip addresses its display memory. Nevertheless, in order for the map to reside within 4K (4,096 bytes) of memory, the number of rows multiplied by the number of columns must not exceed 4096. Therefore, the safe range is given to let you know the maximum number of rows you can have before the columns have to be taken in to compensate for the extra rows.

Essentially, if you enter a value beyond the safe range given, then the columns will have to be decreased in order to keep the map within the 4K boundary. If you do this, then you will be asked if it is all right to decrease the columns, since you will lose part of your map if you do. If you do not confirm this, then you will be prompted to enter a new number of rows.

If you enter a number of rows that is less than what you currently have, you will be prompted as to whether it is all right to decrease the rows, as you once again stand to lose part of your map by doing so. Nevertheless, when you have entered the desired number of rows for your map, your map will then contain the number of rows that you specified, and the CHANGE PARAMETERS sub-menu will again be displayed. To abort the CHANGE ROWS sub-command, press <ESC> during any of its prompts.

<C> - Change Number of Map Columns

This sub-command works in much the same way as CHANGE ROWS above, but with a few extra rules. Again, you are informed of the current number of columns, along with the safe range. The slight difference is that the ABSOLUTE minimum number of columns you can EVER have is 20. Since every text mode needs at least enough characters to cover the width of the screen display, you must have at least 40 columns in Antic modes 2, 3, 4, and 5. You can never have less than 40 columns in these modes. In Antic modes 5 and 6, however, only 20 characters are required to cover the width of the screen display. Therefore, the absolute minimum number of columns you can have in these modes is 20. You can abort this command in the same way as in the CHANGE ROWS sub-command.

<S> - Change Joystick Speed

This sub-command allows you to increase or decrease the speed at which the map cursor moves when using the joystick. The default speed is 2. When selected, this sub-command will tell you the current speed setting, and will prompt you for a new speed from 1 to 3. Speed 1 is the fastest, and speed 3 is the slowest (sorry - least fast - ENVISION is not a sluggish program). Since this is a single key input, you need NOT press the <RETURN> key after making your selection. After selecting a new speed, control will go back to the sub-command menu. You may exit this command (without entering a new speed) by pressing the <ESC> key.

<M> - Change Text Mode

This final sub-command of the CHANGE PARAMETERS menu allows you to change the text mode that you are working with. When selected, you will be informed of the current Antic mode that you are working with, and you will also be prompted to select a new Antic mode from 2 to 7. Since this, too, is a single key input, pressing the <RETURN> key is not necessary.

There is a small exception to the range mentioned above. If you are currently working in Antic mode 6 or 7 and you have LESS than 40 columns on your map, the range for this command will be only 6 to 7, instead of 2 to 7. This is because there are not enough columns in your map to support the 40 column text modes 2, 3, 4, and 5. In order to switch to a 40 column text mode, you will have to first increase the number of columns in your display to at least 40 columns using the CHANGE COLUMNS subcommand described quite previously. To abort the CHANGE MODE command, press the <ESC> key.

<SELECT D> - Disk Directory

This command, when activated, will read the directory of the disk in drive 1, and display the filenames of all maps (those filenames with a ".MAP" extension) on that disk. If there are no map files on the disk, you will hear a honk, and you will be informed that there are no map files on that disk. If there are map files on the disk, then their filenames will be displayed in the input area in a two-column format. If the input area becomes filled with filenames, a "Press Any Key" message will appear, prompting you to hit a key when you are ready to see more filenames. When you do hit a key, the filename portion of the input area will scroll up one row, and new filenames (if any) will appear. When the "free sectors" message appears in the filename listing, the end of the directory will have been reached. At this point, pressing any key will exit the DIRECTORY command.

<SELECT E> - Auto-Edit Command

This command is a real dandy, especially when working on the 130XE. If you're looking at part of your map and, say, you see a character that just doesn't look right or needs to be added to, simply move the map cursor over right over top of the offending character, and press the <SELECT> and <E> keys simultaneously. You will be taken instantly to the font editor, and the font cursor will be sitting right on that same character. Not only that, but if you're running ENVISION on a 130XE, the font editor will even be showing you the font that the character is using at that position on your map! This command is a real time-saver. I sure wish it was implemented into ENVISION when I did the map called NCC1701, but alas, t'was not to be.

<SELECT F> - Box/Global Fill

Here we are at the first (alphabetically, anyway) of the BOX/GLOBAL commands. For each of the commands of this type, the first input required will be to choose either a BOX command or a GLOBAL command. Respond with or <G>. If you choose GLOBAL, then the command will affect the entire map. In this case, the FILL command, let us suppose you choose a GLOBAL FILL. The next input required is for you to input the character that is to be used as the "fill character". Do this by typing the desired character on the keyboard. If necessary, use the <CAPS/LOWR> key and the <ATARI LOGO> key in order to make your choice. Once you do this, the map will instantly fill with the character you entered, and the FILL command is complete. Before going on to the BOX FILL, let me dwell on one tiny little problem that you may encounter.

There is one specific character that cannot be typed at the "fill character" prompt (or any other single character entry prompt) by using what seems to be the proper key. The <ESC> key can be entered by simply hitting the <ESC> key at the prompt. However, you cannot enter an INVERSE <ESC> character using the <ESC> key itself - not even with the <ATARI LOGO> key. This is because the inverse <ESC> character has an ATASCII value of 155, which is the ATASCII code for the <RETURN> key. Therefore, in order to enter an inverse <ESC> character, you must hit the <RETURN> key instead of <ESC>. This is fine and dandy for character input at the command prompts, but what if you wanted to print an inverse <ESC> character on the map, rather than by using a fill (type the <RETURN> key when you're not in a command and you'll see what I mean)?

If you choose to do a BOX FILL, then you will be asked to move the cursor (with the joystick) to the upper left corner of the area that you wish to fill on the map. When you have moved the cursor there, push the joystick button, after which the character beneath the cursor (if there is one) will be highlighted. Then, you will be asked to move the cursor (again, with the joystick) to the lower right (opposite) corner of the area that you wish to fill. Move the cursor with the joystick and, when you have positioned the cursor, hit the button again. Note that when you move the joystick before you press the button that you are controlling the size and dimensions of a highlighted box, the upper left corner being where you first positioned the cursor, and the lower right corner being the cursor's present position. Also note that you cannot move the cursor above the upper left corner that you defined (else it would then be the lower left corner, but that is irrelevant), nor can you move the cursor to the left of the upper left corner that you defined. After you press the button, you will be asked to input the character to be used as the "fill character". Follow the character input directions discussed in the GLOBAL FILL command in the previous two paragraphs. Now, then, what do you do if you just simply wish to type an inverse <ESC> character on the map? The only way to do it is to execute the BOX FILL command, and, when prompted, move the cursor to the position where you wish to type the inverse <ESC>, press the button (defining the upper left corner of the box area) and then, when you are prompted for the lower right corner of the box, press the button again without moving the joystick, thus defining the upper left corner and the lower right corner to be on the same character (essentially, you have defined a box that is only as big as one character). Now, when prompted for the "fill character", press the <RETURN> key. Voila! You've printed an inverse <ESC> character on the map! You may abort the fill command at any time by pressing the <ESC> key (such irony) at any of the prompts.

<SELECT G> - GTIA Toggle

This command allows you to create maps that use the extended color power of the GTIA chip (GTIA stands for Graphics Television Interface Adapter). Some Atari 800 machines may have a CTIA chip (Color Television Interface etc.) instead of a GTIA. If so, then this command will have no effect on your maps whatsoever.

When you first select this toggle, the map will change to a GRAPHICS 9 format, which supports 16 shades of one color, the color being determined by the contents of the background register COLBAK. Press it again and you're in GRAPHICS 11, which supports 16 colors of one shade, again, the shade being determined by COLBAK. Press it one more time and you're back to a normal display. Actually, regardless of the GTIA setting, you're always in the same text mode. Using the GTIA chip on text screens is one of the least exploited features of the Atari machines, and now there's an easy way to exploit it. So... exploit it.

To do a GTIA text screen, you "should" always work in ANTIC 2 (GRAPHICS 0), although there's absolutely nothing stopping you from using any text mode you wish, but you will have problems getting nice results from another text mode, since the GTIA modes affect each of the text modes differently. The rest of this explanation will deal only with Antic mode 2 (GRAPHICS 0). When in a GTIA mode, characters suddenly become only two pixels wide instead of 8 pixels. And, as a result, these pixels are WIDE. So, a character is now 2 pixels wide and 8 pixels high. In order to set these pixels to the color that you want (you now have sixteen to choose from), you have to know a little something about the GTIA modes.

GRAPHICS 9 is a special mode in which you have 16 shades displayable at once, but they are all shades of the same color. Color 0 is the darkest of the 16 shades, and color 15 is the brightest shade (0 to 15 is 16 shades in total). The color that is used in this mode, as stated previously, is derived from the value stored in COLBAK (COLor of BACKground).

GRAPHICS 11 is much the same, only you have 16 colors displayable in one of 8 shades. Colors 0 to 15 are all set to different colors, but you cannot change these colors - only the shade that they are all displayed in. COLBAK is also the source for the shade to be used by this mode.

The command that is probably the most useful in the GTIA modes is the TYPE command in the font editor (it's just quicker, that's all). Enter each value in HEX (first type "\$"). Now, you should enter a TWO digit HEX number. The first digit will be the color number of the "left pixel", and the second digit will be the color number for the "right pixel".

<SELECT L> - Load Map from Disk

This command allows you to load up a map that has been previously saved to disk. When selected, you will be prompted to enter the filename of the map that you wish to load. Press <RETURN> after entering the filename. Make sure that the appropriate disk is in drive 1 beforehand. The screen will blank temporarily while the map is loading, and will be restored when the load is complete. Note that, when the loaded map appears, the colors that were set for that map are now set here, along with the text mode that it was created in. When you save a map file to disk, the map colors, text mode, font assignments, animation settings, and the map dimensions are saved as well, so that they are restored when you load it up later. To abort the LOAD command, press the <ESC> key when prompted for the load filename.

<SELECT M> - Box/Global Mirror

This is another of the BOX/GLOBAL commands. This command allows you to mirror a map, or any part of it, horizontally or vertically. You will first be asked as to whether you would like a BOX MIRROR, or a GLOBAL MIRROR. Respond by typing either or <G>. If you choose a GLOBAL MIRROR, then you will be prompted as to whether you wish to mirror the map horizontally or vertically. Respond by typing either <H> or <V>. A vertical mirror will cause the map to be flipped upside down, while a horizontal mirror will cause the map to flip from right to left. If you choose a Box Mirror, you must first define the box area to be affected before choosing the mirror type. See the BOX FILL command for defining a box area. As in the FILL command, you may abort the MIRROR command at any time by pressing the <ESC> key at any prompt. This command does NOT affect the font assignments on your screen.

<SELECT N> - Select New Draw Character

This command allows you to change the DRAW character for the joystick button. When you select this command, you will be asked to type the character that is to be the new DRAW character. See the FILL command concerning single character input. Once you have entered the desired character, the "Draw=" stat in the status line will be updated, completing the command. Now, when you start drawing with the joystick/button, this new character will be left behind. Since the <ESC> key is a legitimate DRAW character, there is no way to abort from this command.

<SELECT P> - Poke Color Register

This command works just like the POKE COLOR REGISTER command in the font editor.

<SELECT R> - Box/Global Replace

Yet another of the BOX/GLOBAL commands that you only seem to find in decent HI-RES drawing programs. This is a very powerful command, indeed. It allows you to replace all occurrences of a character with another character. Doesn't sound like much, but its uses are many. When selected, you are (as usual) prompted as to whether you would like a BOX REPLACE, or a GLOBAL REPLACE. Respond by typing either or <G>. If you select a BOX REPLACE, you must first define the box area for the REPLACE command to affect. See the BOX FILL command as to defining the box area.

You will now be asked to type the character that is to be replaced on the map. See the FILL command for single character input. When you have selected the character to be replaced, you will then be asked to type the character that is to replace it. After you have done this, you will finally be asked if you would like the replace to be done randomly or not. Reply with <Y> or <N> (Yes or No). If you respond by typing <N>, then all occurrences of the first character that you entered will be replaced with the second character entered. However, if you choose to have the replace done randomly by typing <Y>, then approximately 50% of all the occurrences of the first character will be replaced by the second.

You may ask, "Well, where's the sense in that?", so let me give you a little example of the power of the random replace (my favorite command!).

Suppose you are designing a map and it is to be an island of some kind. You went to the font editor and designed a tree character, and now you want to put trees on your map. But, wait! You don't want the whole map to be filled with trees, and it would be tedious to go all over the map deciding whether to put a tree there or not. So, what do you do? Well, let's assume that the map is made up of mostly space characters, and that the only non-space characters on your map are used to make up buildings, or some such item where trees do not usually grow. All you have to do is select the REPLACE command, and choose a GLOBAL REPLACE. Now, when you are asked for the character to be replaced, type the <SPACE BAR>. When you are asked for the character that is to replace the space character, type the character that you have redefined as your tree. Now you will be asked if you would like the replace to be done randomly. Answer <Y>, and TA DA! Now you have trees spread randomly all over your map in an instant.

Now (for the sake of argument), suppose that there are just too many trees on your map, and you'd like to thin out the forest a little. No problem. Simply do a GLOBAL REPLACE again. Enter the tree character as the character to be replaced, the space character as the character to replace it, and do the replace randomly (answer <Y> to the random prompt). TA DA!! Now about 50% of your trees vanish, leaving you with a thinner forest! This is called power. This is called time-saving. This is called a lot of things, but this example is called scratching the surface! As usual, pressing the <ESC> key on any prompt (except the single character entry prompts) will abort the REPLACE command. Note that fonts assignments are NOT affected by this command.

<SELECT S> - Save Map to Disk

This command allows you to save your map to disk. You will be prompted to enter a filename for your map to be saved under. See the LOAD command as to filename entry. Pressing the <ESC> key when prompted for the filename will abort the SAVE command. While saving, the screen will blank temporarily, and will be restored when the save is complete. Note that all fonts used on your screen are automatically saved to disk with your screen, so you won't have to keep loading them up every time you load your screen. Only fonts used on your map will be saved. If you have 5 fonts in memory, but only 3 of them are used on your map, then only those three will be saved with your screen. If you want the extra fonts to be saved anyway, then assign them to blank or unused rows on your map and then save your map.

<SELECT T> - Box Transfer

This command differs from the other BOX/GLOBAL commands, in that it doesn't support a GLOBAL option. This is because of what the command does. The BOX TRANSFER command allows you to define a box area on your map, and move the entire contents of that box to another place on your map. Since (what with this being a transfer) you have to have a place to move the box to, there cannot be a GLOBAL option, since there could be no destination area in which to move the entire map. Anyway, when you select this command, you will be asked to select an upper left corner, and then an upper right corner, just as the FILL and MIRROR commands do. When you have supplied these two inputs, you will then be asked to move the box area (with the joystick) to the place on your map where you would like the contents of the box to be moved. Note that, as you move the joystick around the map, the highlighted box area will move with it, with the lower right corner of the box resting on the map cursor. The box follows you around so that you may more accurately mark your destination area. When you have done this, press the joystick button. Now you will be prompted to specify as to whether you would like the transfer done as a solid transfer, or an overlay (type <S> or <O>). If you choose a solid transfer, EVERY character within the area you first defined will be moved into the destination area, and the area where those characters were moved from will be empty (filled with spaces). If you select the transfer to be done as an overlay, then only non-space characters will be moved to the destination area. That is, if there are any space characters in the area to be moved, they will not be TRANSFERred to the destination area. This option is handy, if, for example, you have a solid shape on your map, and would like to transfer a nifty border from one area on your map over top of the shape. With a solid transfer, all the characters will be moved, including the spaces inside the border shape, so doing a solid transfer would essentially wipe out the shape you're trying to put the border around. With an overlay, the space characters will not be moved, so that the result is a solid shape on your map with the border surrounding it. Only the border characters were moved. To exit the BOX TRANSFER command, press the <ESC> key at any of its prompts. Note that the font assignments on your map are NOT affected by this command.

<SELECT U> - Undo Command

This is another of the great convenience commands of the map editor. With this command supported, you are now allowed to make mistakes on your map. Supposing you wanted to fill a certain part of your map with a single character. You use the BOX FILL command, and do the fill. But wait! You've just realized (too late, you think) that you filled the wrong portion of the map, and now you've wiped out 3 buildings (or whatever data was there). No problem (again). All you have to do is activate the UNDO command, and the FILL has been completely undone, and you have your buildings back, sparing you from having to reenter that whole section of the map. Now you just redo the BOX FILL in the proper place and you're fine. The UNDO command will undo EVERYTHING SINCE AND INCLUDING THE LAST BOX OR GLOBAL COMMAND THAT WAS EXECUTED. This means that, if you do a BOX FILL, and then a BOX TRANSFER right after that, then activating the UNDO command will only undo the BOX TRANSFER. Also note that any typing on the keyboard you have done since the last BOX or GLOBAL command will also vanish, as well as any changes to the font assignments on your map, so a little caution is sometimes needed so that you don't lose a lot of typing.

Since the UNDO command's results are instantaneous, there isn't time to ESCape from this command (nor is there any point).

<SELECT W> - Write Map to Disk as DATA Statements

This command is the equivalent of the WRITE command supported by the font editor. Although the parameter input for this command is relatively the same, there are fewer prompts in this WRITE command than in that of the font editor. This command will write out your map to the disk in drive 1 in the form of data statements. The three types of data statements supported are BASIC (DATA statements), MAC/65 (.BYTE statements), and SynAssembler (.HS statements). When you select this command, you will first be asked to select a data statement type from the three just mentioned. Respond by typing "B" (BASIC), "M" (MAC/65), or "S" (SynAssembler). This is a single key input, and pressing <RETURN> is not necessary. Next, you will be asked to enter a filename under which the data statements for your map will be written. See the LOAD MAP command concerning filename input. After entering the filename and pressing <RETURN>, you will be prompted to enter the starting line number for the data statements, and the line number increment for each line of data written to the disk. See the WRITE command in the font editor for entering these two parameters, as the value range and default values are exactly the same. After these inputs, you will finally be asked if you would like to make any corrections. If you made a mistake while entering the previous parameters, then answer <N> to this prompt, and you will go back to the first prompt of the WRITE routine (choosing a data type). If the data that you entered is correct, answer <Y> to the corrections prompt, and the entire map will be written to the disk under the filename that you specified. While the data statements for your map are being written to the disk, the display will blank temporarily (as it has done with all disk access so far), and it will be restored when the write is complete. To back up through the different prompts in the WRITE command, or to abort, press the <ESC> key on any prompt except the starting line number and line number increment prompts. Note that only the map data is written to the disk, and not the font assignments or animation data. Once the data statements are written to disk, you can load up your own program that you've been writing and, using the ENTER command, you can merge these data statements into your program. If you examine the data statements themselves, you should be aware that the WRITE DATA command tries to fit as many bytes of a map row as possible into each data statement. If all of the bytes comprising one map row just won't fit, then a new data statement is started. A new data statement is ALWAYS started when the last byte of a map row is written. What this means is that your map data isn't all jumbled up into one big ball of data statements. Each row of your map begins on a new line of data. Bear in mind, though, that one row may take up two or more data statements. The easiest way to discern the beginning of a map row in the data statements is to find a line that looks like it could've had a lot more values pushed onto it. This line will mark the END of a map row. The line after it will be the beginning of the next row.

<SELECT X> - Box X-Copy

This is the last of the great box/global commands. The BOX X-COPY command works in much the same way as the BOX TRANSFER command, except that the characters in the area of the map that are to be moved are not erased when they are sent to the destination. Basically, this command allows you to make duplicates of portions of the map and put them anywhere else on your map. As with the TRANSFER command, there is no GLOBAL option, and for the same reason as the TRANSFER command. When selected, this command will first ask you to move the cursor to the upper left corner of the box area that you wish to copy. See the BOX TRANSFER command for precisely how to do this. Next, you will be asked to move the joystick to the lower right corner of the box area that you wish to copy. Again, see the TRANSFER command for specifics on this. Now, you will be asked to move the box to the area that you wish to send the copy (see TRANSFER command). Finally, you will be asked to choose whether you would like the X-COPY done as a solid COPY, or an overlay (see TRANSFER command for explanations of these two selections). Make your choice by pressing either <S> or <O>. This is a single key input, and pressing the <RETURN> key after making your selection is unnecessary. Once you have made your selection, a copy of the characters in the box you defined will be sent to the destination area that you specified. To exit the X-COPY command, press the <ESC> key at any of the prompts. To wipe out the results of the x-copy after you have done it, use the UNDO command. Note that the font assignments are NOT affected by this command.

<SELECT *> - Animation Menu

This keystroke takes you to the animation menu. It is here that you can set up animation sequences and view them at ten different speeds. An explanation of how to set up an animation sequence follows this rundown of the Animation Menu sub-commands:

<S> Set Frame Size

ENVISION supports 128 frames of animation. These frames are all the same size, but you can change this size value to whatever you want, with a maximum size of 20 characters wide by 10 characters high. If you are working in Antic mode 7, frames of this size will take up the entire visible screen. The default size is 5 by 5, and when you go to the Animation Menu, you should see a 5 by 5 highlighted box with the map cursor sitting at its lower right corner. This box is Frame 1 (indicated beneath the Animation Menu, along with the current frame size). The frame should be sitting right up in the upper left corner of the screen. If a frame has not yet been defined (see the DEFINE FRAME sub-command) then it will default to be located at coordinates 1, 1 on your map. When you change the size of the frames, the highlighted frame on the screen will shrink or enlarge accordingly. After working with the Animation Menu for a while, you'll no doubt notice that you are sometimes not allowed sizes like 20 wide by 10 high. If you are only allowed a smaller size by the SET SIZE command, then it is because you have a frame on your map that is sitting too close to the right or bottom edge of your map. Increasing the frame size to anything larger than the maximum size given would cause the frame to extend beyond the boundaries of your map. To remedy this situation, you can either increase the size of your map, or, if that's not possible, you can move the frame that's too close to the edge of your map away from the edge a bit. Press <ESC> to exit this sub-command.

<D> Define Frame

This sub-command allows you to define a given frame to be located wherever you want on your map. You will first be asked which frame you would like to define. The range is 1-128. Next, you will be asked to move the highlighted box over top of the area on your map that is to be that frame. When you have done so, hit the joystick button. You will hear a double click when you hit the button, indicating that the frame is now defined. It is perfectly alright to have many frames defined at the same position on your map. In other words, it's OK to use the same frame data more than once (and usually very time-saving).

<C> Clear Frame

This sub-command will reset all frame definitions back to coordinates 1, 1. It will also reset the frame size back to 5 by 5. You will first be asked to confirm this, so don't worry about hitting this key accidentally.

<G> Goto Frame

This sub-command allows you to send the highlighted frame box to any of the 128 frames. You will be asked for a frame number to go to. This command is basically an easy way to locate frame data on your map. After entering the desired frame number, the chosen frame will be highlighted and you will be taken back to the Animation Menu.

<A> Animate Frames

This sub-command lets you actually see your animation sequence in motion. You will first be asked for a "Start Frame" for the sequence to be animated. The range is 1-127. Next, you will be asked for an "End Frame". The range for the end frame prompt is all frames greater than the first frame value. You will then be asked for an initial animation speed. You can enter any speed (the range is 0-9). It really doesn't matter, since you can change the animation speed while your sequence is animating. Finally, you will be asked to move the highlighted box to the area on your map where you would like the animation to... well... animate. When you have positioned the box on your map, press the joystick button. Now you will see your animation sequence in action! Note that when the end frame has been displayed, the animation will repeat with the first frame again, over and over, until you hit either the <ESC> key or the joystick button to exit the animation command. Note also that you can still move the map cursor around on the map while your animation is running. Also note that font assignments are displayed as well (I swear this documentation has more "notes" than Sostenuto).

Setting Up an Animation Sequence:

Creating animation with ENVISION is really quite simple, since only five commands are necessary to set one up! You must first realize that ENVISION is a text editor, so it naturally follows that animation sequences are comprised of text. The map editor essentially serves two purposes. The first purpose is for designing maps and title pages and the like. The second purpose is using it as a large "worksheet" or "scratchpad" for animation sequences.

To set up an animation sequence, you first need a set of pictures to flip through - these are called frames. This is what you use the map editor for if you're doing animation with it. Say you want to do an animation sequence of a man walking (we could do a man running, but we don't want to move along too quickly, do we? Ha ha). Well, the first thing you need to do is find out how many frames you need. Suppose you decide that you need 6 frames. Begin with the first frame. Using the font editor, define a set of characters and put them in place somewhere on your map to form a picture of a man in 1 of the 6 positions for the sequence. Now redefine some more characters to make up the second frame (that being a picture of the same man a little further into his "step", and put it somewhere else on the map, it doesn't really matter where, as long as you've got room for all 6, which should be no problem (if you run out of room, just add a few columns or rows to your map). When you've created all 6 pictures of your man in different positions of a walking sequence, go to the Animation Menu by typing the <SELECT> and <*> keys together. Press <D> for the DEFINE FRAME sub-command, and enter <1> and then <RETURN> to define frame one. Now move the highlighted box over top of the first frame that you created, and hit the joystick button. Now, press <D> again, and define frame <2> by moving the box over the second frame that you created and hit the button. Do this for all six pictures that you made, so that the first six frames of the 128 frames supported are defined for your sequence. Now all you have to do is press <A> for the ANIMATE FRAMES sub-command, enter <1> for the start frame, enter <6> for the end frame, and enter an initial animation speed from 0-9. Now move the highlighted box to the area on your map that you'd like it to animate and hit the joystick button. Ta Da! You're now seeing your animation in action right before your eyes. To get results right away without a lot of work, just create a bunch of frames made up of simple ROM set characters (the Atari graphics characters are a lot of fun to play with), rather than redefining half of a character set or more, which takes time. As you familiarize yourself with the way animation sequences are set up with ENVISION, you will probably want to try for more and more detail. Take a look at the demo files included with ENVISION to get the hang of it as well.

About the Map Editor Keyboard:

While in the map editor, the keyboard will function almost exactly as it does when you are in the Atari memo pad mode, with a few exceptions (I did say "almost"). The really data-damaging keys, like <SHIFT> <CLEAR>, and <CONTROL> <CLEAR> have been disabled. Now they will only print the "clear" symbol, a lopsided up-arrow. After all, it wouldn't be nice to suddenly have all your work wiped out with a single keystroke (although you could get it back with the UNDO command). Besides, the RESTART MAP EDITOR command key (<START>) is well out of the way, and requires confirmation for clearing the map, making it a practical alternative. The <CONTROL> <2> keypress combination no longer sounds the keyboard bell, but instead treats the keypress combination as if you had previously hit the <ESC> key, printing the "bell" character. Here are the useful editing keys that are enabled while in the map editor, and a brief explanation of the results of using them:

<SHIFT-INSERT> - Insert Blank Row

When you type this keypress combination, all rows on your map below and including the row that the cursor occupies will move down by one row. The row that the cursor is on will now be blank. Any characters that were on the last row of your map will be lost (but not to the UNDO command). The cursor position on your map will not be altered whatsoever. Font assignments WILL be affected by this key, since ALL data on the affected rows is moved down, it follows that the font assignments should be moved as well. Font A will automatically be assigned to the blank row that results from the insert.

<CONTROL-INSERT> - Insert Blank Character

Typing this key combination will affect only the row that the cursor occupies. All characters to the right of and including the character that the cursor rests on will be shifted one character to the right. The character space that the cursor is on will then be blank (actually, a space character). The last character on the row that the cursor is on will be lost. Again, the cursor's position will not be altered as a result. Font assignments are NOT affected by this key.

<SHIFT-DELETE> - Delete Row

This key combination is the opposite of the <SHIFT> <INSERT> keypress. Typing <SHIFT> <DELETE> will cause all rows below but not including the cursor row will be moved up by one row. The last row of the map will now be blank, and the row that was below the cursor's row now occupies the cursor's row. As with the previous two keypresses, the cursor position is not altered. As with <SHIFT> <INSERT>, font assignments are moved with the affected rows, and font A will be automatically assigned to the last row of the map.

<CONTROL-DELETE> - Delete Character

Typing this keypress combination will affect only the row that the cursor is on. When you type this keypress, all characters to the right of (but not including) the cursor character will move one character to the left. The character that was directly to the right of the cursor now lies beneath the cursor, and the last character on that row will be blank (in reality, a space character). As with <SHIFT> <DELETE>, the cursor's position is not altered as a result of using this keypress combination. Font assignments are NOT affected by this key.

<RETURN> - Carriage Return

When you press the <RETURN> key, the map cursor will jump to the first character of the next row on the map. If you press this key while the cursor lies anywhere along the last row of the map, you will hear a "ding" sound, and the cursor will jump to the first character of the first row of your map. The "ding" sound is to let you know that the cursor has wrapped around back to the top of the map (sort of like the way a typewriter will "ding" you when you have reached the end of a line on the page).

The following keys all function normally (i.e., they function in the same way as they do in Atari BASIC):

<CAPS/LOWR>
<SHIFT-CAPS/LOWR>
<CONTROL-CAPS/LOWR>
<TAB>
<CONTROL -> (cursor up)
<CONTROL => (cursor down)
<CONTROL +> (cursor left)
<CONTROL *> (cursor right)
<ESC>
<ATARI KEY> (inverse toggle)

The following keys DO NOT work and are disabled:

<CONTROL-1>
<CONTROL-3>
<SHIFT-TAB>
<CONTROL-TAB>
<BREAK>

Also, typing <CONTROL-2>, <CONTROL-CLEAR>, or <SHIFT-CLEAR> will result in their respective characters to be printed to the map, As if the character had been preceded by an <ESC>.

Console Keys:

<START> - Restart Map Editor

Pressing the START key initiates the RESTART command. You will be prompted to confirm the resetting of the map editor. Typing <N> to this prompt will exit the RESTART command and go back, leaving everything untouched. If you type <Y>, then the map editor will restart, with the following results:

- 1] Your map will be erased completely from memory, so make sure you have previously saved it to disk. Also, all font assignments and animation definitions will be wiped out.
- 2] The map will revert back to the powerup defaults. The map dimensions will be restored to 40 columns by 24 rows, and the map mode will revert back to Antic mode 2.
- 3] The cursor position is reset to 1,1 (the first character of the first row). Also, the DRAW character for the joystick button will be restored to the "A" character.
- 4] Lastly, the cursor speed will be reset to speed 2.

<SELECT> - Activate Command Keys

As you have probably guessed by now, the <SELECT> key is used to activate the command keys on the keyboard. If, for example, you wanted to type an "X" on the map, you would simply type the <X> on the keyboard. However, if you wanted to activate the BOX X-COPY command, you would hold down the <SELECT> key while pressing the <X> key. The <SELECT> key is what differentiates between interpreting the <X> key as a character to be printed to the map, and a command key that activates the BOX X-COPY command. You need only use the <SELECT> key for the command keys listed in the input window. The <SELECT> key is not needed in order to use keyboard editing commands like <SHIFT> <DELETE> and the such.

<OPTION> - Map Editor/Font Editor Toggle

If you hit the OPTION key while in the map editor, control will be sent to the font editor, and its display will appear. Do not worry that your map has been erased or changed in any way when you press the OPTION key. ENVISION's font editor and map editor are completely independent of one another (although they work with each other hand-in-hand - neat, huh?). If you press the <OPTION> key again, you will be back in the map editor, and nothing will have been changed (unless you edited a character or two while in the font editor). Also note that being in the map editor does no damage to any fonts currently in memory whatsoever.

Tips for Beginning Programmers:

First of all, after you ENTER the data statements for a font into your program, you must still POKE these values into an area of your computer's memory. An entire font takes up 1024 bytes or 1K of memory. The computer's memory is divided up into "pages", and each page is 256 bytes in length; exactly one quarter of 1024 bytes. Therefore, a font will use up four pages of memory.

To tell the computer to display characters using your redefined font, you must tell it where in memory your font is. The problem is that the computer doesn't exactly want to know the memory address of the beginning of your font; it wants to know the "page number" that your font starts at. So, the beginning of your font must start EXACTLY at the beginning of a page. This limitation is called a "one-page boundary". But wait, there's more.

The computer not only wants your font to start at the beginning of a page, it wants your font to start on a "four-page boundary", also called a "1K boundary".

Don't let all these terms confuse you. A simple way of explaining it is that there are 256 "pages" in your computer's memory (256 pages times 256 bytes in each page gives 65536, or 64K of total memory). Some of these pages are reserved for use by the computer itself, but you can use most of them for your own stuff. A font must begin on an even page boundary that is divisible by 4, such as page 4, page 8, or page 64.

Let's say, for example, that you want to put your font on page 128 (a safe place for now). Well, that means you have to know the precise memory address that page 128 begins on. Well, all you have to do to get this address is to multiply the page number (128) by the number of bytes in a single page (256), and you get the address (32768).

Now that you know the address in memory of where to put your font, you have to go and, well, put it there. A simple BASIC routine to put an entire font in memory at page 128 looks like this:

```
10 PAGE=128
20 ADDRESS=PAGE * 256
30 FOR C=0 TO 1023
40 READ A
50 POKE ADDRESS+C,A
60 NEXT C
```

Pretty simple, eh? If you want to use a different page, just change the number "128" in line 10 to the desired page number. Notice that the loop range in line 30 is 0 to 1023 instead of 1 to 1024. We start at zero so that the first byte of data is read in at 32768+0 instead of 32768+1, which is one byte too far into the page. Of course, to use this little routine, you'll have to have the data statements to go along with it.

Now then, all that's left to do is to tell the computer to start displaying your font instead of the dull and uninteresting ROM font. To do this, you simply have to POKE the starting page number of your font (128 in our example) into memory location 756. So our little routine now looks like this:

```
10 PAGE=128
20 ADDRESS=PAGE * 256
30 FOR C=0 TO 1023
40 READ A
50 POKE ADDRESS+C,A
60 NEXT C
70 POKE 756 ,PAGE
```

There you go! That's it. Create data statements for your favorite font with ENVISION, type in this little routine and ENTER the data statements into memory to become part of the routine. Now run it and watch! A bit of a pause and then POOF! There's your font right before your eyes. Go ahead. List the program and look at it shown in the redefined font. Great stuff!

One last note: if you put this routine into a larger program, then you must protect the memory used by the font set and be careful about the GRAPHICS command. Any GRAPHICS command will point the computer back to the ROM set. You will have to POKE the page number of your font back into location 756 again.