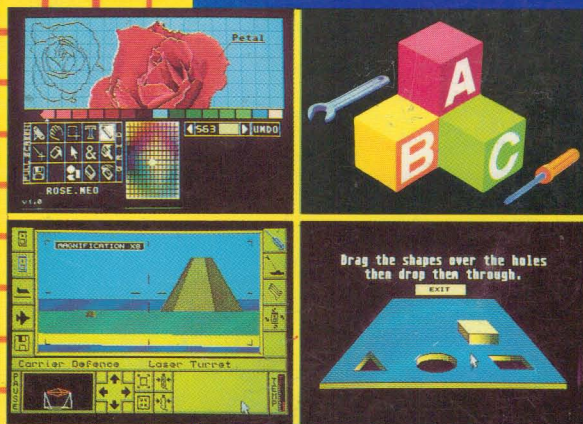


# DISCOVERY PACK

3

NEOCHROME  
FIRST BASIC  
ST TOUR  
UTILITIES  
DESK ACCESSORIES  
AND GAMES  
INSTRUCTIONS



BY HI-SOFT AND ATARI

ATARI



# DISCOVERY PACK

3

NEOCHROME  
FIRST BASIC  
ST TOUR  
UTILITIES  
DESK ACCESSORIES  
AND GAMES  
INSTRUCTIONS

BY HI-SOFT AND ATARI

 ATARI

# IMPORTANT INFORMATION

NEOchrome was designed and programmed by Dave Staugas; special effects by Jim Eisenstein; graphics by Jerome Domurat. Box illustration is used by permission of Atari Games Corporation, Copyright© 1985 Atari Games Corporation. All Rights Reserved. Demonstration .NEO files used by permission of Atari Games and Alan Murphy.

FirST BASIC was designed and programmed by HiSoft Computer Software. Thanks go to Stephan Somogyi, David Nutkins, Andy Pennell, Dave Howorth, Simon Goodwin, Sue, Julie, Natalie and the girl with the dog for their invaluable help in the production of FirST BASIC, Power BASIC and HiSoft BASIC.

ST Tour was designed and programmed by Craig Suko of Atari Corporation.

Many thanks to ST Club for their assistance in selecting and providing a variety of Desk Accessories for inclusion in this pack.

Every effort has been made to ensure that the contents of this manual are correct. However, because it is constantly improving and updating its software and hardware, Atari Corporation cannot guarantee the accuracy of printed material after the date of publication and disclaims liability for changes, errors, or omissions.

No part of this publication may be reproduced or transmitted in any form or by any means, including photocopying and recording, without the prior written permission of Atari Corporation. Such written permission must also be obtained before any part of this publication is stored on a retrieval system of any nature.

It is an infringement of the copyright pertaining to the products covered by this manual and their associated documentation to copy, by any means whatsoever, any of the Discovery Pack disks for any reason other than for the purposes of making a security back-up copy of the object code as detailed in this guide and the Atari ST Owner's Manual.

ATARI, MEGA, ST Tour, NEOchrome, ST, and TOS are trademarks or registered trademarks of Atari Corporation. GEM is a trademark of Digital Research, Inc. FirST BASIC, Power BASIC and HiSoft BASIC are trademarks of HiSoft. Space Harrier and Outrun© 1985 Sega. Both manufactured under license from Sega Enterprises Ltd., Japan. Space Harrier™, Outrun™ and Sega® are trademarks of Sega Enterprises Ltd. Bomb Jack™ Copyright 1985 TECMO Ltd. Copyright© 1988 Elite Systems International Ltd. Outrun is manufactured under license from Sega Enterprises Ltd., Japan. Carrier Command© 1988 Realtime Games Software. Manufactured under license from Rainbird Software.



Copyright© 1986,1990 Atari Corporation, Sunnyvale, CA 94086. All Rights Reserved.  
Portions© 1989, 1990 HiSoft. All Rights Reserved  
Printed in the United Kingdom.

# MULTI-MANUAL CONTENTS

<b>FirST BASIC</b>	1
Introduction	3
Editor	7
Entering Text	7
Saving Text	11
Loading Text	12
Running Programs	15
Quick Reference Section	17
<b>NEOchrome</b>	35
Introduction	37
Painting a Picture	34
Saving and Loading	43
Printing Pictures	45
Running NEOshow	46
NEOchrome Components	47
Tools for Moving the Canvas	60
Special Tools	67
NEOchrome Techniques	77
System and Extended System Character Sets	103
.NEO File Format	107
Glossary	109
<b>ST TOUR</b>	115
Introduction	117
<b>Desk Accessories</b>	119
Introduction	121
<b>Games Instructions</b>	125
OUTRUN	127
SPACE HARRIER	129
CARRIER COMMAND	131
BOMB JACK	137





**ATARI<sup>®</sup>**  
**FirST BASIC<sup>™</sup>**

Programming Language  
for all Atari ST<sup>™</sup> and MEGA<sup>™</sup> Computers

# ATARI

## FIRST BASIC

Programming Language  
for the Atari 2600 and Atari 5200 Computer



# INTRODUCTION

## Introduction to FirST BASIC

Welcome to **FirST BASIC** from HiSoft for the Atari ST range of computers - a powerful and modern version of the BASIC language that gives you a totally integrated and interactive environment for the production of your programs. **FirST BASIC** is one of a range of HiSoft BASIC compilers for the ST and all are the result of many years design and programming effort during which our goal was to produce a BASIC system with the following features:

- modern, structured dialect of the BASIC language
- interactive edit/compile/run cycle, like an interpreter
- compile Microsoft QuickBASIC 3™ with minimal change
- compile most flavours of BASIC with little modification
- fast compile time and very fast execution time
- full recursive sub-programs and functions
- many structured statements like WHILE...END WHILE, DO...LOOP UNTIL, SELECT...CASE etc.
- full support for the Atari ST and GEM through use of libraries
- clear error reporting and correction
- no limits on variable size

Thanks to the power and flexibility of the Atari ST and its operating system, we have been able to implement all of these design goals and we hope you will enjoy the power, flexibility and portability of **FirST BASIC**.

## System Requirements

The minimum requirements for running **FirST BASIC** on an Atari ST are:

- 512k bytes of RAM
- a disk drive
- a mouse

## Scope of this Guide

This guide to **FirST BASIC** is predominantly a reference guide, intended to allow experienced BASIC programmers to start using the package quickly. If you are new to BASIC programming or would like to get the most out of your ST and BASIC, we recommend the purchase of the book **Your FirST BASIC** which includes complete details of every aspect of the package together with an extensive tutorial that not only teaches you how to use a modern structured language but also gives away many secrets of the Atari ST. Details of how to obtain your copy of this invaluable book can be found at the end of this section.

## Getting Started with FirST BASIC

This section is going to take you through the stages of writing, compiling and running programs in **FirST BASIC**. It will concentrate on how to use the editor and compiler so that you can quickly teach yourself how to use this fast, interactive programming environment. We will not try to teach BASIC programming in this chapter, it is solely intended to get you going with **FirST BASIC**; remember that the book **Your FirST BASIC** is available in good shops or directly from HiSoft.

## Your FirST Programs

Insert the **Discovery Pack Language Disk** and switch on the computer. When a directory window appears double-click on the `1STBASIC.PRG` icon and wait until the editor window appears; this is a normal GEM window with a grow box at the bottom right, a close box at the top left and horizontal and vertical scroll bars. Now type in this 8-line program pressing the Return key at the end of each line:

```
t=TIMER
DO WHILE TIMER<t+20
  col=RND*17+1 : row=RND*16+2
  IF MOUSE(2) THEN
    CLS : BEEP : LOCATE row,col
    PRINT "HiSoft FirST BASIC"
  END IF
LOOP
```

It doesn't matter if keywords like `TIMER`, `DO`, `LOOP` etc. are in upper-, lower- or mixed-case; it is good programming style to put them in upper-case but **FirST BASIC** doesn't care.

This is a very simple little program that randomly chooses a screen position suitable for all screen resolutions (`col=RND*17+1 : row=RND*16+2`), waits for you to click on the left mouse button (`IF MOUSE(2)`) and then clears the window, beeps and prints a message at the random point (`CLS etc.`). After 20 seconds it times out and finishes (`DO WHILE TIMER<t+20 LOOP`).

Ok, you want to run it? Press **[Alternate][X]** by which we mean hold the Alternate key down and press X (lower- or upper-case X). There's another way you can run a program - choose Run from the Program menu. However you have chosen to run the program, you will see a window with the title **FirST BASIC** is working.

If you have an error in your program then the compiler will tell you about it by means of an error number and an error message. It will take you back to the editor screen at the position on the line where the error occurred and with the error message displayed in the top right of the window so that you can correct the error and run the program again.

When the program has compiled successfully, the program will execute ... click a mouse button and see how easy it is to spread the word! After 20 seconds you will be prompted to press a key, then you will be returned to the editor.

That's how easy it is to write a windowed GEM program! **FirST BASIC** automatically opens up a window for your program and closes it down when the program finishes.

One final point on this simple program - did you notice the DO statement? This is a very flexible structure that allows you to do many different types of loops with one statement.

Let's type in a more interesting program. Select **New** from the **File** menu: - a box will appear saying **OK to lose changes?**, click on the **OK** button. Type in this, pressing the **[Return]** key at the end of each line:

```

medhigh=4-PEEKW(SYSTAB)' screen resolution
lowmed=1-PEEKW(SYSTAB)
MOUSE -1'turn mouse off
Patterns:
  CLS
  RANDOMIZE TIMER
  IF medhigh THEN cx=300 ELSE cx=150
  IF lowmed THEN cy=95 ELSE cy=190
  pi=3.14159265#
  xs=cx/4+cx*RND/2 : ys=cy/4+cy*RND/2
  FOR k=0 TO 500 STEP 4*pi*RND+1
    tx=x : ty=y
    x=cx+xs*COS(k) : y=cy+ys*SIN(k)
    IF k=0 THEN PSET(x,y) ELSE LINEF tx,ty,x,y
  NEXT k
GOTO Pattern

```

Press **[Alternate][X]** to run your program. If you typed it in exactly as above **FirST BASIC** should stop showing an error: Pattern is not a label ... on line 16. Can you spot the deliberate mistake? That's right, Pattern should be Patterns (plural) because that was the label defined on line 4 of the program. So change Pattern to Patterns by typing an s at the end of the line. Press **[Alternate][X]** to run the program again - this time it should run without error; fast, isn't it?

The program repeats itself indefinitely and you may begin to get bored after a while! So press **[Shift][Alternate][Help]**; this is a way of interrupting **FirST BASIC** programs when no text is being written to the screen; you will see the message:

```
BREAK pressed at line x
```

Pressing any key will return you to the editor once more.

When text is being output to the screen you can interrupt a program by pressing **[Ctrl][C]** and this will have the same effect as pressing **[Shift][Alternate][Help]**. This is an example of a run-time error, and as you can see, it is reported exactly, allowing easy bug finding.

## What's on the Disk

**FirST BASIC** is supplied on the **Discovery Pack Language Disk**.

Amongst the contents of the Language Disk you will find at the following files:

1STBASIC.PRG	The integrated editor, compiler and library
DEMO.BAS	Demonstration program
JACK.SCR	High resolution screen shot for DEMO
JACKMED.S	CRAs above for medium resolution
DUMP.BAS	Hex and ASCII file dumper
HANOI.BAS	Towers of Hanoi program
SIEVE.BAS	BYTE Sieve benchmark

Only **1STBASIC.PRG** need be present on your work disks; the presence of all the other files is optional.

## The FirST BASIC Editor

The editor section of FirST BASIC is a screen editor which allows you to enter and edit text and save and load from disk, as you would expect. It also lets you print some or all of your text, search and replace text patterns and use any of the ST's desk-accessories. It is GEM-based, which means it uses all the user-friendly features of GEM programs that you have become familiar with on your computer such as windows, menus and mice. However, if you're a die-hard used to the hostile world of computers before the advent of WIMPs, you'll be pleased to know you can do practically everything you'll want to do from the keyboard without having to touch a mouse.

The editor is 'RAM-based', which means that the file you are editing stays in memory for the whole time, so you don't have to wait while your disk grinds away loading different sections of the file as you edit. As the ST range has so much memory, the size limitations often found in older computer editors don't exist with FirST BASIC; if you have enough memory you can edit files of over 300k (though make sure your disk is large enough to cope with saving it if you do!). As all editing operations, including things like searching, are RAM-based they act blindingly quickly.

If at any time you get stuck, pressing the Help key will bring up a display of the keys required for functions not visible in any menus.

## Entering text and Moving the cursor

Having loaded **FirST BASIC**, you will be presented with an empty window with a status line at the top and a flashing black block, which is the cursor, in the top left-hand corner.

The status line contains information about the cursor position in the form of Line and Column offsets as well as the number of bytes of memory which are free to store your text. Initially this is displayed as 9980, as the default text size is 10000 bytes. You may change this default if you wish, together with various other options, by selecting Preferences, described later. The rest of the status line area is used for error messages, which will usually be accompanied by a 'ping' noise to alert you. Any message that gets printed will be removed when subsequently you press a key.

To enter text, you type on the keyboard. As you press a key it will be shown on the screen and the cursor will be advanced along the line. If you are a very good typist you may be able to type faster than the editor can re-display the line; if so, don't worry, as the program will not lose the keystrokes and will catch up when you pause. At the end of each line you press the **[Return]** key (or the **[Enter]** key on the numeric pad) to start the next line. You can correct your mistakes by pressing the **[Backspace]** key, which deletes the character to the left of the cursor, or the **[Delete]** key, which removes the character the cursor is over.

## Cursor keys

To move the cursor around the text to correct errors or enter new characters, you use the cursor keys, labelled ← → ↑ ↓. If you move the cursor past the right-hand end of the line this won't add anything to your text, but if you try to type some text at that point the editor will automatically add the text to the real end of the line. If you type in long lines the window display will scroll sideways if required.

If you cursor up at the top of a window the display will either scroll down if there is a previous line, or print the message `Top of file` in the status line. Similarly if you cursor down off the bottom of the window the display will either scroll up if there is a following line, or print the message `End of file`.

You can move the cursor on a character basis by clicking on the arrow boxes at the end of the horizontal and vertical scroll bars.

For those of you used to WordStar, the keys **[Ctrl][S]**, **[Ctrl][D]**, **[Ctrl][E]** and **[Ctrl][X]** work in the same way as the cursor keys.

To move immediately to the start of the current line, press **[Ctrl][←]**, and to move to the end of the current line press **[Ctrl][→]**.

To move the cursor a word to the left, press **[Shift][←]** and to move a word to the right press **[Shift][→]**. You cannot move past the end of a line with **[Shift][→]**. A word is defined as anything surrounded by a space, a tab or a start or end of line. The keys **[Ctrl][A]** and **[Ctrl][F]** also move the cursor left and right on a word basis.

To move the cursor a page up, you can click on the upper grey part of the vertical scroll bar, or press **[Ctrl][R]** or **[Shift][↑]**. To move the cursor a page down, you can click on the lower grey part of the scroll bar, or press **[Ctrl][C]** or **[Shift][↓]**.

If you want to move the cursor to a specific position on the screen you may move the mouse pointer to the required place and click (There is no WordStar equivalent for this feature!).

### **Tab key**

The **[Tab]** key inserts a special character (ASCII code 9) into the buffer, which on the screen looks like a number of spaces, but is rather different. Pressing Tab aligns the cursor onto the next 'multiple of 4' column, so if you press it at the start of a line (column 1) the cursor moves to the next multiple of 4, +1, which is column 5. Tabs are very useful indeed for making items line up vertically and its main use in **FIRST BASIC** is for such things as indenting structured program lines.

### **Backspace key**

The **[Backspace]** key removes the character to the left of the cursor. If you backspace at the very beginning of a line it will remove the 'invisible' carriage return and join the line to the end of the previous line. Backspacing when the cursor is past the end of the line will delete the last character on the line, unless the line is empty in which case it will re-position the cursor on the left of the screen.

### **Delete key**

The **[Delete]** key removes the character under the cursor and has no effect if the cursor is past the end of the current line.

## Goto a particular line

To move the cursor to a specific line in the text, click on `Goto line...` from the `Options` menu, or press **[Alternate][G]**. A dialog box will appear, allowing you to enter the required line number. Press **[Return]** or click in the `OK` button to go to the line or click on `Cancel` to abort the operation. After clicking on `OK` the cursor will move to the specified line, re-displaying if necessary, or give the error `End of file` if the line doesn't exist.

Another fast way of moving around the file is by dragging the slider on the vertical scroll bar, which works in the usual GEM fashion.

## Goto top of file

To move to the top of the text, click on `Goto Top` from the `Options` menu, or press **[Alternate][T]**. The screen will be re-drawn if required starting from line 1.

## Goto end of file

To move the cursor to the start of the very last line of the text, click on `Goto Bottom`, or press **[Alternate][B]**.

## Quitting FirST BASIC

To leave **FirST BASIC**, click on `Quit` from the `File` menu, or press **[Alternate][Q]**. If changes have been made to the text which have not been saved to disk, an alert box will appear asking for confirmation. Clicking on `Cancel` will return you to the editor, while clicking on `OK` will discard the changes and return you to the Desktop.

## Deleting text

### Delete line

The current line can be deleted from the text by pressing **[Ctrl][Y]**.

### Delete to end of line

The text from the cursor position to the end of the current line can be deleted by pressing **[Ctrl][Q]**. (This is equivalent to the WordStar sequence **[Ctrl][Q][Y]**).



## UnDelete Line

When a line is deleted using either of the above commands it is preserved in an internal buffer, and can be re-inserted into the text by pressing **[Ctrl][U]**, or the **[Undo]** key. This can be done as many times as required, particularly useful for repeating similar lines or swapping individual lines over.

## Delete all the text

To clear out the current text, click on **New** from the **File** menu. If you have made any changes to the text that have not been saved onto disk, a confirmation is required and the requisite alert box will appear. Clicking on **OK** will delete the text, or **Cancel** will abort the operation.

## Saving Text

To save the text you are currently editing, click on **Save As** from the **File** menu, or press **[Alternate][S]**. The standard GEM File Selector will appear, allowing you to select a suitable disk and filename. Clicking **OK** or pressing **Return** will then save the file onto the disk. If an error occurs a dialog will appear showing a TOS error number.

If you click on **Cancel** the text will not be saved. Normally if a file exists with the same name it will be deleted and replaced with the new version, but if **Backups** are selected from the **Preferences** options then any existing file will be renamed with the extension **.BAK** (deleting any existing **.BAK** file) before the new version is saved.

## Save

If you have already done a **Save As** (or a **Load**), **FirST BASIC** will remember the name of the file and display it in the title bar of the window. If you want to save it without having to bother with the file selector, you can click on **Save** on the **File** menu and it will use the old name and save it as above. If you try to **Save** without having previously specified a filename it will present you with the File Selector, as in **Save As**.

## Loading Text

To load in a new text file, click on `Load` from the `File` menu, or press **[Alternate][L]**. If you have made any changes that have not been saved, a confirmation will be required. The GEM file selector will appear, allowing you to specify the disk and filename. Assuming you do not `Cancel`, the editor will attempt to load the file. If it will fit, the file is loaded into memory and the window is re-drawn. If it will not fit an alert box will appear warning you, and you should use `Preferences` to make the edit buffer size larger, then try to load it again.

## Searching and Replacing Text

To find a particular section of text click on `Find` from the `Search` menu, or press **[Alternate][F]**. A dialog box will appear, allowing you to enter the Find and Replace strings. If you click on `Cancel`, no action will be taken; if you click `Next` (or press `[Return]`) the search will start forwards, while clicking on `Previous` will start the search backwards. If you do not wish to replace, leave the Replace string empty. If the search was successful, the screen will be re-drawn at that point with the cursor positioned at the start of the string. If the search string could not be found, the message `Not found` will appear in the status area and the cursor will remain unmoved. By default the search is always case-independent, if you click on the `UPPER & lower case Different` button the search will be case dependent.

To find the next occurrence of the string click on `Find Next` from the `Search` menu, or press **[Alternate][N]**. The search starts at the position just past the cursor.

To search for the previous occurrence of the string click on `Find Previous` from the `Search` menu, or press **[Alternate][P]**. The search starts at the position just before the cursor.

Having found an occurrence of the required text, it can be replaced with the `Replace` string by clicking on `Replace` from the `Search` menu, or by pressing **[Alternate][R]**. Having replaced it, the editor will then search for the next occurrence.

If you wish to replace every occurrence of the Find string with the Replace string from the cursor position onwards, click on `Replace All` from the `Search` menu. During the global replace the **[Esc]** key can be used to abort and the status area will show how many replacements were made.

## Block Commands

A Block is a marked section of text which may be copied to another section, deleted, printed or saved onto disk. The function keys are used to control blocks.

### Marking a block

The start of a block is marked by moving the cursor to the required place and pressing key **[F1]**. The end of a block is marked by moving the cursor and pressing key **[F2]**. The start and end of a block do not have to be marked in a specific order - if it is more convenient you may mark the end of the block first.

### Saving a block

Once a block has been marked, it can be saved by pressing key **[F3]**. Assuming a valid block has been marked, the standard GEM file selector will appear, allowing you to select a suitable disk and filename. If you save the block with a name that already exists the old version will be overwritten - no backups are made with this command.

### Copying a block

A marked block may be copied, memory permitting, to another part of the text by moving the cursor to where you want the block copied and pressing key **[F4]**. If you try to copy a block into a part of itself, the message `Invalid block!` will appear and the copy will be aborted.

### Deleting a block

A marked block may be deleted from the text by pressing **[Shift][F5]**. The shift key is deliberately required to prevent it being used accidentally. Block markers remain during all editing commands, moving where necessary, and are only reset by the commands `New`, `Delete block`, and `Load`. If you edit a line containing block markers the new position of the markers is not defined.

## Miscellaneous Commands

### About FirST BASIC

If you click on `FirST BASIC...` from the `Desk` menu, a dialog box will appear giving various details about **FirST BASIC**, including the version number. Pressing **[Return]** or clicking on `OK` will return you to the editor.

## Help Screen

The key equivalents for the commands not found in menus can be seen by pressing the **[Help]** key, or **[Alternate][H]**. A dialog box will appear showing the WordStar and function keys, as well as the free memory left for the system.

## Preferences

Selecting **Preferences** from the **Options** menu will produce a dialog box allowing you to change several editor settings:

### Tabs

By default, the tab setting is 4, but this may be changed to any value from 2 to 16.

### Text Buffer Size

By default the text buffer size is 10000 bytes, but this can be changed from 4000 to 999000 bytes. This determines the largest file size that can be loaded and edited. Care should be taken to leave sufficient room in memory for compilations - pressing the **Help** key displays free system memory, and for compilations this should always be at least 100k bytes. Changing the editor workspace size will cause any text you are currently editing to be lost, so a confirmation is required if it has not been saved.

### Maximum Size

This defaults to 30k and is the size of a buffer used by **FirST BASIC** to compile your program. It should only be increased in size if you get a **Code generation failed** error when you try to run your program.

### Numeric Pad

The **Numeric Pad** option allows the use of the numeric keypad in an IBM-PC like way allowing single key presses for cursor functions, and defaults to **Cursor**.

### Backups

By default the editor does not make backups of programs when you save them, but this can be turned on by clicking on the **Backups On** radio button.

## Auto Indenting

It can be useful to indent lines from the left automatically, so the editor supports an auto-indent mode. When active, an indent is added to the start of each new line created when you press **[Return]**. The contents of the indent of the new line is taken from the white space (i.e. tabs and/or spaces) at the start of the previous line.

## Saving Preferences

If you click on the `Cancel` button any changes you make will be ignored. If you click on the `OK` button the changes specified will remain in force until you quit the editor. If you would like the configuration made permanent then click on the `Save` button, which will create the file `1STBASIC.INF` on your disk. Next time you run **FirST BASIC** the configuration will be read from that file.

## Running Programs

If you click on `Run` from the `Program` menu or press **[Alternate][X]** you will then run the program in the main window. When your program finishes it will normally wait for a key then return you to the editor. If there are any errors it will not be possible to run the program.

## The Immediate Mode

**FirST BASIC** has a special immediate mode which allows you to try out small sections of program without effecting your main program. To do this, press **[Alternate][0]** (the zero digit, not the letter O) or click on `Immediate window` on the `Program` menu. This will switch the editor into Immediate mode, which works in a similar way to the ordinary mode (known as the *Program Window*). You can type in a few program lines then run them in the normal way, by pressing **[Alternate][X]**. You can also clear out the contents of the Immediate window by pressing **[Alternate C]**, or clicking on `Clear` from the `File` menu. When in immediate mode, most menu items are disabled, as well as the block commands. To return to the Program Window, press **[Alternate][0]**, or click on `Immediate window` on the `Program` menu, or click on the window's Close box.

Don't type too much into the Immediate window as you can't save it anywhere - you can only save the contents of the Program Window.

## The GEM Editor Window

The window used by the editor works like all other GEM windows, so you can move it around by using the Move bar on the top of it, you can change its size by dragging on the size box, and make it full size (and back again) by clicking on the Full box. Clicking on the Close box is equivalent to choosing `Quit` from the `File` menu.

# QUICK REFERENCE SECTION

## Statements and Functions

<b>ABS</b> ( <i>numeric_expression</i> ) returns the absolute value of <i>numeric_expression</i>	Function
<b>ASC</b> ( <i>string_expression</i> ) returns the ASCII code of the first character of <i>string_expression</i>	Function
<b>ATN</b> ( <i>numeric_expression</i> ) returns the arctangent of <i>numeric_expression</i>	Function
<b>BAR</b> <i>x_start,y_start,width,height</i> draws an outlined box in the current output window	Statement
<b>BEEP</b> outputs a short beep to the monitor speaker	Statement
<b>BLOAD</b> <i>filename,address</i> load a binary file into address	Statement
<b>BSAVE</b> <i>filename,address,length</i> saves length bytes starting from address, to filename	Statement
<b>[CALL]</b> <i>sub_program_name</i> [( <i>parameter</i> [, <i>parameter</i> ]...)] calls a sub-program or a library routine	Statement
<b>CALL LOC</b> <i>address</i> [, <i>parameter</i> ]... calls a machine code routine with optional parameters	Statement
<b>CALLS</b> <i>sub_program_variable</i> calls a sub-program indirectly using a variable pointer	Statement
<b>CDBL</b> ( <i>numeric_expression</i> ) converts <i>numeric_expression</i> to a double precision number	Function
<b>CHDIR</b> <i>pathname</i> sets the current directory	Statement
<b>CHR\$</b> ( <i>ASCII_code</i> ) returns a one-character string corresponding to <i>ASCII_code</i>	Function
<b>CINT</b> ( <i>numeric_expression</i> ) converts <i>numeric_expression</i> to an integer by rounding	Function
<b>CIRCLE</b> <i>x_centre,y_centre,radius</i> [, <i>start_angle, end_angle</i> ] draws a circle or an arc in the current window	Statement

<b>CLEAR</b>	clears all variables and closes all channels	Statement
<b>CLOSE</b> <i>[[#]channel_number[,[#]channel_number]...]</i>	terminates I/O to the specified file or device	Statement
<b>CLS</b>	Statement clears the screen and returns the cursor to the upper left corner	
<b>COLOR</b> <i>text_colour[,fill_colour[,line_colour[,index][,style]</i>	sets the current colour and line drawing attributes	Statement
<b>CONST</b> <i>name=integer_constant[,name=integer_constant]...</i>	defines symbolic integer constants	Statement
<b>COS</b> <i>(numeric_expression)</i>	returns the cosine of <i>numeric_expression</i> which must be in radians	Function
<b>CSNG</b> <i>(numeric_expression)</i>	converts <i>numeric_expression</i> to single precision	Function
<b>CSRLIN</b>	returns the current line position of the cursor	Function
<b>CVD</b> <i>(8-byte string of a double precision float)</i>		Function
<b>CVI</b> <i>(2-byte string of an integer)</i>		Function
<b>CVL</b> <i>(4-byte string of a long integer)</i>		Function
<b>CVS</b> <i>(4-byte string of a single precision float)</i>	these functions return the internal numeric values of their string arguments	Function
<b>DATA</b> <i>constant[,constant]...</i>	defines data to be used in conjunction with READ	Statement READ
<b>DATETIME</b>	returns (function) or sets (statement) the current date	Statement/Function
<b>DECR</b> <i>numeric_variable</i>	subtracts 1 from <i>numeric_variable</i>	Statement
<b>DEF FN</b> <i>function_name[(parameter_list)]=expression</i>		Statement
<b>DEF FN</b> <i>function_name[(parameter_list)</i> <i>[LOCAL variable_list]</i> <i>[STATIC variable_list]</i> <i>[SHARED variable_list]</i> <i>statements</i> <i>[EXIT DEF]</i>		Statement
<b>END DEF</b>	define single - or multi-line user functions with parameters	



<b>DEF FN</b> <i>function_name</i> =expression]	define single-line user functions without parameters	
<b>DEF SEG</b> = <i>numeric_expression</i>	defines the mode of operation of PEEK and POKE	Statement
<b>DEFDBL</b> <i>letter_range</i> [, <i>letter_range</i> ]...		Statement
<b>DEFINT</b> <i>letter_range</i> [, <i>letter_range</i> ]...		Statement
<b>DEFLNG</b> <i>letter_range</i> [, <i>letter_range</i> ]...		Statement
<b>DEFSNG</b> <i>letter_range</i> [, <i>letter_range</i> ]...		Statement
<b>DEFSTR</b> <i>letter_range</i> [, <i>letter_range</i> ]...	declare variables to be of double-, integer-, long integer-, single- or string-precision respectively	Statement
<b>DIM</b> [SHARED] <i>variable</i> {( <i>subscripts</i> ...)}[, <i>variable</i> {( <i>subscripts</i> ...)}]...	defines the maximum values for array subscripts and allocates the storage	Statement
<b>DO</b> [{WHILE   UNTIL} <i>boolean_expression</i> ] <i>statements</i>		Statement
[EXIT {LOOP   DO}] <i>statements</i>		
{WEND   LOOP} [{WHILE   UNTIL} <i>boolean_expression</i> ]	repeat statements within the DO/LOOP while the condition(s) are true (WHILE) or false (UNTIL)	
<b>ELLIPSE</b> <i>x_centre</i> , <i>y_centre</i> , <i>x_radius</i> , <i>y_radius</i> [, <i>start_angle</i> , <i>end_angle</i> ]	draws an ellipse or elliptical arc in the current window	Statement
<b>END</b> [{DEF   IF   REPEAT name SELECT   SUB}...]	ends a BASIC program, function definition, IF...THEN...ELSE block, SELECT statement, REPEAT loop or sub-program respectively	Statements
<b>EOF</b> ( <i>channel_number</i> )	returns the end-of-file condition of <i>channel_number</i>	Function
<b>ERASE</b> <i>array_name</i> [, <i>array_name</i> ]...	deallocates arrays previously defined with DIM	Statement
<b>ERL</b>	returns the line number at which the last error occurred	Function
<b>ERR</b>	returns the error number of the last error	Function
<b>ERROR</b> <i>integer_expression</i>	simulates the occurrence of a BASIC runtime error	Statement

<b>EXIT</b> {DEF   DO   IF   FOR   LOOP   SELECT   SUB   identifier}	Statements
exits a function definition, BASIC sub-program or a structured statement respectively	
<b>EXP</b> ( <i>numeric_expression</i> )	Function
returns the exponential value of <i>numeric_expression</i>	
<b>FEXISTS</b> ( <i>filename</i> )	Function
determines whether or not the specified file exists	
<b>FIELD</b> [# <i>channel_number</i> , <i>field_width</i> AS <i>string_variable</i> ...	Statement
allocates space for variables within random access filing	
<b>FILES</b> [ <i>file_spec</i> ]	Statement
lists the names of the files within the directory specified	
<b>FILL</b> <i>x_start</i> , <i>y_start</i>	Statement
fills enclosed shapes already drawn around <i>x_start</i> , <i>y_start</i>	
<b>FIX</b> ( <i>numeric_expression</i> )	Function
returns the truncated integer part of <i>numeric_expression</i>	
<b>FOR</b> <i>counter</i> = <i>start</i> TO end [STEP <i>increment</i> ]	Statement
statements	
<b>NEXT</b> [ <i>counter</i> ][, <i>counter</i> ]...	
executes a series of instructions in loop for a fixed number of times	
<b>FRE</b> {( <i>numeric_expression</i> ) ( <i>string_expression</i> )}	Functions
returns the size of the free heap space or GEMDOS free memory	
<b>GET</b> [# <i>channel_number</i> [, <i>record_number</i> ]File I/O	Statement
reads a record from an already-open random access disk file into a buffer	
<b>GET</b> ( <i>x1</i> , <i>y1</i> )-( <i>x2</i> , <i>y2</i> ), <i>array_name</i> Graphics	Statement
grabs binary images from any part of the screen	
<b>GOSUB</b> { <i>line_number1</i>   <i>line_label1</i> }	Statements
statements	
<b>RETURN</b> { <i>line_number2</i>   <i>line_label2</i> }	
branch to and return from a subroutine	
<b>GOTO</b> { <i>line_number</i>   <i>line_label</i> }	Statements
transfer program execution to the line/label specified	
<b>HEX\$</b> ( <i>numeric_expression</i> )	Function
returns a string that represents the hexadecimal value of <i>numeric_expression</i>	

<b>IF</b> <i>boolean_expression</i> THEN <i>statement_1</i> [ <i>statement2</i> ]Statement [ELSEIF <i>boolean_expression</i> THEN <i>statement_3</i> [ <i>statement_4</i> ]] [ELSE <i>statement_5</i> [ <i>statement_6</i> ]] END IF allows conditional execution or branching based on the value of a boolean expression	
<b>INCR</b> <i>numeric_variable</i> adds 1 to <i>numeric_variable</i>	Statement
<b>INKEY\$</b> reads a character from the keyboard, if one is available, without echo	Function
<b>INP</b> ( <i>BIOS_handle</i> ) returns a byte from the device given by <i>BIOS_handle</i>	Function
<b>INPUT</b> [;][ <i>"prompt"</i> {;},] <i>variable_list</i> prompts for input from the keyboard and then reads input into <i>variable_list</i>	Statement
<b>INPUT</b> # <i>channel_number</i> , <i>variable_list</i> reads data from a file or device into <i>variable_list</i>	Statement
<b>INPUT\$</b> ( <i>n</i> [,{#} <i>channel_number</i> ]) reads <i>n</i> characters from the specified channel	Statement
<b>INSTR</b> ([ <i>start</i> ,]1 <i>st_string</i> ,2 <i>nd_string</i> ) returns the location of the first occurrence, after <i>start</i> , of 2 <i>nd_string</i> within 1 <i>st_string</i>	Function
<b>INT</b> ( <i>numeric_expression</i> ) returns the largest integer less than or equal to <i>numeric_expression</i>	Function
<b>KILL</b> <i>file_spec</i> delete all files fitting <i>file_spec</i> from the current disk	Statement
<b>LBOUND</b> ( <i>array</i> [, <i>dimension</i> ]) returns the smallest available subscript of the given array	Function
<b>LCASE\$</b> ( <i>string_expression</i> ) converts letters within <i>string_expression</i> to lower case	Function
<b>LEFT\$</b> ( <i>string_expression</i> , <i>n</i> ) returns a string made up of the leftmost <i>n</i> characters of <i>string_expression</i>	Function
<b>LEN</b> ( <i>string_expression</i> ) returns the number of characters in <i>string_expression</i>	Function
<b>[LET]</b> <i>variable</i> = <i>expression</i> assigns <i>expression</i> to <i>variable</i>	Statement
<b>LIBRARY</b> <i>library_name</i> {, <i>library_name</i> }... defines which libraries are to be used by the program	Statement

<b>LINEF</b> <i>x_start,y_start,x_end,y_end</i>	Statement
draws a line between two points in the current window	
<b>LINE INPUT</b> [;] [ <i>"prompt"</i> ] <i>string_variable</i>	Statement
assigns an entire line of input to <i>string_variable</i> , ignoring delimiters such as commas	
<b>LINE INPUT</b> <i>#channel_number,string_variable</i>	Statement
reads a sequence of characters terminated by a CRLF into <i>string_variable</i> from the device/file given by <i>channel_number</i>	
<b>LOC</b> ( <i>channel_number</i> )	Function
returns the current position within an open file	
<b>LOCAL</b> <i>variable_list</i>	Statement
declares the list of variables to be local to a function or sub_program	
<b>LOCATE</b> <i>row[,column[,cursor]]</i>	Statement
puts the cursor at the position specified and enables/disables it	
<b>LOF</b> ( <i>channel_number</i> )	Function
returns the length of the given file	
<b>{LOG   LOG10   LOG2}</b> ( <i>numeric_expression</i> )	Functions
returns the logarithms, to various bases, of <i>numeric_expression</i>	
<b>LPOS</b> ( <i>argument</i> )	Function
returns the current position of the printer head	
<b>LPRINT</b> [ <i>expression_list</i> [[;];]]	Statement
LPRINT USING <i>format_string;expression_list</i> [[;];]]	Statement
print data through the current printer port - see PRINT, PRINT USING	
<b>LSET</b> <i>string_variable=string_expression</i>	Statement
left-justify a string variable, normally used for FIELDed variables	
<b>MID\$</b> ( <i>string_expression,n[,length]</i> )	Function
returns length characters from <i>string_expression</i> , starting at the <i>n</i> th position within the string	
<b>MID\$</b> ( <i>string_variable,n[,length]</i> )= <i>string_expression</i>	Statement
modifies part of a string variable	
<b>MKDIR</b> <i>pathname</i>	Statement
creates the sub-directory given by <i>pathname</i>	
<b>MKI\$</b> ( <i>integer_expression</i> )	Function
<b>MKL\$</b> ( <i>long_integer_expression</i> )	Function
<b>MKS\$</b> ( <i>single_precision_expression</i> )	Function

<b>MKD\$</b> ( <i>double_precision_expression</i> )	Function
convert the numeric data of the expressions into strings	
<b>MOUSE</b> ( <i>attribute</i> )	Function
reads the current position of the mouse, status of the buttons and the keyboard shift keys	
<b>MOUSE</b> <i>mouse_type</i>	Statement
sets the mouse pointer to a pre-defined form or remove it from the screen	
<b>NAME</b> <i>old_filename AS new_filename</i>	Statement
renames a file	
<b>OCT\$</b> ( <i>numeric_expression</i> )	Function
returns a string which is the octal representation of <i>numeric_expression</i>	
<b>ON ERROR GOTO</b> { <i>line_number</i>   <i>line_label</i> }	Statement
enable error handling and specify the error handling routine	
<b>ON n GOSUB</b> { <i>line_number</i>   <i>line_label</i> },{ <i>line_number</i>   <i>line_label</i> }... calls one of a list of subroutines depending on the value of n	Statement
<b>ON n GOTO</b> { <i>line_number</i>   <i>line_label</i> },{ <i>line_number</i>   <i>line_label</i> }... causes program execution to branch to one of a list of program lines/labels depending on the value of n	Statement
<b>OPEN</b> <i>file_spec</i> [FOR mode] AS [#] <i>channel_number</i> [LEN= <i>record_size</i> ]	Statement
open a file for reading or writing	
<b>OPTION BASE</b> { 0   1 }	Statement
defines the lowest subscript value for arrays	
<b>OUT</b> <i>bios_handle, integer_expression</i>	Statement
sends a byte to the BIOS output routines	
<b>PALETTE</b> <i>colour_number, physical_colour</i>	Statement
allows you to change the palette of screen colours	
<b>PALETTE USING</b> <i>array_element</i>	Statement
modify the palette using the elements of the array	
<b>PCIRCLE</b> <i>x_centre, y_centre, radius</i> [, <i>start_angle, end_angle</i> ]	Statement
draws a solid circle or arc in the current window	
<b>PCOPY</b>	Statement
dump the current screen to the printer	
<b>PEEK</b> ( <i>address</i> )	Function
returns the byte, word or long value (depending on the DEF SEG statement) of the contents at location <i>address</i>	

<b>PEEKB</b> ( <i>address</i> )	Function
<b>PEEKL</b> ( <i>even_address</i> )	Function
<b>PEEKW</b> ( <i>even_address</i> )	Function
return the contents (of various types) of the memory specified	
<b>PELLIPSE</b> <i>x_centre,y_centre,x_radius,y_radius</i> [ <i>,start_angle,end_angle</i> ]	Statement
draws a solid ellipse or elliptical arc in the current window	
<b>POINT</b> ( <i>x_pixel,y_pixel</i> )	Function
returns the colour of a particular pixel	
<b>POKE</b> <i>address,data</i>	Statement
writes data directly to the address in memory	
<b>POKEB</b> <i>address,byte_value</i> Statement	
<b>POKEL</b> <i>even_address,long_value</i> Statement	
<b>POKEW</b> <i>even_address,word_value</i> Statement	
write data of different types directly into memory	
<b>POS</b> ( <i>x</i> )	Function
returns the column number of the current cursor position	
<b>PRESET</b> [STEP] ( <i>x_pos,y_pos</i> ) [ <i>,colour</i> ]	Statement
resets or sets a pixel to a given colour within the current window	
<b>PRINT</b> [ <i>expression_1</i> ] [{ ;     , } [ <i>expression_2</i> ] [:]...	Statement
prints data on the screen	
<b>PRINT</b> <i>#channel_num</i> , [USING <i>format_string</i> ] <i>expression_list</i> [:]	Statement
writes formatted data to a file or device	
<b>PRINT USING</b> <i>format_string</i> ; <i>expression_list</i> [{ ;   ; }]	Statement
print formatted data to the screen	
<b>PSET</b> [STEP] ( <i>x_pos,y_pos</i> ) [ <i>,colour</i> ]	Statement
plots a pixel of a given colour in the current window	
<b>PUT</b> [#] <i>channel_number</i> [ <i>,record_number</i> ] <i>File I/O</i>	Statement
writes a record from the file buffer to the given random access file	
<b>PUT</b> ( <i>x,y</i> ), <i>array_name</i> [ <i>,verb</i> ] <i>Graphics</i>	Statement
copies (blits) the rectangular image saved with GET on to the screen	
<b>RANDOMIZE</b> [ <i>expression</i> ]	Statement
provides the random generator with a new seed	
<b>READ</b> <i>variable_list</i>	Statement
assigns variables with values given by DATA statements	
<b>REDIM</b> [APPEND] <i>array(subscripts)</i> [ <i>,array(subscripts)</i> ]...	Statement
changes the size of an already-defined array	

<b>REM</b> <i>remarks...Statement</i>	
allows comments to be added to your program (you can also use ‘)	
<b>REPEAT</b> <i>name</i>	Statement
statements	
[EXIT <i>name</i> ]	
statements	
<b>END REPEAT</b> <i>name</i>	
statements within the REPEAT loop are executed until an EXIT or END <b>REPEAT</b> statement for the loop is encountered	
<b>RESET</b>	Statement
closes all open disk files	
<b>RESTORE</b> [{ <i>line_number</i>   <i>label</i> }]	Statement
enables a READ statement to access a DATA statement that has been already read	
<b>RESUME</b> { <i>line_number</i>   <i>label</i> }	Statement
resumes program execution from within an error handling routine at the given line/label	
<b>RETURN</b> [{ <i>line_number</i>   <i>label</i> }]	Statement
returns program execution from a subroutine	
<b>RIGHT\$</b> ( <i>string_expression</i> , <i>n</i> )	Function
returns a string starting at the <i>n</i> th character from the right of <i>string_expression</i>	
<b>RMDIR</b> <i>pathname</i>	Statement
removes an empty sub-directory	
<b>RND</b> [( <i>n</i> )]	Function
returns a pseudo-random single-precision number between 0 and 1	
<b>RSET</b> <i>string_variable</i> = <i>string_expression</i>	Statement
moves data into a random access file buffer. It can also be used to right-justify the string in <i>string_variable</i>	
<b>RUN</b> [{ <i>line_number</i> }]	Statement
restarts the current program	
<b>SADD</b> ( <i>string_expression</i> )	Function
returns the memory address of <i>string_expression</i>	
<b>SCREEN</b> <i>mode</i>	Statement
changes the screen mode	

<b>SELECT</b> [CASE   ON] <i>variable</i>	Statement
[CASE   =] <i>case_list</i>	
statements	
[CASE   =] <i>case_list</i>	
statements	
[CASE   =] <i>case_list</i>	
statements	
END SELECT	
one series of statements is executed depending on the value of variable	
<b>SGN</b> ( <i>numeric_expression</i> )	Function
returns the sign of <i>numeric_expression</i>	
<b>SHARED</b> <i>variable_list</i>	Statement
enables a sub-program to access variables from the main program without having them passed as parameters	
<b>SIN</b> ( <i>numeric_expression</i> )	Function
returns the sine of <i>numeric_expression</i> which must be in radians	
<b>SOUND</b> <i>voice,volume</i> [, <i>note</i> ][, <i>octave</i> ][, <i>duration</i> ]	Statement
allows control over the 3 sound channels	
<b>SPACE\$</b> ( <i>n</i> )	Function
returns a string of <i>n</i> spaces	
<b>SPC</b> ( <i>n</i> )	Function
causes <i>n</i> columns to be skipped in a PRINT statement	
<b>SQR</b> ( <i>numeric_expression</i> )	Function
returns the square root of <i>numeric_expression</i>	
<b>STATIC</b> <i>variable_list</i>	Statement
declares variables as local to a function or sub-program but holds their values constant over multiple calls and does not create a new variable for each call	
<b>STICK</b> ( <i>n</i> )	Function
returns the positions of the two joysticks	
<b>STOP</b> [ <i>process_returncode</i> ]	Statement
causes the program to end; all files are closed and control is returned to the operating system	
<b>STR\$</b> ( <i>numeric_expression</i> )	Function
returns the string representation of <i>numeric_expression</i>	
<b>STRIG</b> ( <i>n</i> )	Function
returns the status of the specified joystick fire button	
<b>STRIG</b> {ON   OFF}	Statement
controls the use of the joystick commands	
<b>STRING\$</b> ( <i>m,n</i> )	Function
returns a string of the ASCII character <i>n</i> of length <i>m</i>	



<b>STRING\$</b> ( <i>m,string_expression</i> )	Statement
returns a string of the ASCII character that is the first character of <i>string_expression</i> , of length <i>m</i>	
<b>SUB</b> <i>global_name</i> ( <i>parameter_list</i> ) [STATIC]	Statement
statements [EXIT SUB] statements END SUB defines a sub-program	
<b>SWAP</b> <i>1st_variable,2nd_variable</i>	Statement
causes the values of the two variables to be exchanged	
<b>SYSTAB</b>	Function
returns the address of an internal system table	
<b>SYSTEM</b>	Statement
causes program execution to end, closes all files and returns to the operating system	
<b>TAB</b> ( <i>n</i> )	Function
moves the print position to the <i>n</i> th column	
<i>(numeric_expression)</i>	Function
returns the tangent of <i>numeric_expression</i> which must be in radians	
<b>TIME\$</b> <i>Statement/</i>	Function
returns or sets the system time	
<b>TIMER</b>	Function
returns an internal timer count in seconds as a single-precision floating point number	
<b>UBOUND</b> ( <i>array_name</i> [, <i>dimension</i> ])'	Function
returns the largest available subscript of the given array	
<b>UCASE\$</b> ( <i>string_expression</i> )	Function
upper-cases every alphabetic character in <i>string_expression</i>	
<b>VAL</b> ( <i>string_expression</i> )	Function
returns the numeric value of <i>string_expression</i>	
<b>VARPTR</b> ( <i>variable_name</i> )	Function
returns the memory address of the variable <i>variable_name</i>	
<b>VARPTR</b> ( <i>#channel</i> )	Function
returns the memory address of the input/output buffer associated with the given channel	
<b>VARPTRS</b> ( <i>subname</i> )	Function
returns the memory address of the sub-program given	

<b>WAVE</b> <i>enable[,envelope[,shape[,period[,delay]]]]</i> controls the waveforms used by the SOUND statement	Statement
<b>WHILE</b> <i>condition</i> statements	Statement
<b>WEND</b> a series of statements is executed in a loop until condition is false	
<b>WIDTH</b> [ <i>#channel_number,</i> ] <i>width [,tab_setting]</i>	Statement
<b>WIDTH LPRINT</b> <i>width [,tab_setting]</i> assigns a line width to the given file, screen or printer	Statement
<b>WINDOW CLOSE</b> [ <i>id</i> ]	Statement
<b>WINDOW CONTRL</b> <i>id,control,value</i>	Statement
<b>WINDOW FULLW</b> [ <i>id</i> ]	Statement
<b>WINDOW GET</b> <i>id,rectangle,xpos,ypos,width,height</i>	Statement
<b>WINDOW LOCATE</b> <i>id,xpos,ypos,width,height</i>	Statement
<b>WINDOW NAME</b> <i>id [,title_string] [,status_string]</i>	Statement
<b>WINDOW OFF</b>	Statement
<b>WINDOW ON</b>	Statement
<b>WINDOW OPEN</b> <i>id,title_string,xpos,ypos,width,height,type</i>	Statement
<b>WINDOW OUTPUT</b> [ <i>id</i> ]	Statement
<b>WINDOW READ</b> <i>id,control,result</i> these statements control multi-window screen output	Statement
<b>WRITE</b> [ <i>expression_list</i> ] prints the data specified in <i>expression_list</i> to the screen	Statement
<b>WRITE</b> <i>#channel_number, expression_list</i> prints the data specified in <i>expression_list</i> to a sequential file The Supplied Library	Statement
<b>GEMVDI: Control</b> <b>fnGDOS</b>   <b>fnHANDLE</b>   <b>CHANGE_HANDLE</b> <i>handle</i>   <b>fnRESOLUTION</b>   <b>v_opnwk</b> <i>inv()</i> , <i>varptr handle</i> , <i>outv()</i>   <b>v_clswk</b>   <b>v_opnwk</b> <i>inv()</i> , <i>varptr handle</i> , <i>outv()</i>   <b>v_clsawk</b>   <b>v_updwk</b>   <b>v_clrwrk</b>   <b>FNvst_load_fonts</b>   <b>vst_unload_fonts</b>   <b>vs_clip</b> <i>flag, x1, y1, x2, y2</i> <b>GEMVDI: Drawing Primitives</b> <b>v_pline</b> <i>n, pts()</i>   <b>v_pmarker</b> <i>n, pts()</i>   <b>v_gtext</b> <i>x, y, text\$</i>   <b>v_fillarea</b> <i>n, pts()</i>	

v\_contourfill x,y,colour | vr\_rectfl x1, y1, x2, y2 | v\_bar x1, y1, x2, y2 |  
v\_arc x, y, r, start\_angle, end\_angle | v\_pieslice x, y, r, start\_angle, end\_angle  
v\_circle x, y, r | v\_ellarc x, y, xr, yr, start\_angle, end\_angle |  
v\_ellipse x, y, xr, yr, start\_angle, end\_angle | v\_ellipse x, y, xr, yr |  
v\_rbox x1, y1, x2, y2 | v\_rfbox x1, y1, x2, y2 |  
v\_justified x, y, text\$, length, wordflag, charflag  
GEMVDI: Attributes  
vswr\_mode mode | vs\_color index, r, g, b | vsl\_type type | vsl\_udsty pattern  
vsl\_width width | vsl\_color colour | | vsl\_ends beginstyle,endstyle |  
vsm\_type type | vsm\_height height | vsm\_color colour | vst\_height height |  
vst\_point points | vst\_rotation angle | vst\_font font | vst\_color colour |  
vst\_effects effects | vst\_alignment horizontal,vertical | vsf\_interior fill\_interior  
vsf\_style style\_index | vsf\_color colour | vsf\_perimeter flag |  
vsf\_updat plane(), planes  
GEMVDI: Raster Functions  
vro\_cpyfm mode, xy(), source&, dest& |  
vrt\_cpyfm mode, xy(), source&, dest&, foreground, background |  
v\_get\_pixel x, y, varptr value, varptr index  
GEMVDI: Mouse & Keyboard State Functions  
v\_show\_c flag | v\_hide\_c |  
vq\_mouse varptr button, varptr x, varptr y | vq\_key\_s varptr kstate  
GEMVDI: Enquiries  
vq\_extnd flag, info() | vq\_color colour\_index, flag, rgb() | vql\_attributes info()  
vqm\_attributes info() | vql\_attributes info() | vqt\_attributes info() |  
vqt\_extent text\$, info() | vqt\_width char, varptr cellwidth, varptr left, varptr right  
FNvqt\_name(number, varptr name\$) |  
vqt\_fontinfo varptr firstchar, varptr lastchar, distances(), varptr maxwidth, effects()  
GEMAES: Message Passing  
appl\_read ap\_id,length,message& |  
appl\_write ap\_id,length,message& | FNappl\_find name\$  
GEMAES: Events  
FNevnt\_keybd |  
FNevnt\_button(clicks, mask ,state, varptr xout, varptr yout, varptr button, varptr  
kstate) |  
evnt\_mouse flag, x, y, w, h, varptr xout, varptr yout, varptr button, varptr kstate |  
evnt\_mesag message& | evnt\_timer time&  
FNevnt\_multi( eflags, clicks, mask, state, flag1, x1, y1, w1, h1, flag2, x2, y2, w2,  
h2, message&, time&, varptr xout, varptr yout, varptr button, varptr kstate, varptr  
key\_pressed, varptr gotclicks) | FNevnt\_dclick newvalue, setorget  
GEMAES: Menus  
FNmenu&(menu\_string\$) | menu\_bar tree&,flag |  
menu\_ichck tree&,item,flag | menu\_ienable tree&,item,flag  
menu\_tnormal tree&,title,flag | menu\_text tree&,item,text\$  
GEMAES: Objects  
FNobjc\_add(tree&, parent, child) | FNobjc\_delete(tree&,object) |  
FNobjc\_draw(tree&, object, depth, x, y, w, h) |  
FNobjc\_find(tree&,object,depth,x,y)  
FNobjc\_offset(tree&,object,varptr x,varptr y) |  
FNobjc\_order(tree&,object,newpos) |  
FNobjc\_edit(tree&,object,char,index,kind,varptr newindex) |  
FNobjc\_change(tree&,object,x,y,w,h,newstate,flag)

GEMAES: Forms

FNform\_do(tree&,object) |

form\_dial flag, x1, y1, w1, h1, x2, y2, w2, h2 | FNform\_alert(default,alert\$) |

FNform\_error(errorcode) | form\_center tree&,varptr x,varptr y,varptr w,varptr h

GEMAES: Graphics

graf\_rubberbox x,y,minw,minh,varptr outw,varptr outh | graf\_dragbox w, h, startx, starty, boxx, boxy, boxw, boxh, varptr outx, varptr outy)

graf\_movebox w, h, x1, y1, x2, y2 |

graf\_growbox x1, y1, w1, h1, x2, y2, w2, h2 |

graf\_shrinkbox x1, y1, w1, h1, x2, y2, w2, h2

FNgraf\_watchbox(tree&, object, instate, outstate)

FNgraf\_slidebox(tree&, parent, object, vertical)

FNgraf\_handle(varptr wcell, varptr hcell, varptr wbox, varptr hbox) |

graf\_mouseflag, user\_form& |

graf\_mkstate varptr x, varptr y, varptr state, varptr kstate

GEMAES: Scrap Directory

FNscrp\_read(varptr directory\_name\$) | FNscrp\_write(directory\_name\$)

GEMAES: File Selector

fsel\_input varptr path\$, varptr name\$, varptr ok

GEMAES: Windows

FNwind\_create kind, x ,y, w, h | FNwind\_open(handle, x, y, w, h) |

FNwind\_close(handle) | FNwind\_delete(handle) |

FNwind\_get(handle, sort, varptr x, varptr y, varptr w, varptr h) |

FNwind\_set(handle, sort, x, y, w, h) | FNwind\_find(x, y)

FNwind\_update(sort) | FNwind\_calc(type, kind, x, y, w, h, varptr xout, varptr yout, varptr wout, varptr hout)

GEMAES: Resource Files

FNrsrc\_load(file\_name\$) | FNrsrc\_free |

FNrsrc\_gaddr(type, index, varptr addr&) | FNrsrc\_saddr(type,index,addr&) |

rsrc\_obfix tree&, object

GEMAES: Shell Routines

FNshel\_read( varptr cmd\$, varptr tail\$) | FNshel\_find(varptr file\$)

shel\_envrn varptr env\$, name\$

## Variable Types

Type	Identifier	Range
Integer (16 bits)	%	-32768 to 32767
Long integer (32 bits)	&	-2147483648 to 2147483647
FFP Single precision	!	-9.2E18 to +9.2E18
IEEE Double precision	#	-1.8D308 to 1.8D308
String	\$	0 to 16,777,215 characters

## Character Set

**FIRST BASIC** uses plain ASCII characters in its input files. The following characters have special meanings:

a-z, A-Z	The letters, which are used in reserved words and the user's variable names, labels and sub-program names. Lower and upper case are treated as the same in variable and reserved word definitions.
E, e, D,	dare also used for exponents in numbers.
0-9	The digits, which are used in numbers and can also be used in names as long as they are not the first character.
.	The full stop or period, which is used as the decimal point in numbers and can also be used in names as long as it is not the first character.
%	The percentage sign, which is used to indicate that a variable is a 16 bit integer i.e. whose values must be in the range -32768 to 32767.
&	The ampersand, which is used to indicate that variables are long integers i.e. whose values must be in the range $-2^{31}$ to $2^{31}-1$ . Also used to introduce hexadecimal, octal and binary constants.
!	The exclamation mark, which is used to indicate that a variable is a single-precision floating point number.
#	The hash or number sign, which is used to indicate that a variable is a double precision floating point number and also used to indicate that certain input/output operations are to be directed to channels rather than the screen (e.g. <code>PRINT #</code> ).
\$	used to indicate string variables.
_	The underline character, which can be used in variables after the first character. If it appears at the start of a symbol it indicates the rest of the line is to be ignored and that the following line is to be considered part of the current one.
"	The quotation mark or double quote which is used to delimit string literals.
'	The apostrophe or single quote which is used to indicate that the rest of this line is to be regarded as a comment.

- ( ) The parentheses or round brackets, which are used to enclose function arguments, over-ride the priority of operators and indicate arrays.
- + - \* / the basic arithmetic operators.
- = assignment operator and equality operator.
- < > less than and greater than comparison operators.
- ^ exponentiation operator.
- \ The back-slash character, which is used as the integer division operator.
- [CTRL][Z]** if ASCII **[CTRL][Z]** (chr\$(26)) is found in a file it is treated as end-of-file. Other characters with ASCII values lower than 32 are treated as white space, and ignored. Other characters may be used in strings, but otherwise will generate a warning and will be ignored.

## Operators

Expressions are made up of constants, variables, array variables, function calls and operators. The order of priority is listed below with the highest priority first:

1. Exponentiation ( to the power of ) (^)
2. Unary Minus (-)
3. Multiplication (\*) and Floating Point Division (/)
4. Integer Division (\)
5. Modulus ( MOD )
6. Addition (+) and Subtraction (-)
7. Comparisons (=,<>,>,<,>=,<=, ==)
8. NOT
9. AND
10. OR and XOR (exclusive or)
11. EQV
12. IMP

The only exception to this is that  $x^y$  is evaluated as  $x^{(y)}$ . To change the order of evaluation use parentheses (round brackets).

## Your FirST BASIC

This complete reference to **FirST BASIC** contains invaluable information on how to get the best out of the package including a full explanation of all the commands and libraries together with an extensive tutorial that not only teaches you, in a friendly, interactive way, how to use **FirST BASIC** but also lets you into the secrets of programming the Atari ST and GEM to produce some fascinating programs. **Your FirST BASIC** costs only 14.95, contains over 300 pages of information and is available from good computer shops or directly from HiSoft. If you have difficulty obtaining your copy, you can order one using the order form on the next page.

## Power BASIC and HiSoft BASIC

HiSoft has many other language and utility products for your Atari ST including two further BASIC compilers, **Power BASIC** and **HiSoft BASIC**. Both packages allow compilation to disk so that you can produce stand-alone programs for your own use or to give-away or sell and with no royalties to pay. They also both include a full set of compiler options allowing comprehensive control over the compilation process that will dramatically speed up your programs (all checks are ON in **FirST BASIC**), complete libraries including BIOS, XBIOS, sound and graphics libraries and extensive manuals covering every aspect of the packages.

**HiSoft BASIC** is intended for the professional programmer and includes many advanced and useful features such as program profiling to fine-tune your programs for speed, user-expandable libraries so that you can write your own extensions to BASIC, the ability to include your program's symbols for low-level debugging and a stand-alone version of the compiler so that you can use your own editing environment if you wish. **HiSoft BASIC** also enables you to create desk accessories from within BASIC.

Both Power BASIC and HiSoft BASIC are available at special upgrade prices to existing users of **FirST BASIC**.

To order simply photocopy and fill out the order form on the next page and send **Your FirST BASIC** disk together with your remittance to HiSoft. Order Form  
Please use the following order form to order your copies of the **Your FirST BASIC** book and your upgrade to either **Power BASIC** or **HiSoft BASIC**.

<p>Please send me ___ copies of Your FirST BASIC @ £14.95 ea ___ Please send me my upgrade to Power BASIC @ £35 ___ Please send me my upgrade to HiSoft BASIC @ £60 ___</p> <p>Name: .....</p> <p>Address: .....</p> <p>.....</p> <p>.....</p> <p>Postcode: .....</p> <p>I enclose a cheque/money order for £ _____ I would like to pay by Access _____ Visa _____</p> <p>My credit card number is: .....</p> <p>Expiry date: .....</p> <p>Signed: _____ Date: _____</p>
--

Send the above form together with your remittance to: HiSoft, The Old School, Greenfield, Bedford, UK MK45 5DE. Please allow 14 days for delivery.

Customers outside the UK should contact their local Atari dealer for information. In case of difficulty, overseas customers may use the above order form, adding £2 postage & packing for **Your FirST BASIC** and £4 postage & packing for **Power BASIC** or **HiSoft BASIC** upgrades.





**ATARI<sup>®</sup>**  
**NEOchrome<sup>™</sup>**

Painting Program  
for all Atari ST<sup>™</sup> and MEGA<sup>™</sup> Computers

ATARI

NEOchrome™

Painting Program

for ATARI ST™ and NEO™ Computers

# INTRODUCTION

## Welcome to NEOchrome!

NEOchrome is the powerful and sophisticated painting program for the ATARI ST Computer. With it, you can create countless shapes, pictures, and graphic designs, each using up to 16 colours from the ST Computer's colour palette.

NEOchrome is a step up from drawing programs. Its bountiful offering of colours, shapes, and Artist's Tools lets you actually "paint" pictures on screen. You can brush boldly on the full-size Canvas or work in minute detail, pixel by pixel, using NEOchrome's Magnifier. In addition, NEOchrome is versatile enough to help you design graphs, charts, or illustrations to be used in other programs.

The **Discovery Pack Language Disk** contains the NEOchrome program, a slideshow program for viewing your NEOchrome paintings, and a library of NEOchrome picture files that you can use as models or as pictures to practice on.

"NEO" means new; "chrome" means colour. NEOchrome is the new, easy way to create high-quality, colourful artwork on your ST Computer.

## Using the Manual

The NEOchrome User's Manual contains complete instructions for using NEOchrome. You will learn how to draw lines, boxes, and circles, fill shapes with colour, and mix text and graphics. You will also discover how to choose and change colours, save and load pictures to and from the disk drive, and create motion in your pictures for special effects.

NEOchrome is best learned by reading the chapters of this manual in order, studying the illustrations, practicing, and experimenting. The following information tells you what's in each manual section:

**Chapter 1: Getting Started** provides basic information so you can start exploring the program immediately. This chapter explains how to load NEOchrome and exit from it, save and load your paintings, and run them in a slideshow. It also identifies the components of the NEOchrome screen and briefly defines their functions.

**Chapter 2: NEOchrome Components** thoroughly explains how to use all features of NEOchrome, in the order they are displayed on the NEOchrome screen. Numerous illustrations help you learn how each component operates.

**Chapter 3: NEOchrome Techniques** explains advanced graphics techniques, step by step. The end of the chapter gives information for achieving special effects, and complete details on running and customizing NEOshow, the NEOchrome slideshow program.

**Chapter 4: System and Extended System Character Sets** describes the characters and symbols available from the ST Computer keyboard for use in NEOchrome. A table lists the commands for keying in special characters.

**Chapter 5: .NEO File Header** provides NEOchrome information for programmers.

**Glossary** helps you define terms used in this manual.

# CHAPTER 1

## GETTING STARTED

### System Requirements

- ATARI ST Computer
- ATARI MicroFloppy Disk Drive
- ATARI Colour Monitor (or colour television if your ST Computer has an RF modulator)

Set up your ATARI ST Computer system. Follow the setup instructions in the Owner's Manual supplied with your computer.

### Loading NEOchrome

NEOchrome is a low-resolution program that uses the ST Computer's 16-colour, 320 x 200 pixel mode. You can load NEOchrome from either a low- or medium-resolution GEM Desktop. To do that, follow these steps:

1. With your ST Computer turned on and the GEM Desktop displayed, insert the **Discovery Pack Language Disk** into the disk drive.
2. Double-click on the correct floppy disk icon to display a directory listing for the disk.

A:\			
128221 bytes used in 4 items			
NEO	PRG	58252	☐
NEOSHOW	PRG	5713	
PICTURE	NEO	32128	
SUNSET	NEO	32128	

Directory Window

The Directory Window lists a number of files including: NEO.PRG, NEOSHOW.PRG, and various .NEO files. NEO.PRG is the NEOchrome program; NEOSHOW.PRG is a slideshow program; and files with the .NEO extender are picture files. Your directory window may list different .NEO files than those shown here and may give different file sizes for NEO.PRG and NEOSHOW.PRG.

**Note:** For screen illustrations in this manual, NEOchrome was loaded from a medium-resolution desktop. If you load NEOchrome from a low-resolution desktop, your screen displays may differ slightly.

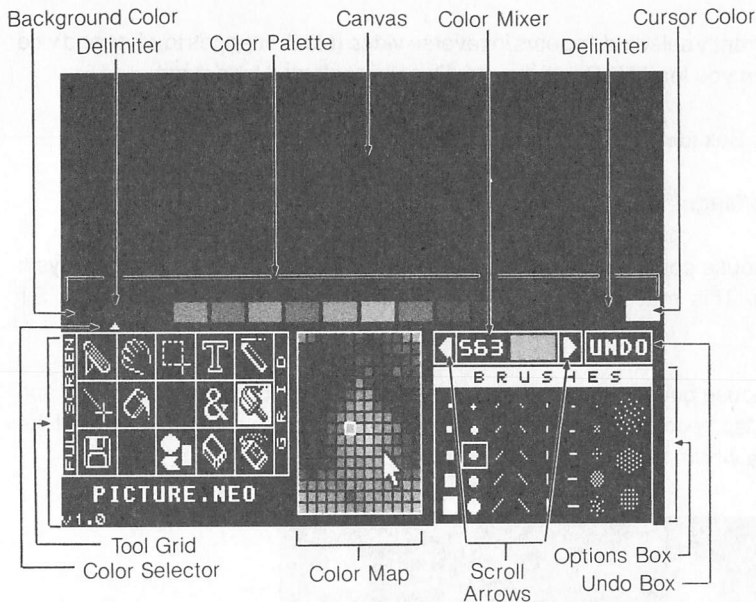
3. Double-click on NEO.PRG. The NEOchrome program screen will appear.

The NEOchrome program screen contains all the components you need to create, save, and redisplay your NEOchrome designs.

**Note:** You can install NEOchrome as a GEM application with document type .NEO. Once installed, NEOchrome will load and display a selected picture in one operation. For instructions on installing a program as a GEM application, see your ST Computer Owner's Manual.

## NEOchrome Components

The components on the NEOchrome screen are your tools for creating pictures, mixing and using colours, and saving and loading your designs.



## Canvas

The Canvas is where you draw, paint, manipulate picture sections, and write text. The Canvas can be viewed in either half size or full size. When NEOchrome first loads, its default Canvas view is half size.

## Colour Palette

The Colour Palette shows the 16 colours available for painting, and is used to change the colour you're working with, NEOchrome's default Palette shows black as the background colour (far left position), white as the cursor colour (far right position), white as the cursor colour (far right position), and 14 hues representing the full colour spectrum. Red and magenta are the Delimiters (their ends are pointed), and red is also the currently selected colour (a small marker appears under it).

## Tool Grid and Options Box

The Tool Grid holds the Artist's Tools. It also contains the Full Screen and Grid features, shows the filename of the picture on the Canvas, and identifies the cursor's position on the Canvas with x/y coordinates. When NEOchrome first loads, the default filename is PICTURE.NEO, and the x/y coordinates are turned on.

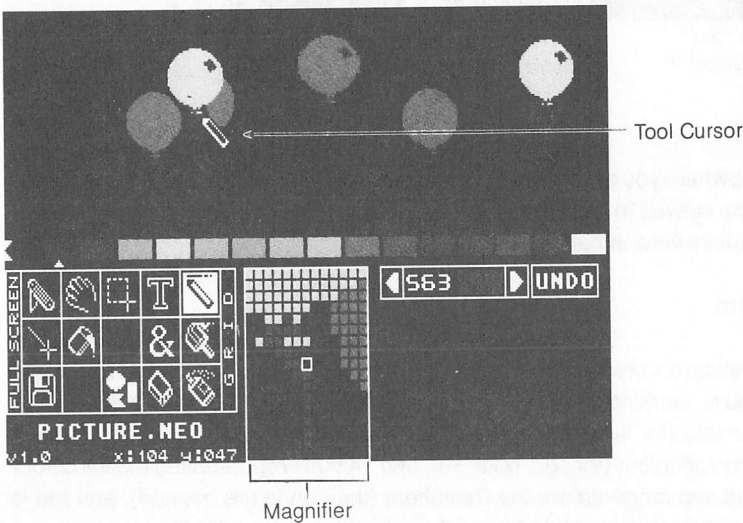
The Tool currently selected appears in reverse video (black changes to white and vice versa). When you load NEOchrome, the Pencil is selected by default.

The Options Box identifies the selected Tool and its available options.

### Colour Map/Magnifier

When the mouse pointer is on or below the Colour Palette, NEOchrome displays a Colour Map. This map is your guide to the 512 colours available from the ST Computer.

When the mouse pointer is moved above the Palette it becomes a Tool cursor, and the Colour Map becomes a Magnifier. The Magnifier shows an enlarged view of the Canvas area where the cursor is positioned.



The Colour Mixer shows a selected colour and its RGB (red, green, blue) value. By changing the RGB value, you can mix any one of NEOchrome's 512 colours and look at it on a swatch before adding it to your Palette.

### Scroll Arrows

The Scroll Arrows are used to scroll the Palette colours one step at a time. They also let you create and control colour cycling.



## Undo Box

The Undo Box lets you restore the Canvas to the way it appeared before your last graphics operation.

## Painting a Picture

You're now ready to start painting with NEOchrome. Move the mouse pointer above the Colour Palette onto the Canvas. Notice that the pointer becomes a Pencil cursor because the Pencil is selected in the Tool Grid.

Hold down the left mouse button and move the Pencil around. The Pencil will draw red lines because red is the selected colour in the Palette.

Try changing colours and Tools. To select a different colour from the Palette, point to it and click the left mouse button. The Selector marker will move under that colour. To select another Tool, point to it in the Tool Grid and click the left mouse button. The selected Tool will appear in reverse video.

To use a Tool, put its cursor on the Canvas, then hold down the left mouse button and move the mouse.

**Note:** Some Tools use either the left or right mouse button depending on the procedure.

When you're ready to learn more about using NEOchrome, read **Chapters 2 and 3**.

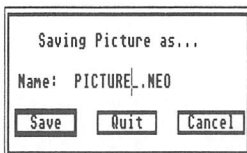
## Saving and Loading Pictures

At any time, you can save your current picture to disk, or load another picture onto the NEOchrome screen. Both these processes use the Disk Tool.

### Saving Pictures

To save the picture currently on the NEOchrome screen, follow these steps:

1. Click the left mouse button on Disk in the Tool Grid. A Dialog Box appears:



**Note:** Dialog Boxes are displayed in the resolution you loaded NEOchrome from. Dialog Boxes displayed in low resolution temporarily change the Colour Map and the far right colour in the Palette, but do not permanently affect your NEOchrome picture.

2. If you wish, type a different picture name on the Name line. Use the **[Backspace]** or **[Esc]** key to erase the current name, then type in a new name using up to eight characters. NEOchrome picture files must have a **.NEO** extender.

**Note:** If you don't change the name, your picture will be saved by whatever name currently appears in the Dialog Box. If a picture already exists by that name, NEOchrome will display a Dialog Box asking how you want to proceed. Click on Yes (or press **[Return]**) to continue.

3. Click on Save (or press **[Return]**).

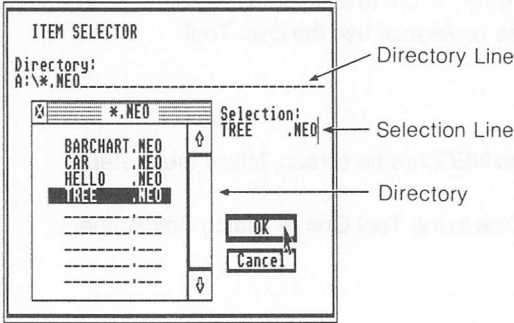
Your picture will be saved to the same disk (and folder, if open) that you loaded NEOchrome from. To save your picture to a different disk, type in the disk identifier followed by a colon before typing in a new name (for example, **B:SUNSET.NEO**).

**Note:** Make sure the disk you are saving to is not write-protected.

### Loading Pictures

The **Discovery Pack Language Disk** contains several **.NEO** files. These are picture files, both those that were included on the Language Disk and those you create using NEOchrome can be loaded onto the NEOchrome screen using the following steps:

1. Click the right mouse button on Disk in the Tool Grid. The Item Selector Box appears:



2. Point to the picture's filename in the Directory and click the left mouse button. Then click on OK (or press **[Return]**). You can also double-click on a filename, or type a filename on the Selection line and press **[Return]**.

The selected picture will be loaded onto the Canvas.

To see a directory of NEOchrome pictures held on a different disk, follow step 1. When the Item Selector Box appears, click the left mouse button on the Directory line. Use the [Backspace] key to erase the line, then enter the correct disk pathname. (For example, to see the .NEO files from Drive B, type "B:\\*.NEO".) Move the pointer into the Directory and click the left mouse button. The new directory will be displayed. Then load a file by following step 2.

**Note:** If you try to load a picture while another painting is in memory, NEOchrome will display a Dialog Box asking how you want to proceed. Clicking on Yes (or pressing **[Return]**) at this point displays the Dialog Box for saving pictures. To continue, follow the steps in Saving Pictures in this chapter.

## Printing Pictures

If you have installed a printer from the GEM Desktop, you can print your NEOchrome pictures. For complete instructions on installing a printer, see your ST Computer Owner's Manual.

To print a NEOchrome picture, follow these steps:

1. Turn your printer on and adjust the printer paper.
2. Load a picture onto the NEOchrome Canvas.
3. Move the mouse pointer to the Tool Grid and click on Full Screen. Your Canvas will appear in full size.

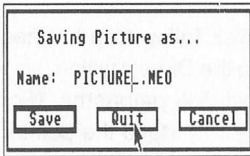
**Warning:** Do not print a NEOchrome picture in half size.

4. Move the mouse pointer into a corner of the Canvas so it does not appear on the printout.
5. Hold down the **[Alternate]** key on your keyboard, then press **[Help]**.

Your printer will print your NEOchrome picture.

## Exiting NEOchrome

To exit NEOchrome and return to the GEM Desktop, click the left mouse button on Disk in the Tool Grid. When the Dialog Box appears, click the left mouse button on Quit.



The NEOchrome program closes and the screen redisplayes the GEM Desktop.

**Note:** After you click on Quit, NEOchrome may display a Dialog Box asking if you want to save your picture. If you do, click on Yes (or press **[Return]**), then follow the instructions under Saving Pictures in this chapter.

## Running NEOshow

NEOshow is NEOchrome's slideshow program. It lets you display your paintings one by one as a slideshow. To use NEOshow, return to the GEM Desktop, display a directory listing for your Language Disk, and double-click on the file `NEOSHOW.PRG`. The program will load and display all `.NEO` (picture) files it finds on the disk, one at a time.

Press **[Control] [S]** to pause the slideshow while it is running; press **[Control] [Q]** to resume it. Press **[Space Bar]** to exit the program and return to the GEM Desktop.

You can customize a NEOslideshow, determining what pictures will be displayed, in what order, and for how long. For complete information on running and customizing NEOshow, see **Running a NEOchrome Slideshow** in **Chapter 3**.

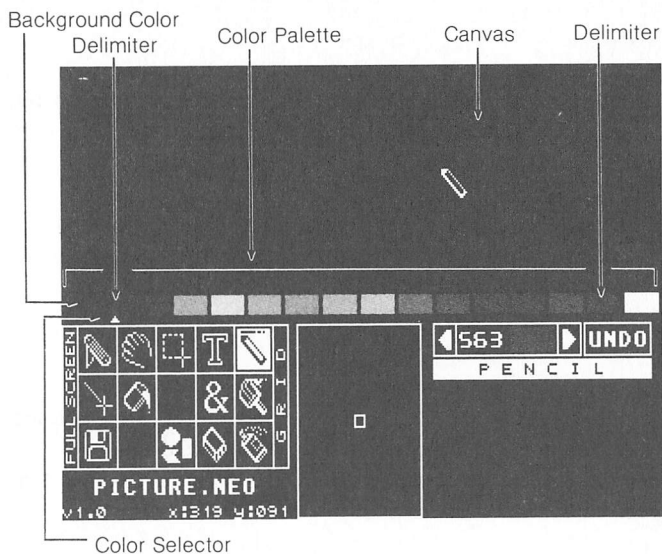
## CHAPTER 2

# NEOCHROME COMPONENTS

This chapter explains in detail how to use all the components of NEOchrome, in the order they are displayed on the NEOchrome screen.

### Canvas

The Canvas is where you create and modify your NEOchrome pictures. You work on the Canvas by moving the ST Computer's mouse and pressing the mouse buttons. Each point on the Canvas is a pixel (picture element), the smallest point of resolution



on a display monitor.

The Canvas can be used in half size or full size. (To find out how to use the full-size Canvas, see **Full Screen** in this chapter.) Press the **[Esc]** key to toggle between the half- and full-size Canvas.

## Colour Palette

The Colour Palette shows the 16 colours currently chosen for your picture. To use a colour, click on it with the left mouse button. The Colour Selector, a small marker below the Palette, moves to point at that colour, which will now appear on the Canvas as you draw or paint.

You can change the Palette colours by using the Colour Map or Colour Mixer. Select the Palette colour you want to change by clicking on it with the left mouse button. The Colour Selector will move under that colour. Then double-click the left mouse button on a colour in either the Map or the Mixer. (**See Changing Colours in the Palette in Chapter 3** for complete instructions.)

When you change a colour in your Palette, the same colour change takes place in your picture. To restore the previous colour in your Palette and on the Canvas, select the Miscellany Tool, then click on colour UNDO in the Options Box.

Three colours on the Palette have special uses; they are the background colour and the two Delimiters.

### The Background Colour

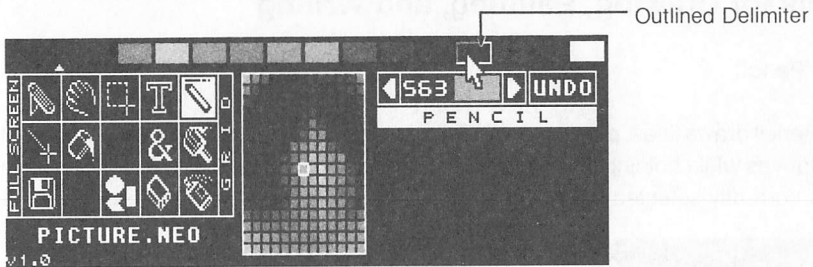
The background colour is on the left end of the Palette. It is the colour of the Canvas, and other colours are painted over it. It is also the colour that surrounds the Canvas, framing your picture.

The background colour becomes transparent in picture sections you define with the Jack Knife or Copy Box in Transparent mode. (See Jack Knife and Copy Box in this chapter for complete information.)

### Delimiters

The Delimiters are the two pointed colour blocks in the Palette. They enclose the colour range used for ramp lines and colour cycling. The Delimiters also identify the two colours used in colour swap and colour substitution. The colour of the Delimiter pointing left is the border colour for filled Shapes.

To move a Delimiter in the Palette, point to it and hold down the right mouse button. An outline appears around the Delimiter. Drag the Delimiter to another colour in the Palette, then release the button. The new colour becomes the Delimiter. You can put the Delimiters anywhere on the Palette, even on the same colour.



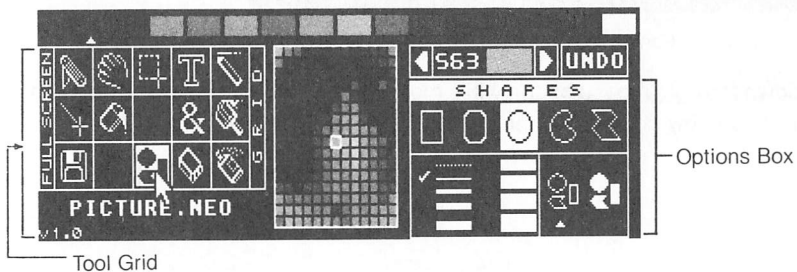
However, you can't cross one Delimiter over the other.

**Note:** The Colour Palette you set up for your picture, including new colours and Delimiter positions, is saved with your picture when you save it to disk.

## Tool Grid

The Tool Grid holds the Artist's Tools you use to create pictures. Each Tool has its own symbol. Some of the Tools draw and paint; others move the Canvas or the images on it. Most Tools are used singly; a few of them work with other Tools.

To select a Tool, point to it in the Tool Grid and click the left mouse button. Selected



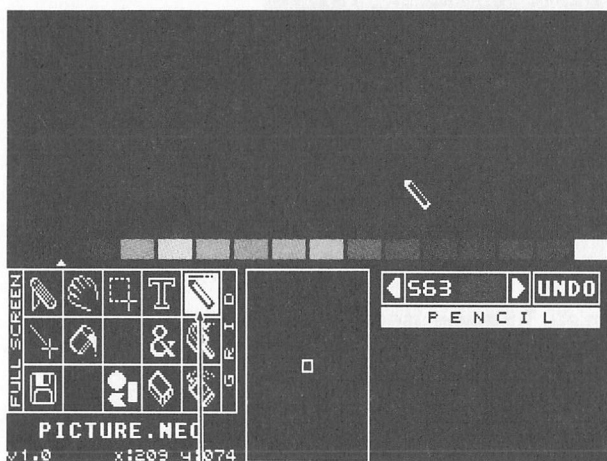
Tools appear in reverse video (black reverses to white and vice versa). For many Tools, an Options Box appears, identifying the Tool and its available options. Options are selected in the same way as Tools: by pointing to them and clicking the left mouse button.

After selecting a Tool (and an option, if needed), move the mouse pointer onto the Canvas. The pointer will change into the symbol for the Tool being used. When you move the Tool down to the Colour Palette or below it, the mouse pointer reappears.

## Tools for Drawing, Painting, and Writing

### Pencil

The Pencil draws lines, point by point. To draw, select the Pencil and move it around the Canvas while holding down the left mouse button. Points and lines will be drawn in the currently selected Palette colour.



Pencil

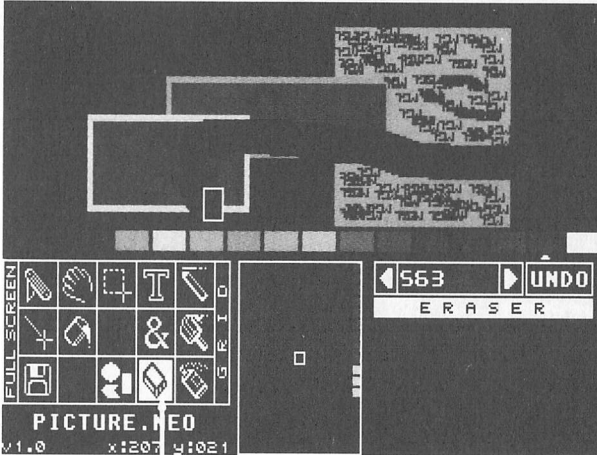
Hold down the right mouse button while moving the Pencil to draw in the background colour. If you now go over anything drawn in another colour, this “erases” it.





## Eraser

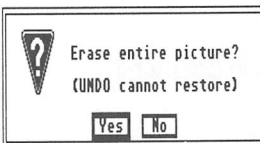
The Eraser erases colour from the Canvas, leaving only the background colour. To erase, select the Eraser and sweep it over the Canvas while holding down either mouse button. Or, position the Eraser on the Canvas and click either mouse button to erase only what's under the Eraser.



Eraser

To restore the erased portion of a picture, click on the Undo Box (or press the **[Undo]** key).

To erase everything on a half-size Canvas in one step, double-click the left mouse button on the Eraser symbol in the Tool Grid. Double-click the right mouse button on the symbol to erase the full-size Canvas at once. A Dialog Box appears:



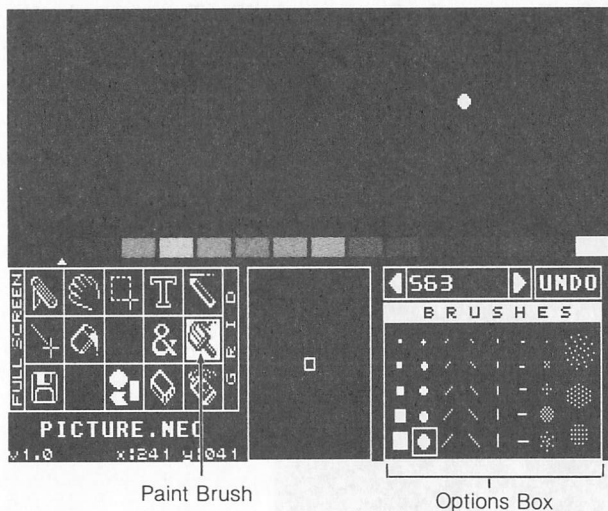
The Dialog Box reminds you that once the full-size Canvas is erased, you cannot retrieve that picture with the Undo Box. To preserve the picture, select No and save the picture, then erase. (To save a picture, see **Saving Pictures** in **Chapter 1**.)

After erasing the half- or full-size Canvas, NEOchrome automatically reselects the Tool you last used before using the Eraser.



## Paint Brush

The Paint Brush brushes paint on the Canvas. When you select the Paint Brush, an Options Box appears showing the 38 Brushes available.



Select a Brush from the Options Box (an outlined box appears around the selected Brush). Then paint by sweeping the Tool across the Canvas while holding down the left mouse button. The Brush will paint in solid strokes.

Paint in multicolours by holding down the right mouse button while using the Paint Brush. Your Brush strokes will paint all the colours enclosed by the two Delimiters in the Colour Palette, flowing in order from left to right and then repeating.

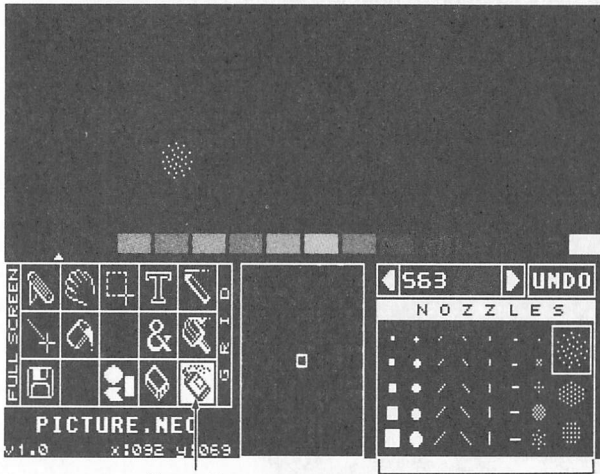
You can stamp a single Brush dab on the Canvas by holding the Paint Brush still and clicking either mouse button.

**Note:** To create your own brushes, see **Custom Brushes** in **Chapter 3**.



## Spray Can

The Spray Can sprays paint on the Canvas. When you select the Spray Can, an Options Box appears showing the 38 nozzles available.



Spray Can

Options Box

Select a Nozzle from the Options Box (an outlined box appears around your selection). Then paint by sweeping the Spray Can across the Canvas while holding down the left mouse button. You will paint in an intermittent spray.

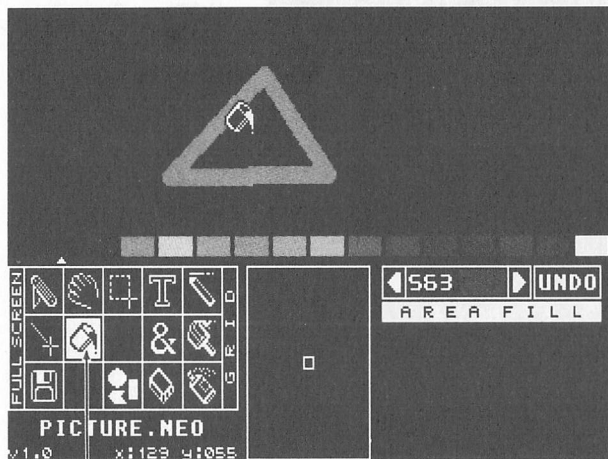
Spray in multicolours by holding down the right mouse button while using the Spray Can. Your sprays will paint all the colours enclosed by the two Delimiters in the Colour Palette, flowing in order from left to right and then repeating.

Quickly sweeping the Spray Can over the Canvas paints colour lightly. Spraying in close, slow strokes covers an area with solid colour. Holding the Spray Can still and clicking either mouse button sprays a single dab.



## Paint Bucket

The Paint Bucket “pours” colour on the Canvas, filling an area. To use the Bucket, move it onto the Canvas until the “drop” from the Bucket is within the colour area to be filled. Then click the left mouse button to pour the colour.



Paint Bucket

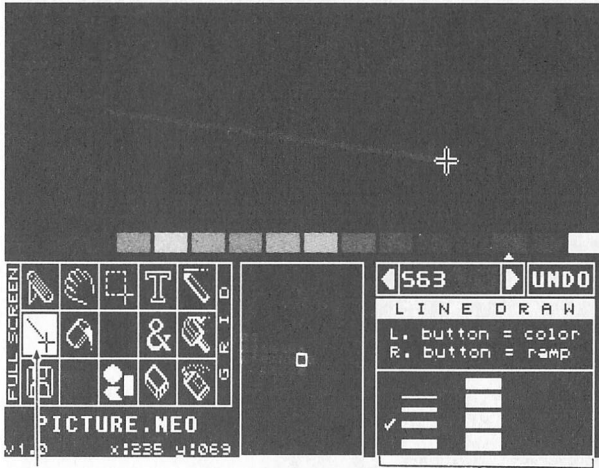
**Note:** Use your Magnifier for exact positioning of the drop within small areas. See **Colour Map/Magnifier** in this chapter for more information.

The area you are filling must be completely outlined by other colours or the paint will leak out. When this happens, click on the Undo Box (or press the **[Undo]** key), shore up the break with another colour, then repour.



## Line Draw

Line Draw draws a straight line from one chosen point (pixel) on the Canvas to another. The Options Box shows eight line-thickness options.

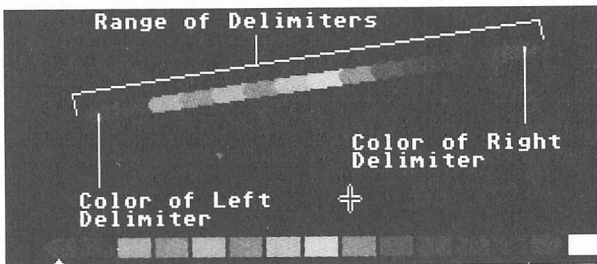


Line Draw

Options Box

To draw a line, position the Tool at a starting point on the Canvas. Then hold down the left mouse button, move the Tool to an ending point, and release the button.

Holding down the right mouse button while drawing lines adds a ramp effect. Instead of one-colour, the lines will be multicoloured, containing all colours enclosed by the Delimiters in the Colour Palette. The colours will “ramp” along the line from the colour of the Left Delimiter to the colour of the Right Delimiter.

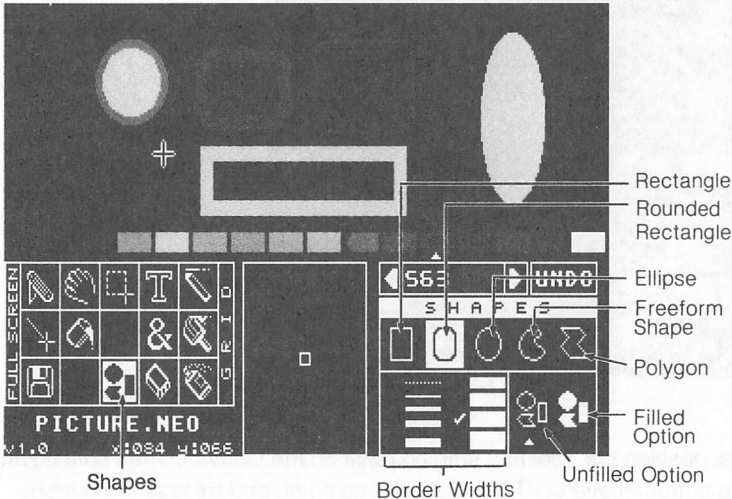


To cancel a line being drawn, click the opposite mouse button before releasing the button used to draw the line.



## Shapes

The Shapes Tool helps you create rectangles, rounded rectangles, ellipses, freeform shapes, and polygons. Shapes can be filled or unfilled; filled shapes can be bordered or borderless. Shapes are drawn in the selected Palette colour; borders are drawn in the colour of the Left Delimiter.

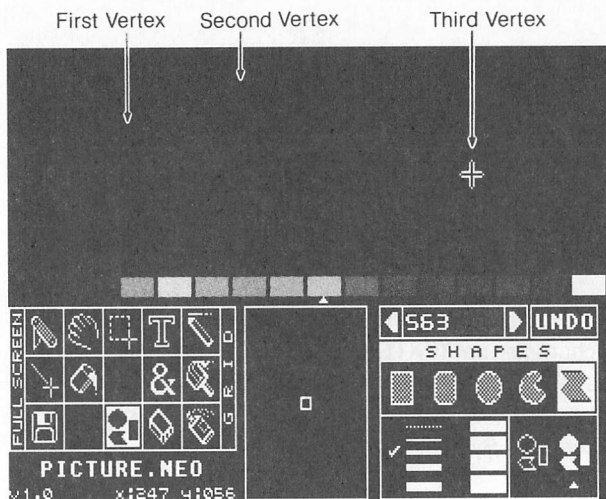


The Options Box contains five different Shapes, a dotted line for drawing borderless Shapes, eight different border widths, and the filled/unfilled options. The Shape you select appears in reverse video. The dotted line or border width selected is indicated by a tick. A pointer shows whether the Filled or Unfilled option has been selected.

To draw rectangles or ellipses, first select your options, then position the Tool at a starting point on the Canvas, hold down the left mouse button, and move the Tool in any direction. Moving the Tool expands or contracts your shape. When you've drawn the shape you want, release the mouse button.

To draw a freeform shape, hold down the left mouse button and draw. When you release the button, NEOchrome will connect the first and last strokes of your shape with a straight line and fill the shape with colour (unless it is an unfilled shape).

To draw a polygon, click the left mouse button at a spot on the Canvas. This sets the shape's first vertex. Then move the Tool (without holding down the button) to draw a line from the first vertex to the second. Click the left mouse button to set the second vertex. Again draw a line, then click the left mouse button, setting the third vertex.

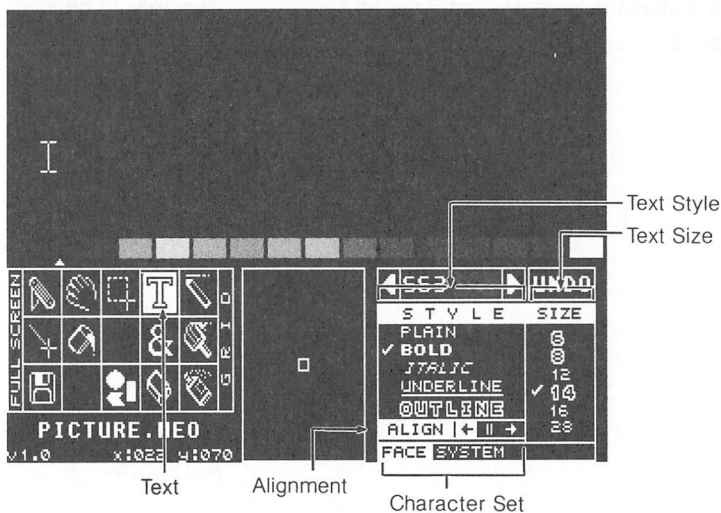


Continue in this way to the last vertex and click the right mouse button. NEOchrome will connect the first and last vertices with a straight line and fill the shape (unless the unfilled shapes option was selected).

**Note:** Filled freeform shapes have a maximum size. If you reach that size while drawing, NEOchrome will automatically close the shape and fill it.

## Text

Text inserts letters, numbers, and symbols in your picture. The Options Box contains five different text styles, six different text sizes, three text alignment options, and two character set options.



Select a text style and size by pointing at the desired option and clicking the left mouse button. Bold, Italic, Underlined, and Outlined text can be used in combination; Plain text must be used alone. The text size is the height in pixels of the largest character in the text you will type. Ticks will show your selections.

Select a text alignment to centre your text ( || ) or align it left ( ← ) or right ( → ). Your choice appears in reverse video.

Click in the box next to FACE to toggle between the System character set (SYSTEM) and the Extended System character set (EXTENDED SYS). Characters from the chosen system will appear on the Canvas when you type from the keyboard. (Most System characters are printed on the ST Computer keyboard; see **Chapter 4** for a list of special System and Extended System characters).

To write text, position the Tool on the Canvas, click the left mouse button, then move the Tool aside. A flashing cursor stays in the spot where you clicked. Then type your text from the keyboard. Centered text will be centred on the flashing cursor; left and right aligned text will move to either side. Press the **[Return]** key to start a new line in the same size and style.

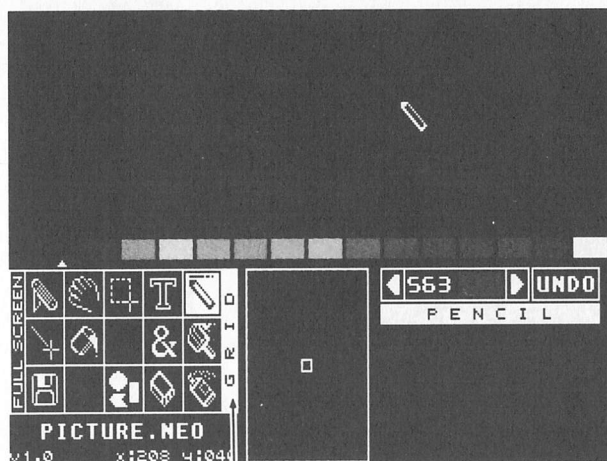
You can change the colour, style, size, and alignment of all text as long as you haven't repositioned the flashing cursor. To make changes, select another colour from the Colour Palette or other options in the Options Box (selecting the Plain option is a quick way to clear text styles.) Your text will change when the option is selected. Toggling between System and Extended System will not affect text already keyed in.

To use more than one colour, style, or size within a line, type to where you want the change, then reposition the flashing cursor on the same line. Change your options and continue typing.



## Grid

When selected, Grid makes the Tool you're using "jump" to grid points as it is moved on the Canvas. Grid points are every eighth pixel on the Canvas, and their coordinates are always evenly divisible by eight.



Grid

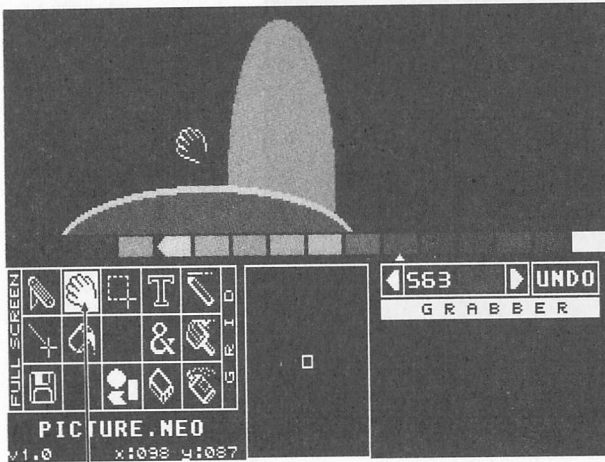
Grid works in conjunction with other Tools, and does not display an Options Box.

## Tools for Moving the Canvas and Picture Sections



### Grabber

The Grabber moves the half-size Canvas up and down, and identifies colours painted on the Canvas.



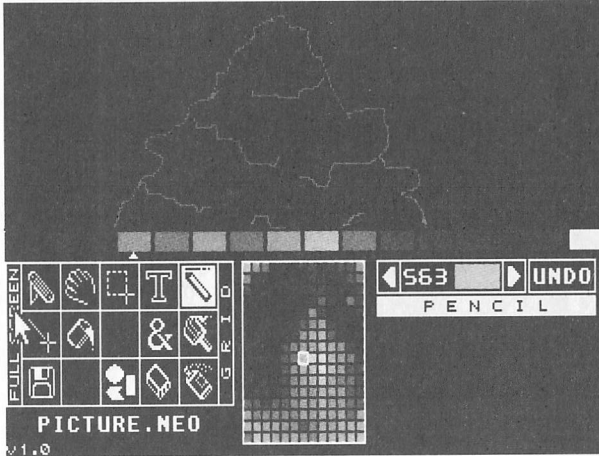
Grabber

Select the Grabber and move it onto the Canvas, hold down either mouse button, and move the Tool up or down. The Canvas will move with the Tool. Then release the mouse button. Your Canvas will stay in its new position.

To identify a colour on the Canvas, move the Grabber to that colour (watch the Magnifier for exact positioning in small colour areas). Click the right mouse button. The Colour Selector will reposition under that colour in the Palette so you can see exactly which colour it is.

## Full Screen

Selecting Full Screen displays the full-size Canvas, hiding the NEOchrome components. This lets you see and work on the whole Canvas at once.



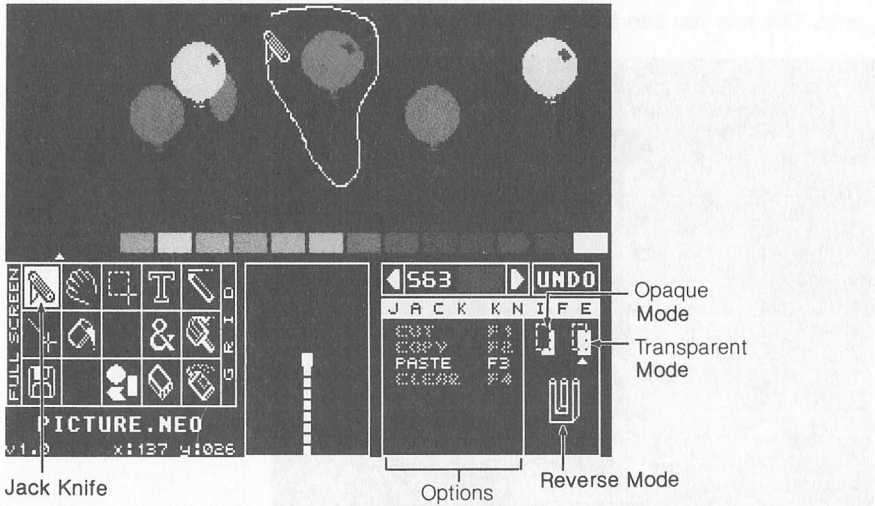
Half-Size Canvas

To redisplay the NEOchrome components, move the Tool as far as possible into any corner (except the upper left corner) of the Canvas and click the left mouse button. Or press the **[Esc]** key to toggle between the half and full-size Canvas.



## Jack Knife

The Jack Knife defines a freeform picture area in order to move or erase it.



Jack Knife

Options

Reverse Mode

Opaque Mode

Transparent Mode

Select the Jack Knife, then move it onto the Canvas, holding down either mouse button and “cutting” around an image or picture area. Then release the mouse button. A cycling line marks where you cut. Since the Jack Knife does not cut out background colour, the cycling line will shrink to fit around the closest colour other than background.

To move the section, push the tip of the Jack Knife into it. When the tip is in the section, the Jack Knife becomes an arrow. Now hold down the left mouse button and drag the section anywhere else on the Canvas. If you click the right mouse button first, then hold down the left mouse button, you can stamp the section in its original spot and move a copy of it.

When the section is where you want it, release the mouse button. Then move the Tool out of the section and click the left mouse button to erase the cycling line.

Holding down the right mouse button while dragging the picture area lets you “spray” with it. Moving the mouse quickly paints a few copies of the area along the way; dragging the mouse slowly paints many copies in the pattern of the mouse’s motion.

The Jack Knife Options Box contains four options and three modes.

The options are available when they appear in bold letters. Select an option by clicking on it or pressing the corresponding function key on the keyboard. Click on a mode to select it; a marker indicates your choice.

CUT (**[F1]**) removes the defined area and stores it in the Cut Buffer.

COPY (**[F2]**) leaves the defined area on the Canvas and stores a copy of it in the Cut Buffer.

PASTE (**[F3]**) pastes the image from the Cut Buffer onto the centre of the Canvas.

CLEAR (**[F4]**) erases the defined area without storing it in the Cut Buffer.

Opaque mode lets you move the defined area over the picture, covering up anything behind it.

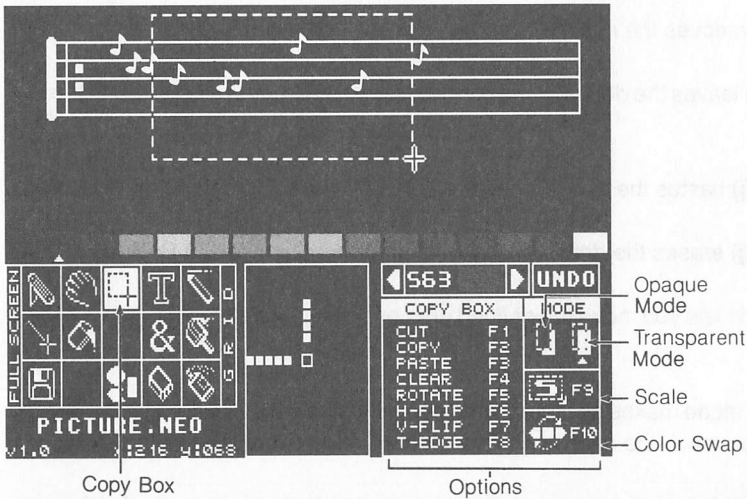
Transparent mode makes background colour within the defined area transparent, so that when you move the area, anything behind it shows through the "holes."

Reverse mode lets you move the defined area behind other images on the Canvas.



## Copy Box

The Copy Box defines a rectangular picture area and manipulates it.



Select the Copy Box and move the Tool onto the Canvas to one corner of an area you want to define. Hold down the left mouse button, then drag the Tool diagonally to the opposite corner of the area. A cycling box will form around the area. Release the mouse button.

To move the picture area, point inside the box (the Tool changes to an arrow), hold down the left mouse button, and drag the area to another part of the Canvas. (If you click the right mouse button first, then hold down the left mouse button, you will leave the area in its original spot and move a copy of it.) Then click the right mouse button to stamp the image in its new location. You can repeat dragging and stamping the image as many times as you want. To erase the cycling box, move the Tool out of the box and click the left mouse button.

The Options Box contains eight options and four modes.

The options are available when they appear in bold letters. Select an option by clicking on it or pressing the corresponding function key on the keyboard.

Select a mode by clicking on it. A marker shows whether Opaque or Transparent mode has been selected; Scale appears in reverse video when selected. You can also select Scale or Colour Swap by pressing the corresponding function key.

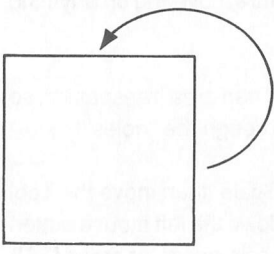
**CUT ([F1])** removes the defined area and stores it in the Cut Buffer.

**COPY ([F2])** leaves the defined area on the Canvas and stores a copy of it in the Cut Buffer.

**PASTE ([F3])** pastes the image from the Cut Buffer onto the centre of the Canvas. If you have defined a picture area, the image from the Buffer will be pasted in the area. (If you want to preserve the image in the defined area and paste the Cut Buffer image over it, click the right mouse button in the area before selecting PASTE.) If the image in the Buffer is larger or smaller than the defined area, it is scaled to fit.

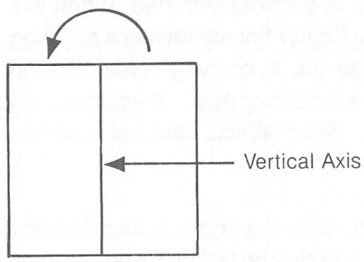
**CLEAR ([F4])** erases the defined area without storing it in the Cut Buffer.

**ROTATE ([F5])** rotates the defined area 90 degrees counter-clockwise. Four rotations takes the image through a complete circle.



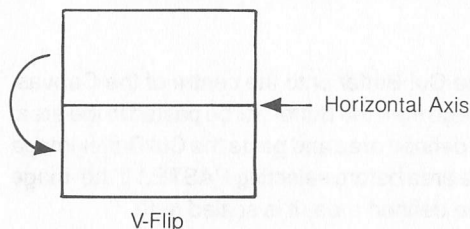
Rotate

**H-FLIP ([F6])** horizontally flips the defined area about its central vertical axis. A second flip restores the image to its original orientation.



H-Flip

**V-FLIP ([F7])** vertically flips the defined area about its central horizontal axis. A second flip restores the image to its original orientation.



**T-EDGE ([F8])** traces a border around the edges of anything in a Copy Box that was drawn with the currently selected Palette colour. The image so bordered is replaced with the background colour.

Opaque mode lets you move the defined area over the picture, covering up anything behind it.

Transparent mode makes background colour within the defined area transparent, so that when you move the area, anything behind it shows through the "holes."

**Scale ([F9])** stretches or shrinks the defined area. Select Scale, then move the Tool into the defined area (the Tool changes to an arrow). Hold down the left mouse button and drag the arrow to stretch or shrink the area. Click on Scale again (or press **[F9]**) to turn off the option. (See **Expanding and Shrinking Picture Sections** in **Chapter 3** for a thorough description of how to use Scale.)

**Colour Swap ([F10])** swaps the Delimiter colours in your defined area. When this option is selected, the Right Delimiter colour in the Colour Palette replaces anything within the defined area drawn with the Left Delimiter colour, and vice versa. Clicking on Colour Swap again (or pressing **[F10]**) swaps the colours again. If you have not defined an area with Copy Box, selecting Colour Swap affects your entire visible Canvas.

You can substitute, rather than swap, a colour. To do that, put the Colour Selector under one of the Delimiters in the Colour Palette. Then click on Colour Swap (or press **[F10]**). The selected Delimiter colour will remain in your picture, as well as replace anything drawn in the other Delimiter colour. This operation affects either the area defined with Copy Box, or the entire visible Canvas if no area was defined.



## Cut Buffer

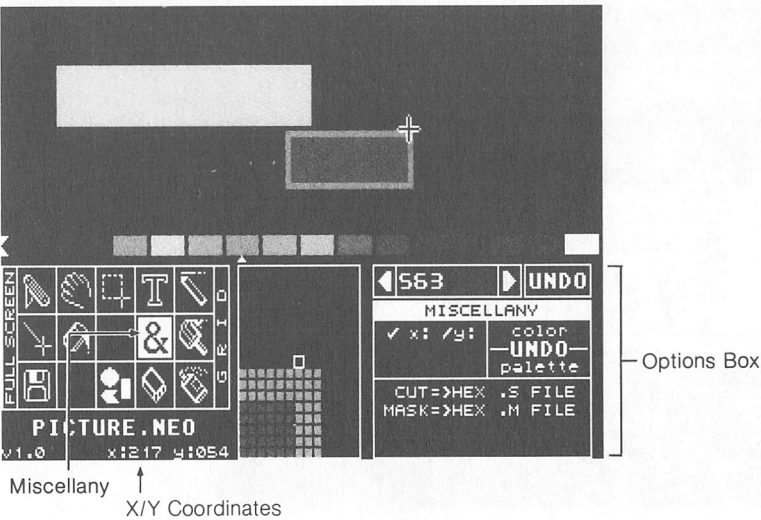
The Cut Buffer is the area of NEOchrome where a picture section is held after being cut out with the Jack Knife or Copy Box. The section can then be re-pasted into the original picture or into another picture. The image in the Cut Buffer is shared by both the Jack Knife and the Copy Box, so you can cut an image with one Tool and paste it with the other.

Only one image at a time can be held in the Cut Buffer. The Cut Buffer is emptied when you exit NEOchrome.

## Special Tools

### Miscellany

Miscellany lets you turn the x/y coordinates display on or off, and save portions of your picture in source code format for use in programming. This Tool also offers colour and palette UNDO options.



When you choose the x:y: option (indicated by a tick), a Tool's position on the Canvas is shown in x/y coordinates below the Tool Grid. (For a complete description, see **X/Y Coordinates** in this chapter.)

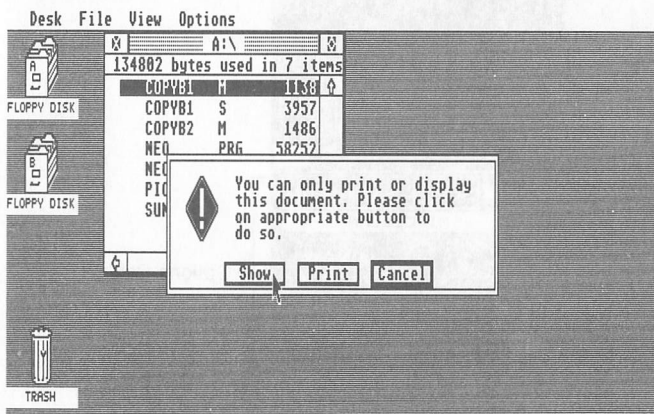
The UNDO option restores colours to their previous value. Clicking on the word “color” restores the selected Palette colour to its previous value; clicking on the word “palette” restores the entire Colour Palette to its previous value.

The Cut option, when selected, sends the contents of the Cut Buffer to an ASCII text file named COPYBn.S (NEOchrome replaces n with a number from 1 to 999).

The Mask option, when selected, sends the contents of the Cut Buffer to an ASCII text file named COPYBn.M (NEOchrome replaces n with a number from 1 to 999). The file is a monochrome mask of the graphics where pixels are either 0 for the background colour or 1 for all other (non-background) colours.

**Note:** If the Cut Buffer is empty, NEOchrome tells you so in the Options Box when you select Cut or Mask.

These .S and .M files are useful for programmers who want to incorporate NEOchrome graphics into an application. To view a file, return to the GEM Desktop by exiting NEOchrome (see **Exiting NEOchrome** in **Chapter 1**). When the disk directory is displayed, double-click on the filename you want. Then click on the buttons in the Dialog Box to either print or display the file.



You can also load these files into a word processor for editing.



## Disk

Disk lets you save the current picture to disk, load another picture onto the NEOchrome screen, or quit NEOchrome.

Select Disk with the left mouse button to save a picture or quit NEOchrome. Select it with the right mouse button to load a picture. (For complete instructions on Disk operations, see **Saving and Loading Pictures** and **Exiting NEOchrome** in **Chapter 1**.)

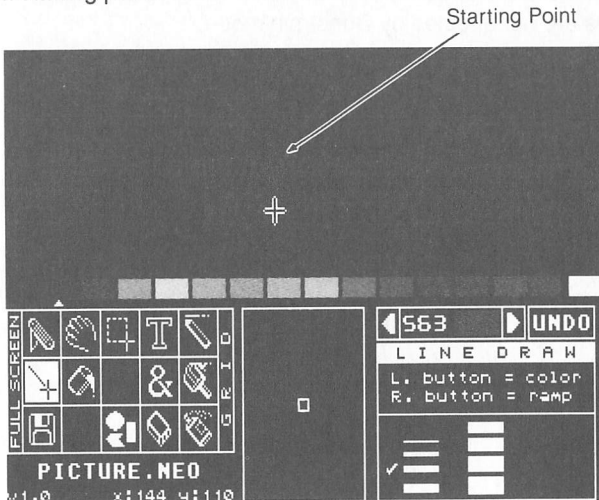
## Constrain Mode

Constrain mode holds an action to a vertical or horizontal line. To use Constrain mode, hold down either **[Shift]** key, then begin moving a tool on the Canvas. Once the Tool is moving, you can release the **[Shift]** key; the Tool stays in Constrain mode until you release the mouse button.

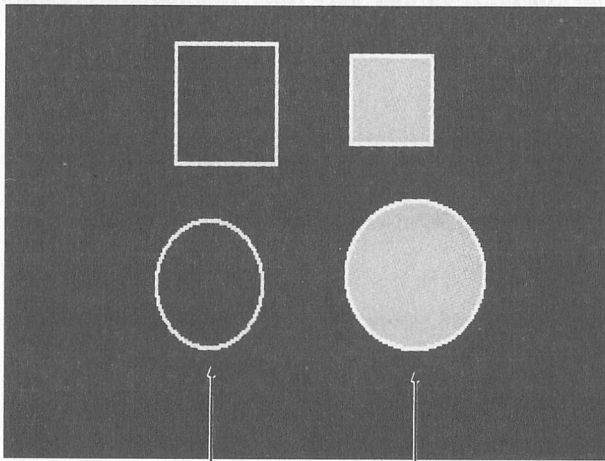
For the Pencil, Eraser, Paint Brush, and Spray Can, Constrain mode allows only horizontal or vertical painting or erasing.

For the Jack Knife and Copy Box, Constrain mode allows only horizontal or vertical movement when a defined picture section is dragged.

For Line Draw, Constrain mode allows lines to be drawn in only eight directions from a starting point.



For Shapes, Constrain mode draws rectangles as squares, and ellipses as circles. Using the left **[Shift]** key forces the shape to be the same number of pixels horizontally and vertically, resulting in true squares and circles. Using the right **[Shift]** key corrects for the 9:10 aspect ratio of the pixels, making the vertical measure of the shape .9 times its horizontal measure, resulting in shapes that appear square or circular. (For instructions on drawing squares and circles, see **Squares and Circles** in **Chapter 3**.)



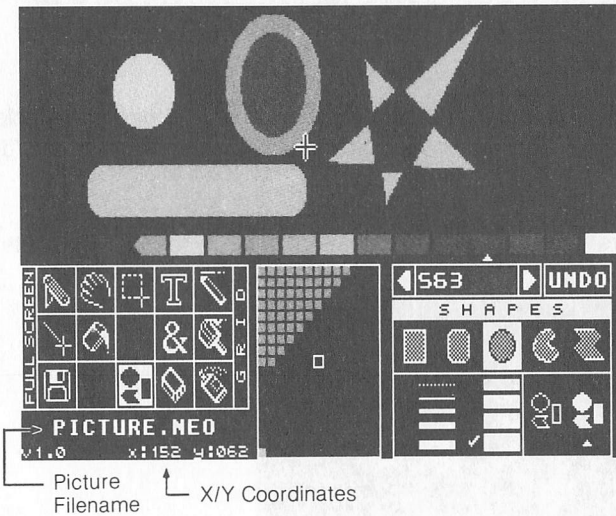
Drawn with Left **[Shift]** Key

Drawn with Right **[Shift]** Key

Polygons are affected like Line Draw: lines can be drawn in only eight directions from a vertex. Freeform shapes are not affected by Constrain mode.

### Picture Filename

The filename of the picture currently on the Canvas is displayed below the Tool Grid. If you haven't named your picture, the filename will be `PICTURE.NEO`.



## X/Y Coordinates

The x/y coordinates show a Tool's location in pixels on the Canvas. These coordinates change as you move the Tool, and are not displayed when the Tool is moved off the Canvas.

The x coordinate identifies the Tool's horizontal pixel, and can be any number from 0 to 319, starting from the left edge of the Canvas. Its value increases as you move the Tool to the right.

The y coordinate identifies the Tool's vertical pixel, and can be any number from 0 to 199, starting from the top of the Canvas. Its value increases as you move the Tool down.

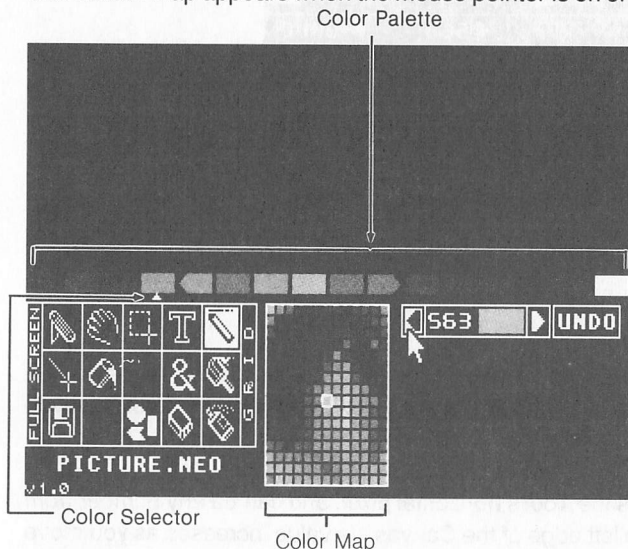
The upper left corner of the Canvas is position 0:0; the bottom right corner of the full-size Canvas is position 319:199.

You can turn the x/y coordinates display on and off with **Miscellany** (see **Miscellany** in this chapter).

## Colour Map/Magnifier

The Colour Map/Magnifier displays either an orderly map of the 512 colours available in NEOchrome, or a magnified view of your picture showing individual pixels for detailed drawing.

The Colour Map appears when the mouse pointer is on or below the Colour Palette.

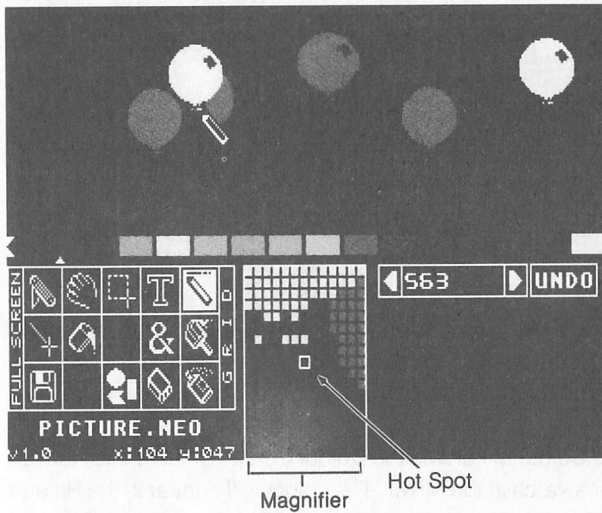


The Map contains 512 colours; 208 colours at a time are displayed. To look at all the colours, move the pointer onto the Map and hold down the right mouse button. The Grabber will appear. Move the Grabber left or right while holding down the button to view the entire range of 512 colours. The colour outlined in white in the Map is currently showing in the Colour Mixer.

You can copy a colour from the Colour Map into the Colour Palette. Point to a colour in the Map and double-click the left mouse button. The colour will appear in the Colour Palette above the Colour Selector, replacing the previous colour there and on the Canvas.

You can also put colours from the Colour Map into the Colour Mixer. Point to a colour in the Map and single-click the left mouse button. The colour will appear in the Colour Mixer.

The Magnifier appears when a Tool is on the Canvas.



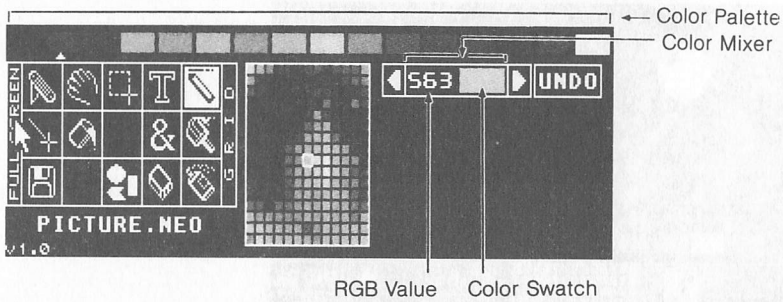
The Magnifier displays an enlarged (5X) view of the Canvas, clearly showing individual pixels. The “Hot Spot,” the outlined pixel in the centre of the Magnifier, is a Tool’s pixel position on the Canvas. By watching the Magnifier, you can position a Tool exactly where you want it on the Canvas in order to identify a colour or paint in precise detail.

## Colour Mixer

NEOchrome offers 512 colours, each combining different intensities of red, green, and blue. This colour system is called RGB (Red, Green, Blue). On NEOchrome Computer, the intensity levels of red, green, and blue are numbered from 0 (least intense) to 7 (most intense).

A colour is identified by its RGB value: three digits representing the red, green, and blue intensities, in that order, of the colour. RGB 000 is black, RGB 777 is white, and RGB 700, 070, and 007 are pure red, pure green, and pure blue, respectively. There are 512 RGB combinations ( $8 \times 8 \times 8 = 512$ ), making possible 512 colours.

The Colour Mixer displays a colour and its RGB value. You can mix colours by changing the digits in the RGB value.

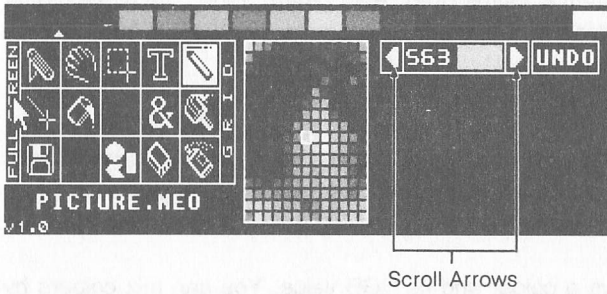


To change the RGB value, click either mouse button on a digit. Clicking the left mouse button increases the digit's value; clicking the right mouse button decreases its value. As each digit changes, the colour in the Mixer adjusts to reflect the new value.

To copy the colour from the Colour Mixer into the Colour Palette, double-click the left mouse button on the colour swatch in the Mixer. The colour will appear in the Palette above the Colour Selector, replacing the previous colour there and on the Canvas. (To copy a colour from the Colour Palette into the Mixer, double-click the left mouse button on the colour in the Palette: the colour swatch and RGB value will be updated.)

### Scroll Arrows

The Scroll Arrows let you scroll the Colour Palette left or right, colour by colour. You also use the Arrows to achieve colour cycling, which is continuous scrolling. Scrolling the Palette rearranges the positions of the Palette colours between the Delimiters, one colour at a time. To scroll, click the left mouse button on a Scroll Arrow.



The Arrow will blink and all Palette colours enclosed by the Delimiters will scroll one position in the direction of that Arrow. As a colour leaves one Delimiter, it reappears in the other Delimiter. NEOchrome will adjust the colours in your Canvas to reflect the new positions of the colours in the Palette.

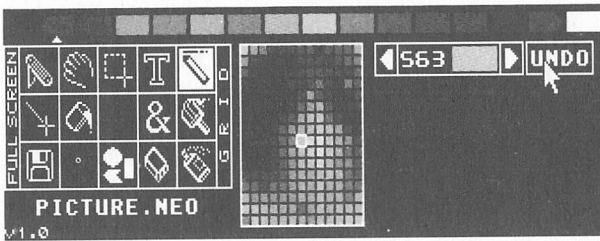


Cycling colours lets you create motion or blinking effects on the Canvas. To cycle, click the right mouse button on a Scroll Arrow. The Arrow will appear in reverse video, and all Palette colours enclosed by the Delimiters will begin cycling in the direction of that Arrow. As a colour leaves one Delimiter, it reappears in the other. The colours in your Canvas will cycle as the Palette cycles. To stop cycling, click either mouse button on the opposite Scroll Arrow.

You can also set the speed of your colour cycling. For more information, see **Colour Cycling** in **Chapter 3**.

## Undo Box

The Undo Box lets you erase (“undo”) the graphics action most recently performed on the Canvas. To undo, click the left mouse button on the Undo Box (or press the



**[Undo]** key). NEOchrome will erase only your last action; you can't “undo” any actions before that. However, you can undo an Undo operation. Click on the Undo Box again. The former look of the Canvas will reappear.



## CHAPTER 3

# NEOCHROME TECHNIQUES

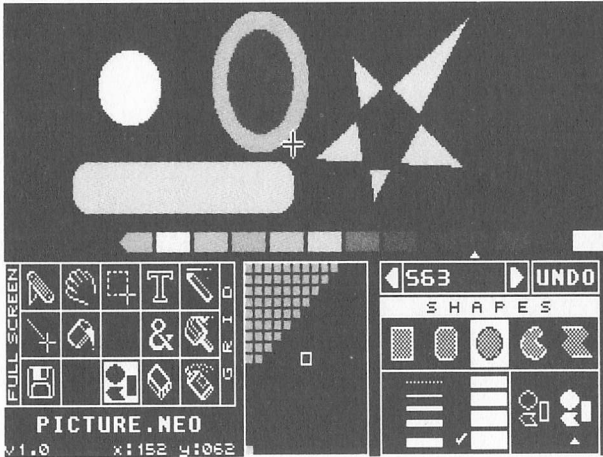
This chapter explains the more advanced graphics techniques available in NEOchrome, in step-by-step instructions. The end of the chapter gives information for achieving special effects and running the NEOchrome slideshow.

### **Drawing Shapes**

NEOchrome provides five basic shapes: rectangle, rounded rectangle, ellipse, freeform shape, and polygon. Shapes can be drawn filled (bordered or borderless) or unfilled.

### **Filled Shapes**

Filled Shapes appear as solid objects, either bordered or borderless. These Shapes are drawn in the selected Palette colour; borders are the colour of the Left Delimiter.



To draw a filled Shape, follow these steps:

1. Select Shapes from the Tool Grid and a colour from the Palette.
2. Select the Filled option by clicking on the filled Shapes in the lower right corner of the Options Box. (The Shapes across the top of the Options Box will become filled.)
3. Select a Shape by clicking on one of the Shapes across the top of the Options Box.
4. Select a border width by clicking on one of the border options along the left edge of the Options Box. For a borderless Shape, choose the dashed line. Select a border colour by moving the Left Delimiter to your colour choice.

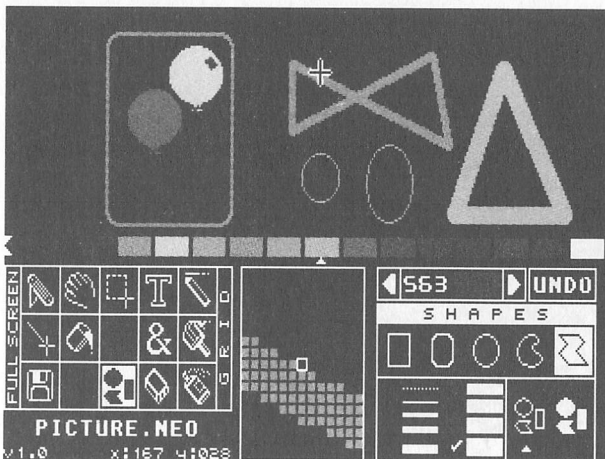
**Note:** You cannot move the Left Delimiter past the Right Delimiter. If the Right Delimiter is in the way, first drag it to the far right of the Colour Palette, then move the Left Delimiter to your desired colour.

5. Position the Tool on the Canvas, hold down either mouse button, and move the Tool in any direction.
6. When you've drawn the Shape you want, release the mouse button.

**Note:** Borderless, filled freeform shapes and polygons will have a border while being drawn, but will be correctly coloured and borderless when completed.

## Unfilled Shapes

Unfilled Shapes appear as outlines, framing the picture inside them. These Shapes are drawn in the selected Palette colour.



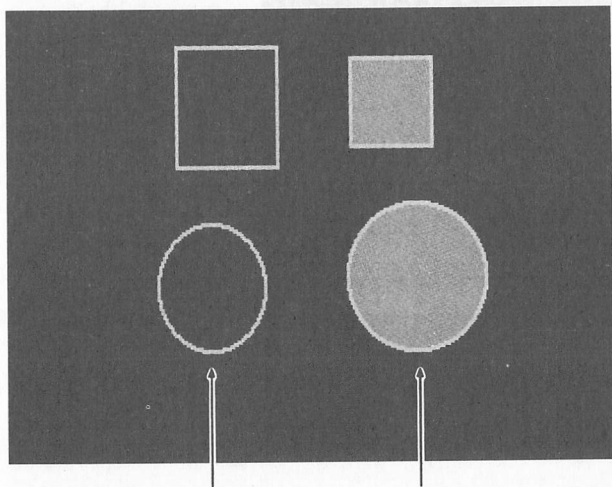
To draw an unfilled Shape, follow these steps:

1. Select Shapes from the Tool Grid and a colour from the Palette.
2. Select the Unfilled option by clicking on the unfilled Shapes in the lower right corner of the Options Box. (The Shapes across the top of the Options Box will become unfilled.)
3. Select a Shape by clicking on one of the Shapes across the top of the Options Box.
4. Select a border width by clicking on one of the border options along the left edge of the Options Box.
5. Position the Tool on the Canvas, hold down either mouse button, and move the Tool in any direction.
6. When you've drawn the Shape you want, release the mouse button.

To place an unfilled Shape on the Canvas without having the picture under it show through, first draw the Shape, then use the Paint Bucket to fill it with background colour.

## Squares and Circles

Squares and circles are drawn with the Shapes Tool and Constrain mode (the right and left **[Shift]** keys).



Drawn with Left **[Shift]** Key

Drawn with Right **[Shift]** Key

To draw squares and circles, follow these steps:

1. Select Shapes from the Tool Grid and a colour from the Palette.
2. Select your options from the Options Box. For squares, choose the rectangle; for circles, choose the ellipse.
3. Press and hold down the right or left **[Shift]** key to activate Constrain mode. Holding down the left **[Shift]** key gives you a true circle or square with the same number of pixels in its vertical axis as in its horizontal axis. Holding down the right **[Shift]** key makes the vertical axis of the shape .9 times its horizontal axis, correcting for the 9:10 aspect ratio of the pixels and resulting in a shape that appears square or circular.
4. Position the Tool on the Canvas, hold down either mouse button, and move the Tool in any direction. (Once you begin moving the Tool, you can release the **[Shift]** key.)
5. When you've drawn the square or circle you want, release the mouse button.

## Erasing with Painting Tools

Painting Tools (Paint Brush, Spray Can, Line Draw, Paint Bucket, and Shapes) can also be used as erasers by painting in the background colour. To erase with a painting Tool, follow these steps:

1. Select a painting Tool from the Tool Grid and a size or shape from the Options Box. For Shapes, choose the borderless and filled options.
2. Select the background colour in the Colour Palette.
3. Move the Tool onto the Canvas and erase by holding down the left mouse button and painting in the background colour.

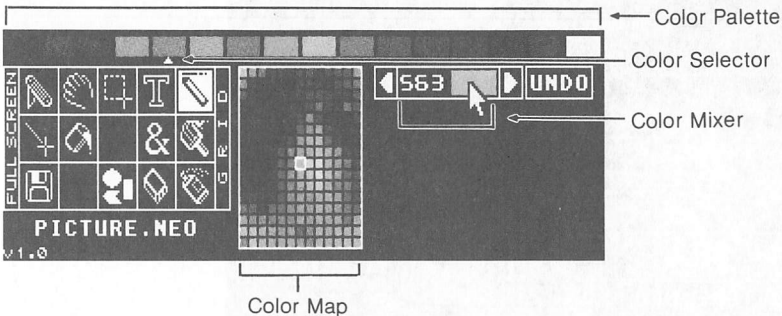
To restore the picture, click on the Undo Box (or press the [Undo] key).

## Working with Colours

### Changing Colours in the Palette

You can change Palette colours by using either the Colour Mixer or the Colour Map. To do that, follow these steps:

1. Click on a Palette colour you want to change. The Colour Selector will move under that colour.
2. To copy the colour currently in the Colour Mixer into the Palette, move the mouse pointer onto the colour swatch in the Mixer and double-click the left mouse button.



3. To copy a colour from the Colour Map into the Palette, point at the desired colour in the Map and double-click the left mouse button. (To view all colours in the Map, move the pointer onto the Map, then press and hold down the right mouse button. When the Grabber appears, move it left and right.)

The new colour replaces the previous colour in the selected Palette position and on the Canvas. To restore the previous colour, select Miscellany from the Tool Grid, then click on colour UNDO in the Options Box.

## Copying a Palette Colour

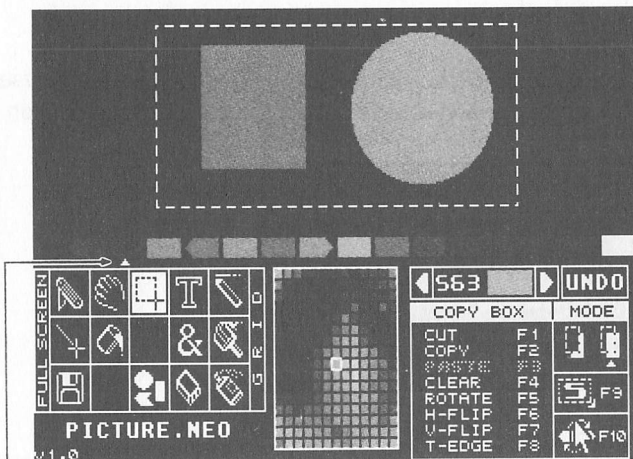
You can copy a colour from one Palette position into another (this is especially useful when setting up colour cycling). To do that, follow these steps:

1. In the Palette, double-click the left mouse button on the colour to be copied. That colour appears in the Colour Mixer.
2. Single-click the left mouse button on the Palette location you want to copy the colour into. The Colour Selector moves under that position.
3. Double-Click the left mouse button on the colour swatch in the Colour Mixer.

The colour is copied from the Colour Mixer into the selected Palette position. You now have the same colour in two different locations in the Palette.

## Swapping Colours

Colour Swapping exchanges two colours on the Canvas, so that each colour replaces the other.



Color Selector



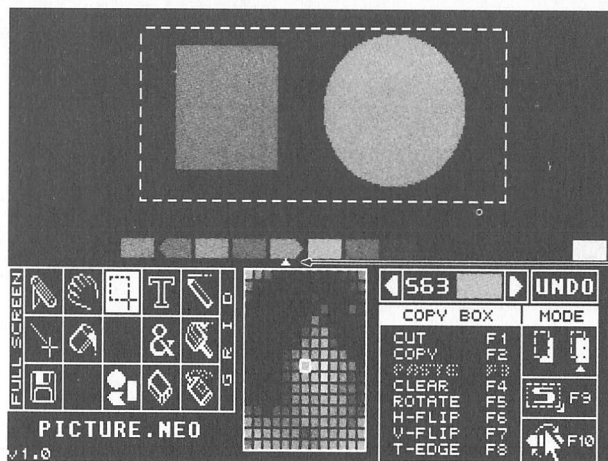
To swap colours, follow these steps:

1. Move the Left and Right Delimiters to the two Palette colours you've drawn with.
2. Make sure the Colour Selector is NOT under one of the Delimiters.
3. Choose Copy Box from the Tool Grid.
4. Define a Copy Box around the area you want to affect.
5. Click the left mouse button on Colour Swap mode in the Options Box (or press **[F10]**). The two Delimiter colours will replace each other in the Copy Box.

You can swap colours throughout the visible Canvas. To do that, omit step 4 (do not define a Copy Box).

## Substituting Colours

Colour Substitution replaces one colour on the Canvas with another. For example, if you draw a square in one colour and a circle in another, you can then repaint the circle in the same colour as the square.



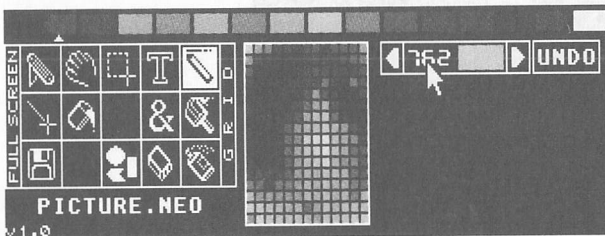
To substitute colours, follow these steps:

1. Move the Left and Right Delimiters to the two colours you've drawn with.
2. Decide which of the two colours will be the Source Colour (the colour you will substitute with). Click the left mouse button on that colour to select it with the Colour Selector. The other colour becomes the Destination Colour (the colour to be replaced with the Source Colour). If the Colour Selector is NOT placed under one of the Delimiter colours, you will get colour swapping instead of colour substitution.
3. Choose the Copy Box from the Tool Grid.
4. Define a Copy Box around the area you want to affect.
5. Click the left mouse button on Colour Swap in the Options Box (or press **[F10]**). The Source Colour replaces the Destination Colour in the Copy Box.

You can substitute one colour for another throughout the visible Canvas. To do that, omit step 4 (do not define a Copy Box).

## Using the Colour Mixer

You can mix any of NEOchrome's 512 colours in the Colour Mixer.



To mix a colour in the Mixer, follow these steps:

1. Point the cursor at one of the digits in the Colour Mixer. (The digits represent the red, green, and blue components of the colour in the Mixer. Values range from 0, least intense, to 7, most intense.)
2. Press the left mouse button to increase the digit; press the right mouse button to decrease it. The colour in the Mixer will adjust to reflect the new value.

3. To copy a colour from the Mixer into the Colour Palette, double-click the left mouse button on the colour swatch in the Mixer. The Mixer colour will appear in the selected Palette position, replacing the previous colour there and on the Canvas.

Working with a colour in the Mixer does not affect the Palette or the Canvas until you move the Mixer colour into the Palette. To restore the previous Palette colour, select Miscellany from the Tool Grid, then select colour UNDO from the Options Box.

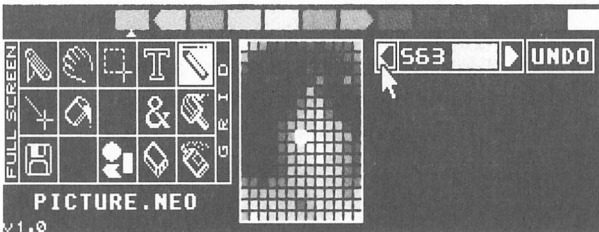
You can copy colours into the Mixer from both the Colour Palette and Colour Map. To copy from the Palette, double-click the left mouse button on the colour in the Palette. To copy from the Map, single-click the left mouse button on the colour in the Map.

## Cycling Colours

You can cycle (continuously scroll) the Colour Palette left or right between the two Delimiters, creating motion or blinking effects in your picture. You can also set the cycling direction and speed.

To cycle colours, follow these steps:

1. Select the Left Delimiter and drag it to the leftmost Palette colour you want to include in the cycle.
2. Do the same with the Right Delimiter, moving it to the rightmost Palette Colour you want to include.
3. Point to one of the two Scroll Arrows on either side of the Colour Mixer and click the right mouse button. The Arrow will appear in reverse video.



When you click the right mouse button, the Palette colours will begin a continuous scroll in the direction of the Scroll Arrow clicked on. As a colour leaves one end of the range enclosed by the Delimiters, it reappears at the opposite end. The colours on your Canvas will continuously adjust to reflect the scrolling action.

You can adjust the direction and speed of your cycling by working with the Scroll Arrows and mouse buttons. To do that, follow this chart:

Original Palette State	Mouse Button Clicked	Scroll Arrow Clicked On	
		◀ Left	▶ Right
No scroll	Left Right	Scrolls left once Starts left color cycling	Scrolls right once Starts right color cycling
Left color cycling	Left	Increases cycling speed	Stops cycling
	Right	Decreases cycling speed	Stops cycling
Right color cycling	Left	Stops cycling	Decreases cycling speed
	Right	Stops cycling	Increases cycling speed

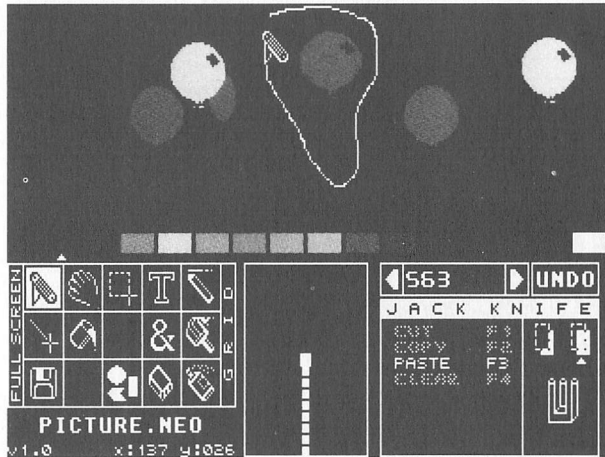
For example, you can begin colour cycling to the left by clicking the right mouse button on the left Scroll Arrow. Then, you can speed up the cycling by clicking the left mouse button on the same (left) Scroll Arrow. Next, you can decrease the cycling speed by clicking the right mouse button on the same (left) Scroll Arrow. Finally, you can stop cycling by clicking the left mouse button on the opposite (right) Scroll Arrow.

The information you set up for cycling your colours is saved along with your picture when you save the picture to disk. To activate colour cycling for a NEOshow slide program, be sure colour cycling is turned on when you save your picture. (See **Running the NEOchrome Slideshow** in this chapter for more information.)

# Working with Picture Sections

## Using the Jack Knife

The Jack Knife lets you cut out a freeform area of the screen and move it about.



To use the Jack Knife, follow these steps:

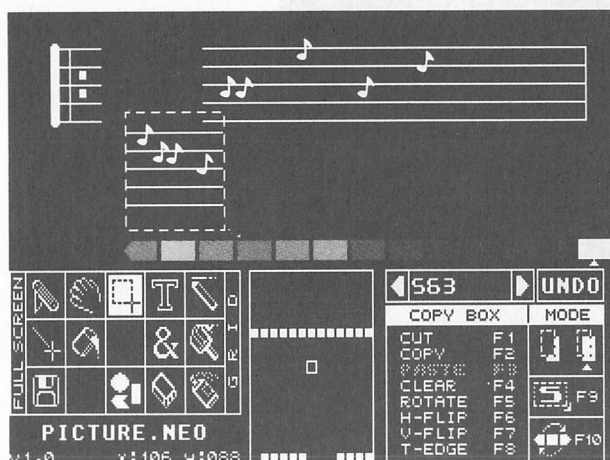
1. Select the Jack Knife from the Tool Grid.
2. With the Jack Knife on the Canvas, hold down the left mouse button and “cut” around a picture section. Then release the mouse button. A cycling “cut” line shrinks itself around all pixels in the section that are not background colour. This defines the section.
3. Move the tip of the Jack Knife into the defined area. When the tip of the Knife touches a pixel in the defined area, the Tool becomes an arrow.
4. When the arrow appears, press and hold down the left mouse button.
5. Move the arrow. The cut section will move along with it, out of its original position.
6. To move a copy of the cut portion (and not actually cut it out of the picture), click the right mouse button before moving the arrow. This stamps a copy of the section in its original spot.

You can stamp a copy of the cut portion onto the Canvas at any time by clicking the right mouse button (while continuing to hold down the left mouse button). You can paint with the cut portion by holding down the right mouse button as you move the arrow.

For instructions on using the Jack Knife options, see **Jack Knife** in **Chapter 2**.

## Using the Copy Box

The Copy Box defines a rectangular area of your picture for moving, copying, and manipulating.



To use the Copy Box, follow these steps:

1. Select the Copy Box from the Tool Grid.
2. Move the Tool onto the Canvas to the spot where you want to start your Copy Box.
3. Hold down the left mouse button and move the Tool in any direction. A rectangular Copy Box, anchored at the original spot, will follow the Tool.
4. Release the mouse button to set the Copy Box, defining a picture section.
5. Move the Tool into the Copy Box. When the Tool changes to an arrow, press and hold down the left mouse button.

6. Move the arrow. The defined section moves along with the arrow, out of its original position.

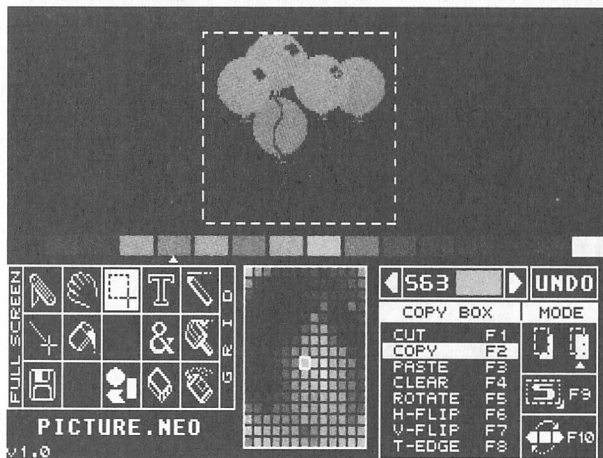
7. To move a copy of the cut portion (and not actually cut it out of the picture), click the right mouse button before moving the arrow. This stamps a copy of the section in its original spot.

You can stamp a copy of the defined section onto the Canvas at any time by clicking the right mouse button.

For instructions on using the Copy Box options, see **Copy Box** in **Chapter 2**.

## Pasting Picture Sections

After you cut picture sections out of a picture, you can paste them back in, or paste them into another picture. To do that, you use the Jack Knife, Copy Box, and Cut Buffer.



To cut out and paste picture sections, follow these steps:

1. Define a picture section with the Jack Knife or Copy Box.
2. Select CUT from the Options Box (or use the **[F1]** key) to cut the image from the picture into the Cut Buffer. Or select COPY from the Options Box (or use the **[F2]** key) to copy the image from the picture to the Buffer, leaving the original intact. The PASTE option in the Options Box now becomes active, indicating that something is in the Cut Buffer.

3. Now that the image is in the Cut Buffer, you can go on to other tasks, including loading a new picture, without disrupting the image.

4. To bring the picture back from the Cut Buffer, select the Jack Knife or Copy Box from the Tool Grid and click on PASTE in the Options Box (or press **[F3]**).

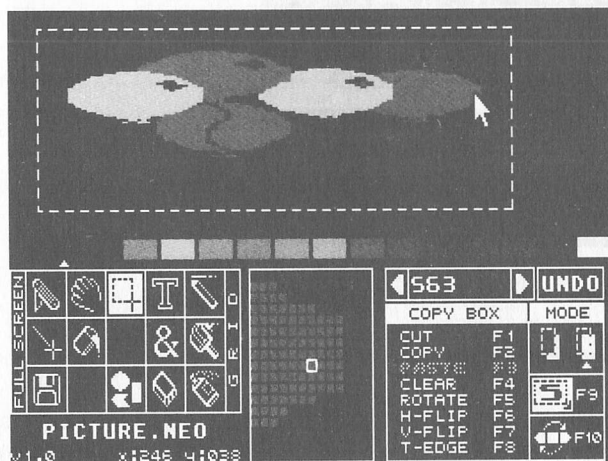
How the image is pasted from the Cut Buffer depends on the Tool selected. If you are using the Jack Knife, the image is pasted into the centre of the Canvas. You can drag this image anywhere you want on the Canvas before stamping it with the right mouse button or setting it with the left mouse button.

If you selected the Copy Box and did not define a Copy Box on the picture, the pasted image will create its own Copy Box in the centre of the Canvas. If a Copy Box is defined when you paste, the image appears inside this Box, scaled to fit. You can drag this image anywhere on the Canvas before stamping it with the right mouse button or setting it with the left mouse button.

You can use the pasting process to make a library of images. Cut out images from different pictures and paste them onto a separate, single Canvas reserved for that purpose.

## Expanding and Shrinking Picture Sections

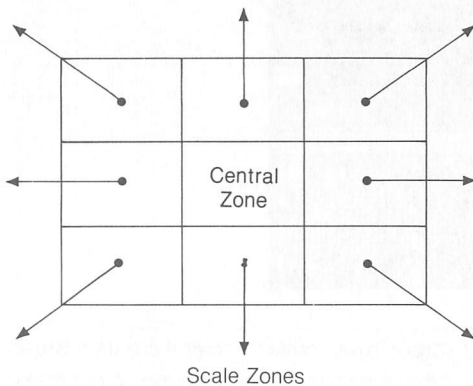
The Copy Box lets you scale a selected image to a new size.





To scale an image, follow these steps:

1. Select the Copy Box from the Tool Grid.
2. With the Tool on the Canvas, draw a Copy Box around the portion of the picture you want to resize. Or paste an image from the Cut Buffer by selecting PASTE (**[F3]**) in the Options Box.
3. Select Scale mode from the Options Box (or press **[F9]**). The option symbol will appear in reverse video.
4. Move the Tool into a Scale Zone of the Copy Box (the Tool changes to an arrow), hold down the left mouse button, and move the arrow in the desired direction of scaling. The diagram below divides a defined area into zones. By putting the arrow in any of the outside zones and dragging it in the direction shown, you can stretch and shrink the defined area. By putting the arrow in the central zone and dragging it, you can move the defined area.



The image is resized according to your mouse movements. To set the image, move the Tool outside the Copy Box and click the left mouse button. Then click on Scale again (or press **[F9]**) to turn off the mode.

## Special Effects

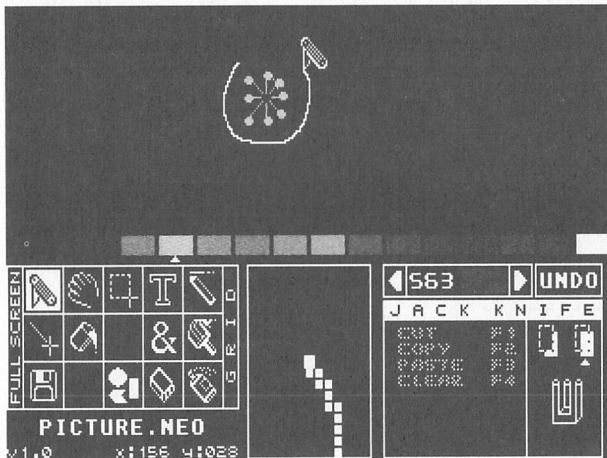
This section offers instructions for creating special effects in your NEOchrome pictures. As you work, you will want to change colours in your Palette, customizing it for each picture. To use your colours most effectively, try working with a few colours of similar shades. Then bring in highlights as your picture progresses.

Save your picture when it's at a point where you like it, or when you're ready to make a bold change. You can always retrieve this original later if you don't like how your picture turns out. The Palette you set up for your picture, including background colour, Delimiter positions, and colour cycling will be saved with your picture.

Keep in mind that in the half-size Canvas your actions will only affect the visible portion of your picture.

## Custom Brushes

You can create your own single- or multicoloured Paint Brushes by following these steps:



1. Use an out-of-the-way portion of your screen as a scratch pad and create a Brush pattern there in one or several colours. Patterns with more background colour make better Brushes.
2. Cut the Brush out using the Jack Knife.
3. Paint with the Brush. Move the Jack Knife into the defined area, hold down the left mouse button, and then drag the area, pressing and releasing the right mouse button.

Each time you press the right mouse button, the defined image will be painted on the Canvas. If you hold down the right button and move the mouse, you can spray the image on the Canvas. Store your Brush in the Cut Buffer or on another Canvas.

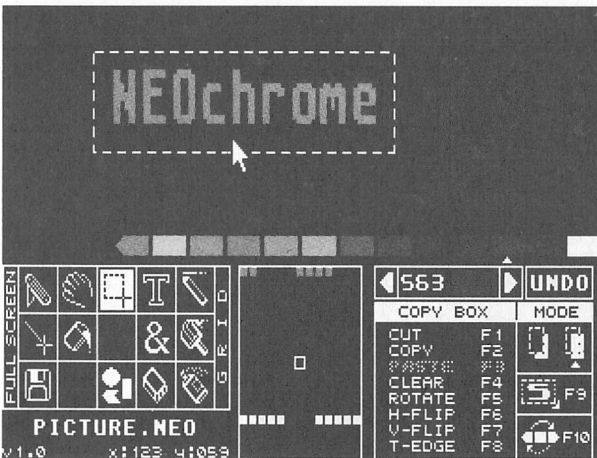
## Shading

You can create a dark to light shading effect by using successively smaller Spray Can Nozzles. Follow these steps:

1. Select the Spray Can from the Tool Grid.
2. For the dark region, select a dense nozzle from the Options Box and paint with slow, deliberate strokes.
3. For lighter regions, select less dense Nozzles and paint intermittently (clicking the left mouse button repeatedly) with quick strokes.
4. Heighten the shading effect by using progressively lighter colours and ramp colours.

## Shadowed Text

You can create text with a background shadow.



Begin with an empty Canvas. Then follow these steps:

1. Decide on two colours: one for shadow, one for text. For best results, the shadow colour should be darker than the text colour. Move the Delimiters onto the two colours.

2. Select Text from the Tool Grid and options from the Options Box. On the empty Canvas, write your text in the shadow colour.

3. Select Copy Box from the Tool Grid. In the Colour Palette move the Colour Selector to a colour you are not using.

4. Draw a Copy Box around your text, stamp it with the right mouse button, then move it to another section of your Canvas.

5. With the Copy Box still surrounding your text, select Colour Swap ([F10]) from the Options Box. The text colour will replace the shadow colour in the Copy Box.

6. Select Transparent mode from the Options Box.

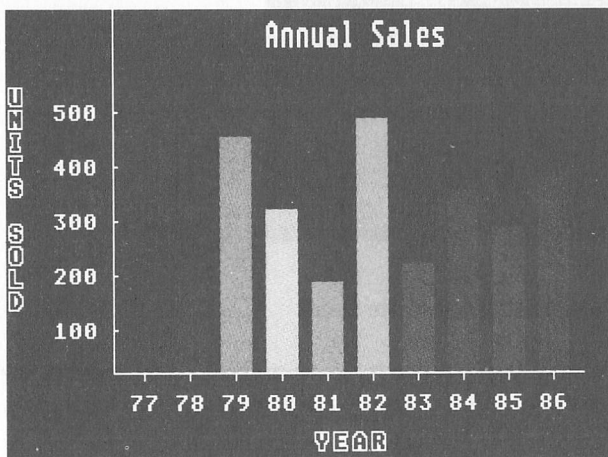
7. Drag the Copy Box over the original shadow-coloured text. Adjust the Copy Box until the text and its shadow are positioned the way you want them. Then release the mouse button.

8. Select COPY [F2] from the Options Box to save your text into the Cut Buffer.

You can now move your shadowed text into a NEOchrome picture by loading the picture, and selecting the Jack Knife or Copy Box and the PASTE ([F3]) option.

## Charts and Graphs

NEOchrome is well suited for making charts and graphs.



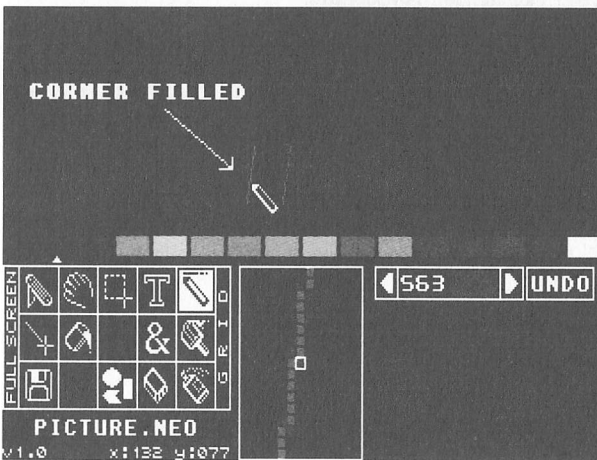
Follow these steps:

1. Select Grid from the Tool Grid to keep everything evenly spaced.
2. Use Line Draw or Shapes to draw the border of the graph, and label it with Text.
3. Various Tools will come in handy depending on the type of graph you're drawing. Use a filled rectangle for bar graphs and Line Draw for line graphs. Draw a picture, such as a car or dollar bill, and use the Jack Knife to make a pictograph.

Once you've created a series of designs, you can use the NEOshow program to present your work.

## Anti-Aliasing (No Jaggies)

Because the pixels on the Canvas are rectangular, lines drawn at peculiar angles have a staircase look.



An anti-aliasing algorithm can be applied to smooth the line, but calculating this by hand is time consuming. You can easily get a close approximation of an anti-aliased line by "corner filling." To do that, follow these steps:

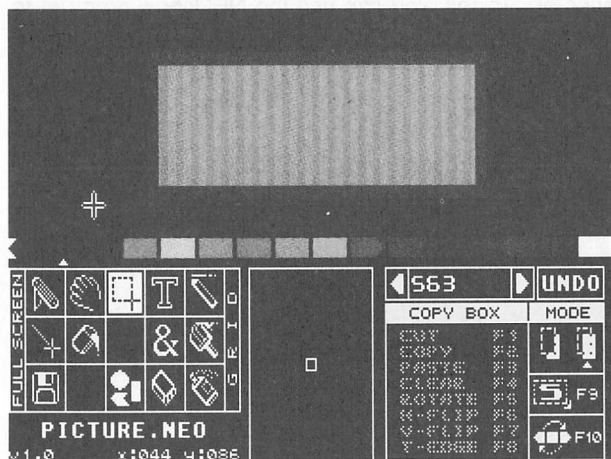
1. Select a colour halfway between the colour of the line and its surrounding background. For example, for a light blue line on a dark background, choose a dark or medium blue.

2. Use the Pencil and the Magnifier to fill in the corners of the line with the selected colour.

The corner-filled line appears much smoother and adds a professional touch to your graphics. It is especially useful on italicized text.

## Pattern Fills

You can fill a region with a coloured pattern.



Follow these steps:

1. Using the Magnifier, draw an 8 x 8 pattern (8 pixels wide by 8 pixels long), and cut it into the Cut Buffer with the Copy Box.
2. Paint the area you want to pattern fill with an unused Palette colour.
3. Move the Left Delimiter onto the background colour and the Right Delimiter onto the colour you just filled with. Make sure the Colour Selector is NOT under either Delimiter.
4. Select the Copy Box from the Tool Grid and click on Colour Swap ([F10]) in the Options Box. This swaps the background and fill colours.
5. Turn on Grid in the Tool Grid.

6. Select Jack Knife from the Tool Grid and Reverse mode from the Options Box.
7. Select PASTE (**[F3]**) from the Options Box. This pastes your pattern from the Cut Buffer to the Canvas.
8. Drag your pattern into the background area and paint with it by holding down the right mouse button while moving the mouse. The pattern will paint only over the background colour.
9. Restore your colours by choosing the Copy Box from the Tool Grid and Colour Swap (**[F10]**) from the Options Box.

## Creating Motion

You can create the effect of motion by using NEOchrome's colour-ramping and colour-cycling abilities. Try it with a ramp line by following these steps:

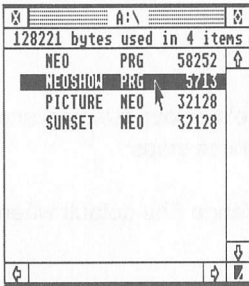
1. Move the Left and Right Delimiters to enclose a colour range (the default when NEOchrome loads is good for this).
2. Select Line Draw from the Tool Grid and draw ramp lines with the right mouse button.
3. Begin colour cycling by clicking the right mouse button on one of the Scroll Arrows.

As the Palette scrolls between the Delimiters, its colours scroll through the ramp lines, creating the illusion of motion. Sophisticated effects, such as waterfalls and rotating galaxies, can be created using this process with the Paint Brush and Spray Can.

## Running the NEOchrome Slideshow

The **Language Disk** contains a file called `NEOSHOW.PRG`. This is the NEOchrome slideshow program, NEOshow, which lets you view your paintings one by one in a continuous slideshow. NEOshow is run from the GEM Desktop as a separate program, although it displays the paintings you created and saved using NEOchrome.

To run NEOshow, exit NEOchrome and return to the GEM Desktop. Obtain a directory listing for the Language disk, then double-click on the file `NEOSHOW.PRG`.



A screenshot of a directory listing window. The window title is "A:\\" and it shows "128221 bytes used in 4 items". The listing contains the following entries:

File Name	Type	Size
NEO	PRG	58252
NEOSHOW	PRG	32128
PICTURE	NEO	32128
SUNSET	NEO	32128

The "NEOSHOW.PRG" entry is highlighted with a mouse cursor. The window has standard GEM-style window controls (minimize, maximize, close) in the top right corner.

The program will load, then cycle through and display all files with the `.NEO` extender on that disk.

**Note:** `NEOSHOW.PRG` must be on the disk or in the folder with the NEOchrome paintings you want to view, although `NEO.PRG` need not be on the disk. Each `.NEO` picture is 32128 bytes long. A single-sided disk (357376 bytes) can hold ten `.NEO` pictures, `NEOSHOW.PRG`, and `NEOSHOW.INP`. A double-sided disk (726016 bytes) can hold 22 `.NEO` pictures, `NEOSHOW.PRG`, and `NEOSHOW.INP`. (See **Creating a NEOSHOW.INP** File in this chapter for an explanation of `NEOSHOW.INP`.)



## Setting a Painting's Slideshow Timer

During the slideshow, each painting stays on screen for a certain length of time. You can tell NEOshow how long to display a painting by using the Scroll Arrows in NEOchrome.

Follow these steps:

1. Load the NEOchrome program, then load a painting onto the Canvas.
2. Turn on colour cycling by clicking the right mouse button on a Scroll Arrow (see **Cycling Colours** in this chapter). The painting's slideshow "timer" begins at this point.
3. When the desired length of time has passed, turn off colour cycling by clicking either mouse button on the opposite Scroll Arrow.
4. Turn on colour cycling again, and immediately save the painting to disk. Colour cycling must be on when you save the painting; otherwise, the picture's display time (just set) will not be saved to disk.

When you run NEOshow, the picture will stay on screen for the length of time set with the Scroll Arrows.

**Note:** Setting the slideshow timer with the Scroll Arrows also creates colour cycling in your painting. If you don't want to cycle colours, move the Left and Right Delimiters to the same colour in the Colour Palette before setting the timer.

## Creating a NEOSHOW.INP File

You can customize your NEOshow slideshow by creating a `NEOSHOW.INP` (input) file. This file lets you tell NEOshow which pictures to display and in what order, the length of time to display each picture, and what transition effects to use between pictures.

NEOshow input files are written on a text editor or word processor. To write a file, return to the GEM Desktop and load an editing or word processing program. Then create a file which you must name `NEOSHOW.INP`.

In the file, instructions for each picture must be entered on a separate lines. Each line has the following form:

```
[filename [.ext]], [#nnn] [, %n] <CR>
```

Each parameter enclosed in brackets is optional; it does not need to be included in the instruction line. Parameters can be separated by commas or blank spaces. Lines must end with a carriage return (**[CR]**), entered by pressing the **[Return]** key.

The first two parameters, filename and .ext, are your picture's filename and extender. A file must have an extender to be read by NEOshow. If a filename is specified without an extender, NEOshow assumes that the extender is .NEO. (You can rename a picture file from the GEM Desktop using a different extender, then specify that filename and extender in the input file.)

The third parameter, #nnn, is the length of time in tenths of seconds that you want the picture to be displayed. For example, to display a picture for 12 seconds, enter #120. You must include the hash sign (#); do not set the number to more than 10000 (1000 seconds). A duration set in the input file overrides a duration set using the Scroll Arrows in NEOchrome.

The fourth parameter, %n, is the transition mode from one painting to the next. For example, entering %5 results in the bottom-up transition, in which the next picture comes on screen from the bottom, scrolling upward. You must include the percent sign (%).

The eight transitions possible in NEOshow are:

- 0 Straight cut. The transition is directly to the next painting.
- 1 Fade out/in to white. The current picture fades out to white, and the next picture fades in from white.
- 2 Fade out/in to black. The current picture fades out to black, and the next picture fades in from black.
- 3 Fade out/in to background. The transition is from the current picture's background to the next picture's background.
- 4 Top down. The next picture comes on screen from the top, scrolling downward.
- 5 Bottom up. The next picture comes on screen from the bottom, scrolling upward.
- 6 Both top down. The current picture leaves the screen scrolling downward while the next picture comes on screen from the top, scrolling downward as well.
- 7 Both bottom up. The current picture leaves the screen scrolling upward while the next picture comes on screen from the bottom, scrolling upward as well.

A sample `NEOSHOW.INP` file might look like this:

```
PICTURE1.NEO,#20,%3  
PICTURE2.NEO,#100,%7  
PICTURE3.NEO,#1000,%0
```

When `NEOshow` is run with the sample `NEOSHOW.INP` file on disk, it will first display `PICTURE1.NEO` for two seconds, then make Transition 3 (fade out/in to background) to `PICTURE2.NEO`. It will display `PICTURE2.NEO` for ten seconds, then make Transition 7 (both bottom up) to `PICTURE3.NEO`. `PICTURE3.NEO` will be displayed for 100 seconds, then make Transition 0 (straight cut) back to `PICTURE1.NEO`. Any other `.NEO` pictures on the disk will not be displayed.

You can write instruction lines that include only one or two parameters. For example:

```
#100,%7
```

displays all pictures for 10 seconds using Transition 7.

```
#50
```

displays all pictures for 5 seconds using the cycle of transitions.

```
PICTURE1.NEO,%4  
#150
```

displays `PICTURE1.NEO`, uses Transition 4 to display the next picture found, displays that picture for 15 seconds, then uses the cycle of transitions to redisplay `PICTURE1.NEO`.

If you do not specify a filename, `NEOshow` displays the `.NEO` files in the order it finds them on disk.

If you do not specify a duration, `NEOshow` displays the file for as long as it takes to read it in from disk and make a transition to the next file. The default duration is dependent on the disk (floppy or hard) and the transition used. If you do not specify a duration in the input file, but have set the duration with the Scroll Arrows, `NEOshow` uses that duration.

If you do not specify a transition, `NEOshow` cycles through the transitions in order from Transition 1 to Transition 7, then repeats. Transition 0 is not used.

Before saving `NEOSHOW.INP` to disk, you must delete all formatting codes or instructions except the carriage return at the end of each instruction line. Then, save the file to the same disk containing `NEOSHOW.PRG` and the `.NEO` files you want to display.

The first picture in the slideshow always fades in from white. Press the **[Help]** key to display the current picture's filename during the slideshow. (One or two pictures will be displayed before the filename display begins.) Press **[Undo]** to deactivate the filename display.

Press **[Control][S]** to pause the slideshow. Press **[Control][Q]** to resume. Press **[Space Bar]** to exit NEOshow and return to the GEM Desktop.

## CHAPTER 4

# SYSTEM AND EXTENDED SYSTEM CHARACTER SETS

The ST Computer's Character Set offers 255 different characters and symbols. NEOchrome divides this set into the System and Extended System sets. You use these sets when adding text to your pictures.

When you select Text from the Tool Grid, the Options Box lists an option called FACE. By clicking the left mouse button in the box next to FACE, you can toggle between the System and Extended System sets.

## System Set

The System set contains the characters printed on the ST Computer keyboard plus some special characters. Select SYSTEM from the Text Options Box, then press the appropriate keys on the keyboard. The keystrokes that return the set's special characters are listed below.

Character	Keystrokes	Character	Keystrokes
␣	[Control] [A]	!	[Control] [Q]
␤	[Control] [B]	2	[Control] [R]
␥	[Control] [C]	3	[Control] [S]
␦	[Control] [D]	4	[Control] [T]
␧	[Control] [E]	5	[Control] [U]
␨	[Control] [F]	6	[Control] [V]
␩	[Control] [G]	7	[Control] [W]
␪	[Control] [H]	8	[Control] [X]
␫	[Control] [J]	9	[Control] [Y]
␬	[Control] [K]	a	[Control] [Z]
␭	[Control] [L]	␣	[Control] [␣]
␮	[Control] [N]	␤	[Control] [␤]
␯	[Control] [O]	␥	[Control] [␥]
␰	[Control] [P]	␦	[Control] [␦]
		␧	[Delete]

## Extended System Set

The Extended System set contains international characters and many special characters. Select EXTENDED SYS. in the Options Box, then press the keys that return the characters you want. Keystrokes that return the Extended System characters are listed below.

Character	Keystrokes	Character	Keystrokes
ü	[Control] [A]	ï	[Control] [K]
é	[Control] [B]	î	[Control] [L]
â	[Control] [C]	ñ	[Control] [N]
ä	[Control] [D]	ñ	[Control] [O]
à	[Control] [E]	É	[Control] [P]
ã	[Control] [F]	æ	[Control] [Q]
ç	[Control] [G]	œ	[Control] [R]
ë	[Control] [I]	ô	[Control] [S]
è	[Control] [J]	ö	[Control] [T]
û	[Control] [V]	ò	[Control] [U]
ù	[Control] [W]	†	[;]
ÿ	[Control] [X]	¶	[Shift] [,]
ö	[Control] [Y]	©	[=]
ÿ	[Control] [Z]	©	[Shift] [.]
¢	[Control] [0]	™	[Shift] [/]
£	[Control] [1]	ij	[Shift] [2]
¥	[Control] [2]	l	[Shift] [A]
β	[Control] [6]	x	[Shift] [B]
f	[Control] [-]	1	[Shift] [C]
á	[Space Bar]	2	[Shift] [D]
í	[Shift] [1]	7	[Shift] [E]
ó	[Shift] [']	ñ	[Shift] [F]
ú	[Shift] [3]	l	[Shift] [G]
ñ	[Shift] [4]	7	[Shift] [H]
ñ	[Shift] [5]	ñ	[Shift] [I]
æ	[Shift] [7]	ll	[Shift] [J]
g	[']	ˆ	[Shift] [K]
¿	[Shift] [9]	ˆ	[Shift] [L]
ˆ	[Shift] [0]	ˆ	[Shift] [M]
ˆ	[Shift] [8]	ˆ	[Shift] [N]
½	[Shift] [=]	j	[Shift] [O]
		o	[Shift] [P]

Character	Keystrokes	Character	Keystrokes
¼	[,]	½	[Shift] [Q]
ı	[-]	Ɔ	[Shift] [R]
«	[.]	Ɔ	[Shift] [S]
»	[/]	Ɔ	[Shift] [T]
ã	[0]	Ɔ	[Shift] [U]
õ	[1]	Ɔ	[Shift] [V]
ø	[2]	Ɔ	[Shift] [W]
ø	[3]	ı	[Shift] [X]
œ	[4]	Ɔ	[Shift] [Y]
œ	[5]	Ɔ	[Shift] [Z]
à	[6]	Ɔ	[ ]
ã	[7]	Ɔ	[ \ ]
õ	[8]	Ɔ	[ ]
..	[9]	^	[Shift] [6]
˙	[Shift] [;]	∞	[Shift] [-]
α	[']	±	[Q]
β	[A]	≥	[R]
Γ	[B]	≤	[S]
π	[C]	Ɔ	[T]
Σ	[D]	J	[U]
σ	[E]	÷	[V]
μ	[F]	≈	[W]
τ	[G]	°	[X]
Ö	[H]	•	[Y]
θ	[I]	•	[Z]
Ω	[J]	√	[Shift] [ ]
δ	[K]	n	[Shift] [ \ ]
φ	[L]	2	[Shift] [ ]
φ	[M]	3	[Shift] [']
€	[N]	-	[Delete]
Π	[O]		
≡	[P]		

**Note:** Characters available in System are not available in Extended System and vice versa. You must switch between the two sets to mix characters from both sets.



## CHAPTER 5

# .NEO FILE FORMAT

A NEOchrome picture file consists of a header followed by a screen memory image. The header is 128 bytes long and the memory image is 32000 bytes long. The table below illustrates the .NEO file format.

Offset	Variable Name	Length	Explanation
+0	File ID	1 word	NEOchrome expects 0
+2	Resolution	1 word	0=low, 1=med, 2=high. NEOchrome accepts only 0, but NEOshow can accept 0,1, and 2.
+4	Palette	16 words	Hardware Colour Palette for this picture.
+36	(Unused filename)	12 bytes	Unused by NEOchrome but may be used by other programs as a filename.
+48	CRAMPTR	1 word	Colour ramp pointers. Bit 15 set low if 8 bits are valid. If valid, high nibble of low byte is Left Delimiter, low nibble is Right Delimiter (0-15 each).

Offset	Variable Name	Length	Explanation
+50	CSCRLSPD	1 word	Colour Scroll Speed. Bit 15 set if valid. If valid, low order byte is signed number of frames per colour step, negative is left (information is read by slideshow only).
+52	Duration	1 word	Number of colour steps (defined by CSCRLSPD) to display this picture for before going to next (information is read by slideshow only).
+54	NEOX	1 word	Reserved. Value = 0. Horizontal offset into Canvas for leftmost display edge for NEOshow.
+56	NEOY	1 word	Reserved. Value = 0. Vertical offset into Canvas for topmost display edge for NEOshow.
+58	NEOWIDTH	1 word	Reserved. Value = 320. Total Canvas width in pixels.
+60	NEOHEIGHT	1 word	Reserved. Value = 200. Total Canvas height in pixels.
+62	Unused	33 words	Reserved by Atari Corporation for future expansion.
+128	Canvas Memory	32000 bytes	The size of the memory image is derived from NEOWIDTH, NEOHEIGHT, and Resolution. Future release may have Canvas sizes larger than one screen.

# GLOSSARY

**Alignment** A text entry option that lets you choose whether to enter your text centred, aligned left, or aligned right. The default position is left alignment.

**Aspect Ratio** The ratio of a pixel's width to its height. The aspect ratio of a low-resolution pixel on the ST Computer is 9:10.

**Background Colour** The colour in the far left block of the Colour Palette. The default background colour is black (RGB 000).

**Canvas** The area of the NEOchrome screen where you create artwork, using the tools from the Tool Grid and the colours from the Colour Palette. The Canvas can be displayed in either half or full screen.

**Clear** A Jack Knife or Copy Box option that lets you delete a picture area defined by either of those Tools.

**Colour Cycling** The effect of movement, such as a waterfall flowing, included in a NEOchrome picture. To create colour cycling, use the Delimiters in the Colour Palette and the Scroll Arrows on either side of the Colour Mixer.

**Colour Map** The gradated map showing the 512 colours available in the NEOchrome Palette. The Colour Map is displayed when the Canvas is in half screen and the cursor is positioned below the Canvas.

**Colour Mixer** The NEOchrome component that displays a selected colour and its RGB (red, green, blue) value.

**Colour Palette** A colour bar showing the 16 colours available for painting, from which the colour currently in use is selected.

**Colour Selector** The marker under the Colour Palette that points to the colour currently selected.

**Colour Substitution** Replacing pixels of one colour on the Canvas with pixels of another colour, using the Copy Box SWAP option and both Colour Palette Delimiters. One of the Delimiters must also be the selected colour.

**Colour Swap** Exchanging pixels of two different colours on the Canvas, using the Copy Box SWAP option and both Delimiters in the Colour Palette.

**Copy** A Jack Knife or Copy Box option that lets you define an area of the picture and leave it on the Canvas while also storing a copy of it in the Cut Buffer.

**Copy Box** The Artist's Tool that lets you define and manipulate a rectangular portion of your picture. Copy Box includes eight options and three modes.

**Constrain Mode** A mode that forces an action to follow a "straight edge," typically horizontally or vertically, and affects the Pencil, Eraser, Paint Brush, Spray Can, Jack Knife, Copy Box, Line Draw, and Shapes. The mode is effected by holding down either [Shift] key before using a Tool.

**Cursor** A moveable symbol on the screen that shows where your next action will take place. The cursor is manipulated with the mouse, and its form changes depending on its position on screen and the Artist's Tool selected.

**Cursor Colour** The colour in the far right block of the Colour Palette. The default cursor colour is white (RGB 777).

**Cut** A Jack Knife or Copy Box option that lets you remove an area of the picture, and store it in NEOchrome's internal memory area.

**Cut Buffer** The internal memory area of NEOchrome where a section of a picture is held after being cut or copied with the Jack Knife or Copy Box. The section can then be re-pasted into the original picture or into another picture. Only one image at a time can be held in the Cut Buffer.

**Delimiters** The left- and right-pointing blocks in the Colour Palette that specify the colour range of ramp lines and colour cycling and the two colours used in colour swapping and colour substitution.

**Disk** The Artist's Tool used to save the current picture to disk, load a previously created picture, or quit NEOchrome.

**Eraser** The Artist's Tool used to delete parts of your picture, in strokes or all at once, leaving only the background colour on the Canvas.

**Extended System Face** The second half of the ST Character Set, which includes foreign characters, accented letters, and symbols.

**Face** The Text option that lets you select System or Extended System characters.

**Full Screen** The NEOchrome feature used to display the full-screen Canvas. All Artist's Tools and options except the Grabber are functional in full screen.

**Grabber** The Artist's Tool used to move the Canvas up and down in half screen and identify particular colours on the Canvas.

**Grid** The NEOchrome feature that makes the Tool you're using "jump" to grid points as it is moved on the Canvas. Grid points are every eighth pixel on the Canvas and their coordinates are always evenly divisible by eight.

**H-Flip** A Copy Box option that lets you horizontally flip an area of the picture about its central vertical axis.

**Hot Spot** The outlined pixel in the centre of the Magnifier that marks the cursor's position on the Canvas.

**Jack Knife** The Artist's Tool used to define a free-form picture area in order to move or replace it. Jack Knife includes four options and three modes.

**Line Draw** The Artist's Tool used to draw a straight line from one selected pixel to another. Lines can be either solid-colour or ramp lines.

**Magnifier** The NEOchrome feature that displays an enlarged (5x) view of the Canvas showing individual pixels for detailed drawing.

**Miscellany** The Artist's Tool used to display the x/y coordinates of the cursor and save pictures in ASCII code. Miscellany includes a colour/palette UNDO option.

**Mix** To manipulate the RGB (red, green, and blue) values of a colour to modify it. See RGB Value.

**Opaque Mode** A mode available with the Jack Knife or Copy Box that lets you move the defined area over the picture on the Canvas. Anything behind the defined area is covered.

**Options Box** The box that appears in the lower right of the screen when you select the Paint Brush, Spray Can, Text, Line Draw, Shapes, Miscellany, Jack Knife, or Copy Box. It provides you with different ways to use the chosen Tool.

**Paint Brush** The Artist's Tool used to brush paint on the Canvas in a selected colour or multicoloured strokes. Paint Brush includes 38 brush options.

**Paint Bucket** The Artist's Tool used to "pour" paint on the canvas, filling a defined area.

**Paste** A Jack Knife or Copy Box option that lets you place the image that's in the Cut Buffer onto the centre of the Canvas, or onto a picture area previously defined with the Copy Box.

**Pencil** The Artist's Tool used to draw lines, point by point, on the Canvas.

**Pixel** Picture element. An individual screen dot; the smallest point of resolution on a display monitor.

**Ramp Lines** Multicoloured lines composed of equal segments of each colour in the range specified by the Colour Palette Delimiters. The colour of the line "ramps" from the Left Delimiter to the Right Delimiter.

**Reverse Mode** A mode available with the Jack Knife that lets you move the defined area "behind" other images on the Canvas. In other words, defined areas moved in this mode are visible only when placed on background colour.

**RGB Value** A number made up of three digits representing the red, green, and blue values, respectively, of the colour shown in the Mixer. Each of these values can range from 0 to 7, making possible 512 colours ( $8 \times 8 \times 8 = 512$ ).

**Reverse Video** A feature of the monitor display that produces the opposite combination of image and background colours from the usual combination; for instance, reverse video is black images on a white background, if white images on a black background is normal.

**Rotate** A Copy Box option that lets you rotate a picture area 90 degrees counter-clockwise.

**Scale** A Copy Box option that lets you stretch or shrink a picture area.

**Shapes** The Artist's Tool used to draw five shapes: rectangles, rounded rectangles, ellipses, freeform shapes, and polygons. Shapes can be drawn as outlines, solid shapes, or bordered images, with eight different border-width options.

**Spray Can** The Artist's Tool used to spray paint on the Canvas in a selected colour or multicoloured strokes. Spray Can includes 38 Nozzle options.

**Stamp** To copy a picture area defined by the Jack Knife or Copy Box to another place on the Canvas by clicking the right mouse button.

**System Face** The first half of the ST's ASCII Character Set, which includes standard alphanumeric characters and symbols.

**T-Edge** A Copy Box option that lets you trace the border of any image drawn in the currently selected Palette Colour, and then replace the colour within the border with the background colour.

**Text** The Artist's Tool used to insert letters, numbers, and symbols in a picture. Text options include five styles and six sizes. All 255 characters in the ST Character Set are available. See Text Size and Text Style.

**Text Size** The height in pixels of the largest character in the text you will type into your NEOchrome picture. The Text Options Box contains six text size options.

**Text Style** The look or form of the text you will type into your NEOchrome picture. The Text Options Box lists five text style options.

**Transparent Mode** A mode available with the Jack Knife or Copy Box that makes any background colour within the defined area transparent. When you move the defined area on the Canvas, any other painting behind it shows through the "holes."

**Undo** To eliminate the result of your most recent action. Also, a Miscellany option that restores colours in the Palette to their previous values.

**V-Flip** A Copy Box option that lets you vertically flip an area of the picture on its central horizontal axis.

**X/Y Coordinates** Numbers identifying the cursor's pixel position on the Canvas. Coordinates are numbered from the upper-left pixel of the Canvas, position 0,0. The x coordinate is the cursor's horizontal position, ranging from 0 to 319; the y coordinate is the cursor's vertical position, ranging from 0 to 199.

The first step in the process of developing a business plan is to conduct a market analysis. This involves identifying the target market, understanding the needs and preferences of the customers, and assessing the competitive landscape. A thorough market analysis provides valuable insights into the opportunities and challenges of the industry, which can inform the development of a realistic and profitable business plan.

Once the market analysis is complete, the next step is to define the business's mission, vision, and core values. These statements serve as the foundation for the business plan, guiding the company's strategic decisions and providing a clear sense of direction for the organization. A well-defined mission and vision statement can also help attract investors and build a strong brand identity.

The third step in the process is to develop a detailed financial plan. This involves projecting the company's revenue, expenses, and cash flow over a period of time, typically three to five years. A financial plan provides a clear picture of the company's financial health and helps identify potential areas of concern, such as cash flow shortages or high operating costs. It also serves as a key tool for securing financing from investors or lenders.

Finally, the business plan should include a marketing and sales strategy. This involves identifying the most effective ways to reach the target market, promote the company's products or services, and generate sales. A well-developed marketing and sales strategy is essential for the success of any business, as it ensures that the company is effectively communicating its value proposition to the right audience at the right time.

In conclusion, developing a business plan is a complex but essential task for any entrepreneur. By following a structured process that includes market analysis, defining the company's mission and vision, developing a financial plan, and creating a marketing and sales strategy, entrepreneurs can increase their chances of success and build a strong, sustainable business. A business plan is not just a document; it is a roadmap for the future, providing a clear path forward and a framework for decision-making.

As the business plan is developed, it is important to remain flexible and open to change. The market and the company's needs may evolve over time, and the business plan should be updated accordingly. Regularly reviewing and revising the business plan ensures that the company remains on track and responsive to changing market conditions. A business plan is a living document, and its effectiveness depends on the entrepreneur's ability to adapt and adjust as needed.

Overall, the process of developing a business plan is a critical component of the entrepreneurial journey. It provides a clear framework for the company's operations, financials, and marketing efforts, and helps entrepreneurs make informed decisions about the future of their business. By investing time and effort into developing a comprehensive business plan, entrepreneurs can increase their confidence and increase their chances of long-term success in a competitive market.

The business plan is a key document for any entrepreneur, and it should be developed with care and attention to detail. It is a tool that can help entrepreneurs understand their business, identify opportunities and challenges, and develop a clear strategy for success. A well-developed business plan is essential for the success of any business, and it is a document that every entrepreneur should have. By following the steps outlined in this document, entrepreneurs can develop a business plan that is realistic, profitable, and tailored to their specific business goals.





**ATARI®**  
**ST Tour**

Interactive Tutorial  
for all Atari ST and MEGA Computers

ATARI

ST JOUR

Interactive Fiction

1250 AVENUE OF THE COUNTRIES

# INTRODUCTION

Computing to many people is a mysterious subject and the thought of working with a microcomputer system can be quite frightening. The inclusion of **GEM** with the Atari ST goes a long way toward dissipating those fears by simplifying the routine jobs which are essential to maintaining a computer. Where tasks can be simplified to a mere "point and click" movement, this has been implemented and the **GEM Desktop** is a fine example of how quick and easy computing should be, and is, in the case of the Atari ST.

**ST Tour** is an interactive tutorial that introduces you to GEM and guides you around the world of mice, icons, windows and menus. Once mastered, the concepts and terms described during your guided tour of the ST will stand you in good stead not only with the GEM Desktop, but with most of the hundreds of applications and utilities available for the Atari ST

## Running ST Tour

Insert the **Discovery Pack Language Disk** and obtain a directory of drive A. Double-click on the `TOUR.PRG` icon.

Once ST Tour has loaded follow the on-screen instructions.

# INTRODUCTION

During the many months of preparation for this volume, the authors have been fortunate to have had the opportunity to work with a number of individuals who have provided us with their expertise and insights. We would like to thank them for their contributions. In particular, we would like to thank the following individuals for their assistance and support: [List of names]

The authors would also like to thank the following individuals for their assistance and support: [List of names]

Copyright © 2000

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or by any information storage and retrieval system, without the prior written permission of the publisher.

Printed in the United States of America



# Desk Accessories

for all Atari ST™ and MEGA™ Computers

ATARI

Desk Accessories

for all Atari 800 and MSX Computers

# DESK ACCESSORIES

## Introduction

A Desk Accessory is a special kind of program that is loaded automatically from disk into memory when the computer is switched on. Accessories can be called at any time from the GEM Desktop or other GEM applications by accessing the left hand menu (usually called Desk) and clicking on the name of the accessory you wish to use.

Desk Accessories are called **RAM resident programs**. they reside in memory at the same time that other programs are being used. The ST keeps the accessories 'on hold' until you require them for example, you could be working with a word processor and need to insert the result of a calculation. Simply call the calculator accessory from within the word processor, key in your calculation, type the result into your document and carry on.

A Desk Accessory is recognised by the `.ACC` extender at the end of its filename. If any `.ACC` files are on the disk in your drive when you switch on, they will be loaded into memory automatically. GEM can handle up to six Desk Accessories at once, always make sure that there are no more than six `.ACC` files on your startup disk.

The four accessories provided with your **Discovery Pack Language Disk** give you a wide selection of useful tools such as a clock, and calculator. Any of these files can be copied to other startup disks you may have and used on their own or with other accessories you obtain.

## Control Panel

One of the most useful desk accessories is the Control Panel (`CONTROL.ACC`). This accessory allows you to configure the ST to suit your own requirements or preferences. Refer to your **ST Owner's Manual** for a full explanation of how to use the Control Panel.

## VT52 Emulator

The VT52 Emulator (`EMULATOR.ACC`) is an industry-standard telecommunications terminal program. When a modem is connected to your ST computer, this program lets your computer communicate with other computers. Since the program is a bare terminal emulator only, it is most useful when used to hook into a large mainframe system.

To call the VT52 Emulator, point to and click on the `VT52 emulator` option from the `Desk Menu`. The VT52 Emulator immediately comes up on your screen.

To configure the modem port, press **[Help]** (see the instructions on the **Set RS232 Configuration** option in the **ST Owner's Guide**). To return to the GEM Desktop, press **[Undo]**.

## Calculator

This accessory (`CALC.ACC`) installs a handy calculator accessory ready to be used from within any GEM application. To call the calculator, point to and click on the `Calculator` option from the `Desk Menu`. The Calculator immediately comes up on your screen.

Numbers are entered either by clicking on the calculator's number keys or by pressing the equivalent keys on the numeric keypad. `CE` (Clear Error) is represented by the keypad's left bracket `[[]` and `CLR` (Clear) by the keypad's right bracket `[][]`.

Remove the calculator from the screen by clicking on the Close Box at its top left-hand corner.

## Clock

This accessory (`CLOCK.ACC`) displays a simple analogue clock (i.e. one with an hour and minute hand). The time shown is taken from the ST's internal system clock. This is set using the Control Panel Accessory described above.

To call the Clock, point to and click on the `Clock` option from the `Desk Menu`. A small window containing the clock appears on the screen. The size of the clock can be set by clicking and dragging the clock window's Size box (at the bottom right-hand corner of the clock window). Remove the calculator from the screen by clicking on the Close Box at its top left-hand corner. The time is continuously updated even when the clock is not displayed.



## **Acknowledgements**

Many thanks to **ST Club** for their assistance in providing a variety of Desk Accessories for inclusion in the **Atari Discovery Pack**.

ST Club are an excellent source of very low-cost, public domain software including a wide range of additional accessories and utilities. For further details contact:

**The ST Club  
49 Stoney Street  
Nottingham  
NG1 1LX  
0602-410241**

... of the ... in ...  
... of the ... in ...  
... of the ... in ...

The ...  
...  
...  
...  
...



# Game Instructions

Outrun, Space Harrier, Carrier Command, Bomb Jack  
for all Atari ST™ and MEGA™ Computers

ATARI

Game instructions

Other Space Invaders, Asteroids, Battle Tank, Bombs, Jack  
for all Atari 800 and MEGA computers

# OUTRUN

## Loading Instructions

Insert the Discovery Pack Game Disk containing **Outrun** into drive A and switch on the computer. Follow the on-screen instructions to load the game.

## Description

Outrun is the chart-topping car racing game in which you must hurtle across the open road at breath-taking speeds in your high performance sports car. Your only opponent is the clock that ticks away as you charge towards the check point that will give you extra time and a new target. Complete five stages and you'll earn the title king of the road.

Although five stages complete the course, there are 15 in total that you select by the direction you take at tree lined junctions. To find out the best route you'll have to play the game or look at the course route map displayed at the end of each game.

## Menus

Once the game has loaded, press and hold down the space bar until a menu appears at the top of the title screen. Moving the cursor over the appropriate title reveals the menu options. The four menus are:-

- OUTRUN:** Displays the game credits
- TUNE:** Tunes the music on and off
- GAME:** Start the game, continue a game, watch a demo, display the hi-score table and select between some and many cars
- CONTROL:** Choose between mouse and joystick control

## Game Controls

You can use either the mouse or a joystick to control and steer your car around the course. Moving the mouse or joystick left and right moves your car to the left or right on the road.

- FIRE:** Pressing the left-hand mouse button or the joystick fire button toggles between high and low gears
- ESC:** Pressing the **[Esc]** key Resets the game
- T:** Pressing the **[T]** key selects the tune being played
- M:** Pressing the **[M]** key selects mouse control
- J:** Pressing the **[J]** key selects joystick control
- SPACE:** Pressing the **[Space Bar]** selects the options menu and pauses the game

Collisions with other cars and obstacles will send you crashing off the track. You can continue but you will have lost valuable time.

# SPACE HARRIER

## Loading Instructions

Insert the **Discovery Pack Game Disk** containing **Space Harrier** into drive A and switch on the computer. Follow the on-screen instructions to load the game.

## Description

Our hero, a seasoned veteran of many space wars is on the scene again. This time to save the Dragon land which is occupied by barbaric and evil creatures and controlled by supernatural phenomena.

Space Harrier is an original action packed adventure that pits you in mortal combat with aliens of another planet.

## Gameplay

Space Harrier is probably the most alien game you will ever play as it pits you alone, against a planetful of dangers. Armed with a laser bolt gun and a jetpack that allows you to fly on this low gravity world you must fight alien monsters while avoiding meteor storms as well as natural hazards such as giant rock columns and trees.

The result is a game in which your reactions will have to be finely tuned and combined with an ice cool nerve, as any mistake will cost you one of your three lives.

Space Harrier is played in a number of stages each culminating in a duel with a flying dragon. To help you start your quest here is what you will face in the first stage.

- METEORS:** These are designed to catch unwary flying players. They can not be shot and must be avoided.
- WAVE ALIENS:** These attack in set patterns that can be learned to plan your best moment to attack them. Hint - They only fire at you when they fly towards you and try and catch you in a crossfire.
- TRIPLE ALIENS:** Three ball shaped aliens fly towards you and then open up to fire spinning bolts at you. Take them out before they get you.
- THE DRAGON:** A mighty monster that breathes fire as it charges at you. You will have to hit this monster several times before you destroy it. As you weaken it, it changes colour until it turns purple before exploding. The way is then open to the next stage and even greater challenges.

## Hints and Tips

- Use the continue option to learn the stages before going for those high scores.
- Learn what monsters do and which can be shot or must be avoided.

## Lives and Credits

Space Harrier gives you four credits when you begin a game which are used up every time you lose your third life. As long as you press fire within the nine second countdown the game will continue from your previous position. If the countdown reaches zero you will begin the game again from the first screen.

## Game Controls

Space Harrier can be controlled using either the mouse or a joystick. On loading the mouse is selected automatically. To select the joystick or revert back to the mouse press the following keys when the title screen is displayed.

- M:** Pressing the **[M]** key selects the mouse  
**J:** Pressing the **[J]** key selects the joystick  
**SPACE:** Pressing the **[Space Bar]** pauses the game, hitting any other key continues.

Mouse and joystick controls are identical and are as follows:

- Left:** Fly or run left  
**Right:** Fly or run right  
**Up:** Take off or fly upwards  
**Down:** Land or fly downwards  
**Fire:** Fire laser bolt gun



# CARRIER COMMAND

## Loading Instructions

Insert the **Discovery Pack Game Disk** containing **Carrier Command** into drive A and switch on the computer. Follow the on-screen instructions to load the game.

## Introduction

It is three years since the mining research ship SS Delta discovered a potential source of energy to rescue the failing industries of Earth. A seemingly random group of islands had surfaced in mid-ocean, the result of violent tectonic forces that were creating tidal surges and volcanic activity merely reported as "a minor geological fault".

This activity and the power that could be generated from it is far from minor. Two vast carriers have been developed to move from opposite ends of the islet group towards the centre, to load complex self-constructing centres on the group that could be populated in order to tap the tremendous volcanic power and mineral base of the area. By the time the giant Drizaw Industries Corporation had completed the massive ACC Epsilon, with its full complement of mining droids, aircraft, amphibious tanks, Command Centres and mining lasers, it had been decided to make the sister ship completely computer controlled. In the most daring and, unfortunately, most accomplished piece of computer sabotage ever, the ACC Omega, half of the saviour of the industrialised Earth, has been irreversibly programmed by the STANZA organisation to occupy and destroy all islands unless we pay a ransom of \$15 billion. Even then, where is our guarantee? We cannot disable the computer, it has been cryptically time-encoded, leaving us requiring more than the 7 hours remaining before the destruction of the Northern Side begins.

You have the ACC Epsilon at your disposal, with the added weapon of a bomb pod containing a computer virus which will logically recode the enemy Command Centres to the purpose they were intended. The ACC Omega is too strong to attack directly, Commander. Build a solid network of home bases by launching a Walrus with an Automatic Control Centre Builder - don't worry, this technology will soon be mastered. Your Base Island will support you logistically, but do not move out of range of a Stockpile Island. If you run out of fuel, or fail in your mission the future of Earth hangs in the balance. Good luck.

## Operations Guide

Having loaded either the Action or Strategy Game the screen which you will see contains rows of icons on left and right and a control panel at the bottom. The row of icons to the left remains constant, while the row to the right and the control panel change relevant to the functions of the mode you have chosen from the icons on the left.

### The Opening Screen

The row of icons to the left represent

- 1) The entire carrier
- 2) The armaments and defences of the carrier
- 3) The Water and Land Roving Utility Shuttle (Walrus) Amphibious Assault Vehicle (AAV)
- 4) The Manta (Multi-role Aircraft for Nautical Tactical Assault)
- 5) Game Status Information and Options.

These are the modes you will be operating in. If, for example, you have set a course for an island in Carrier mode and are cruising toward it, you have time to select the AAV icon to load the necessary weapons before the landing. When you receive the message "Carrier has reached destination" you may wish to select the Weaponry icon to launch a carrier-based attack on a hostile island before launching your Walrus.

### The Carrier Screen

**Icons on the right of the screen:**

#### Direct Control

The view you see upon loading the game is of the bow of your carrier and the island Vulcan, your base. You are in the **Direct Control** mode, shown by the blue icon top right of your screen. You may enter manual control by clicking the right mouse button (or Spacebar if you are using a joystick) and using the mouse move the carrier left or right, and by clicking the mouse button on the up chevrons next to the "Speed" icon you can increase speed and entirely move the ship by manual means. The **Hand** icon stops the ship, and the **Centre Up** icon will stop the ship rotating.

## Course Plotting

In selecting the icon below this, you can plot the course of the ship and it will arrive at the selected point by **Autopilot**. The two icons in the centre of the bottom control panel allow zoom in and out in ten stages. To set a course, position the pointer where you wish the ship to be, click the pointer and an orange cross will appear. Click the **PROG** icon and hold it down to select the **Autopilot**. The ship's course is now set. The course plotter works in the same way for the Manta and the Walrus.

## Damage Report

This icon, with a spanner across the icon of the ship, will reveal damage status.

## Stores

The stores icon - a packing case - will show the number of ACCBs, fuel pods, weapons and vehicles left. Clicking on any of these will bring information about use and quantity remaining onto the screen.

## Text Icon

The text icon below this will reveal all messages that have been displayed.

## The Control Panel

From left to right: **PAUSE** will pause the action until you resume by clicking on the icon again.

The radar screen shows enemy ships, missiles, coastlines and all tangible items. The zoom in/out may be used to focus the radar.

If the **Autopilot** is switched on, the "A" will show blue.

The small control panel to the right of this shows position in the standard X/Y axis configuration, and bearing is relative to North, with South as 180 degrees, West as 270 and East as 90. If any island is in range, it will be shown in the panel.

## The Armaments Screen

### Icons on the right of the screen

The initial screen when in **Carrier** mode is the laser sights - the **Laser** icon at the top right is highlighted - to fire the laser, click the right mouse button to go into actual command mode and lose the pointer, and use the left mouse button to fire. As with left/right controls for the ship, moving the mouse will alter the direction of the laser turret.

Decoy flares may be launched to outfox incoming missiles by selecting the second (**Flares**) icon on the right and clicking on the **LAUNCH** icon on the control panel.

A Remote Viewing Drone may be launched by clicking on the **Missile** icon and selecting **LAUNCH** as for the flares. Whilst you are receiving a view from the air, you may target a part of the island by clicking the right mouse button to give a cross-hair: to launch a surface-to-surface missile, click the left mouse button. You may only launch a missile when a drone is in the air.

Another defensive measure is the fleet of inflatable drones which may be positioned around the ship. Select the **Drone Manoeuvre** (fourth) icon on the right and then **Drone Exit** on the control panel to send a drone to its position. These positions may be altered by dragging them with the mouse.

The **Text** (last) icon shows status of all defence systems.

## The Walrus AAV

Four AAVs are available to you at any one time. When you select the **AAV** icon, a black screen will appear stating that Walrus 1 is in the dock. Before you launch an AAV, ensure it is armed, refueled and that it will not be more than 26km from the Carrier, or the signal will break up and the AAV will self-destruct. You must use this vehicle to make bases on free islands and attack or convert enemy islands.

### Icons on the right of the screen

Rather than going through the icons from top to bottom, it is most important for you to first equip your AAV. The third (**Weapon Loading**) icon, brings up a screen of equipment for the AAV. Use the “+” or “-” sign to review the weapons and pods, and add them to the grey cargo bays by dragging them across the screen with the mouse. You may initiate launch from this screen by selecting the **AAV out** icon which gives you a view of the port of the ship.

At this point it may be wise to set your destination, as with the carrier, by selecting the map movement icon and pointing a destination, remembering the AAV can travel on land. Launch the AAV by selecting the **AAV out** icon in the centre of the control panel and it will automatically move to its destination. Click on to **Direct Control Mode** to view the progress of the AAV. It will automatically circle when it has stopped. Select Centre Up to correct this.

To re-dock, reverse the procedure and select the **Re-dock** icon when the AAV is in range.

You may unload pods and use weapons in the AAV only when in **Direct Control** mode. As before, use the right mouse button to lose the pointer and the left to fire or unload.

To create an ACCB base on an island, drop a pod as outlined above. The base will build itself, but you must decide what kind of base you need.

**Defence** ACCB is for protecting the network

**Resource** ACCB to mine and refine raw materials

**Factory** ACCB to manufacture and build equipment

If an island is under enemy occupation, re-programming is necessary with a **Virus Bomb**. You can only carry one pod. Better bring a laser as well! After you have dumped the Virus Bomb pod, the enemy ACCB will be logically recoded and you will have gained a friendly island.

If an AAV runs out of fuel, another AAV must be loaded with a fuel pod, dispatched to the stricken AAV, parked nose-to-nose and the fuel pod jettisoned. Refuelling will occur automatically.

The AAV is also equipped with a nose-up radar which is operated in the same way as the carrier radar.

## The Manta

The loading and launching procedures for the Manta are similar to that for the AAV. When launching, the bottom left of the control panel shows icons for bringing the aircraft on deck, replacing it in the hangar and launching. Landing is an entirely manual operation, in which you must approach the landing deck from the back as "flat" and slowly as possible. Once you have made contact, the automatic landing procedure will take over and taxi the Manta to the refuelling bay.

As with the AAV, you may pre-programme a course but you may take the controls, whilst in **Direct Control** mode, at any time by clicking off the **A** for **Autopilot**, clicking the right mouse button and then using the mouse as a direct control device.

The armoury of payloads and weaponry includes missiles and lasers designed to attack enemy bases. As with the AAV, the operating range is limited by its remote guidance links to the carrier- more than 20km and the control telemetry becomes weak, more than 26km and the plane will self-destruct to avoid falling into enemy hands.

The radar and other icons on the control panel mirror those of the AAV and should be familiar to you by now.

## Game Status

### Icons on the right of the screen

By clicking on the **Disk** icon, your score, elapsed time and the status of friendly/enemy islands will be shown.

The **Flag** icon on the right gives you the option to surrender,

The **Disk Save** icon allows you to save the game to a formatted disk, and in case you have forgotten to format a disk you can do so whilst still in the game by selecting the **Disk Format icon** (third down).

The **Games Option** icon allows you to toggle on/off sound and visual effects.

## Playing Tips

You will grow accustomed to the procedures for attacking an enemy island even though they may seem somewhat complex and time consuming at first.

Try arming all your AAVs and Mantas whilst you are sailing between islands, and make a note of the contents of each, especially the AAVs.

Remember to build a strong network as you go, and if it helps make a map of the area with a note of the nature of the bases you have built this may prove easier than constantly using the map as a reference.

# BOMB JACK

## Loading Instructions

Insert the **Discovery Pack Game Disk** containing **Bomb Jack** into drive A and switch on the computer. Follow the on-screen instructions to load the game.

## Description

**Bomb Jack**, caped crusader extraordinaire, is all set to save the world by diffusing a series of bombs left carelessly scattered around. The action takes place on 5 separate backdrops including Disneyland, Miami and the Pyramids. There are 24 bombs on each screen to be diffused before Bomb Jack can move on to another job in another part of the world.

## Gameplay

Move Bomb Jack around the screen collecting each of the 24 bombs. If you feel up to it there are significant bonuses to be earned by collecting the bombs in the right order. The next bomb to collect is denoted by the burning fuse. You have to collect twenty two bombs or more in the right order to collect a bonus.

Press the fire button when Bomb Jack is on a platform and he jumps. Press and push the joystick forward for maximum height. Press it again to halt his ascent and press repeatedly to make Bomb Jack hover (this way you can collect the horizontal rows of bombs).

There are a number of additional bonuses in the form of 'pills' that can be collected. There is a bonus pill which increases the value of bombs collected, an extra life pill, a pill that turns all the nasties into coins that can then be collected for additional points.

The number of meanies increases throughout the early part of the screen and some of them are also blessed with a limited intelligence that adds greatly to playability. They do follow you around and are aware of where you are. However, they do not follow you everywhere and you can find safe places where they don't follow you. The main problem is that bombs don't come to you, so you have to go to the bombs - and once you show yourself, the intelligent meanies follow you again.

## Game Controls

Bomb Jack can be controlled using either the mouse or a joystick. On loading the mouse is selected automatically. To select the joystick or revert back to the mouse press the following keys when the title screen is displayed.

- M:** Pressing the [M] key selects the mouse
- J:** Pressing the [J] key selects the joystick
- Space:** Pressing the [Space Bar] pauses the game, hitting any other key continues.

Mouse and joystick controls are identical and are as follows:

- Left:** Fly or run left
- Right:** Fly or run right
- Fire:** Fly, Hover or Drop





# DISCOVERY PACK

3



## NEOCHROME, FirST BASIC, ST TOUR, UTILITIES, DESK ACCESSORIES, AND GAMES INSTRUCTIONS

**THIS MANUAL CONTAINS INSTRUCTIONS FOR THE LANGUAGE DISK  
AND THE GAMES DISKS, INCLUDING:**

- \* FirST BASIC : THE BASIC PROGRAMMING LANGUAGE
- \* NEOCHROME : THE PAINT PACKAGE
- \* ST TOUR : THE TUTORIAL PROGRAM
- \* THE DESKTOP UTILITIES
- \* GAMES INSTRUCTIONS:  
CARRIER COMMAND, OUTRUN, BOMB JACK, SPACE HARRIER

WRITTEN IN A CONCISE YET COMPREHENSIVE STYLE, THE MANUAL OFFERS EASY TO UNDERSTAND INSTRUCTIONS TO GUIDE YOU STEP BY STEP THROUGH THE HOST OF USEFUL PROGRAMS AND UTILITIES YOU GET WITH YOUR MACHINE.



ATARI

DISCOVERY PACK 3