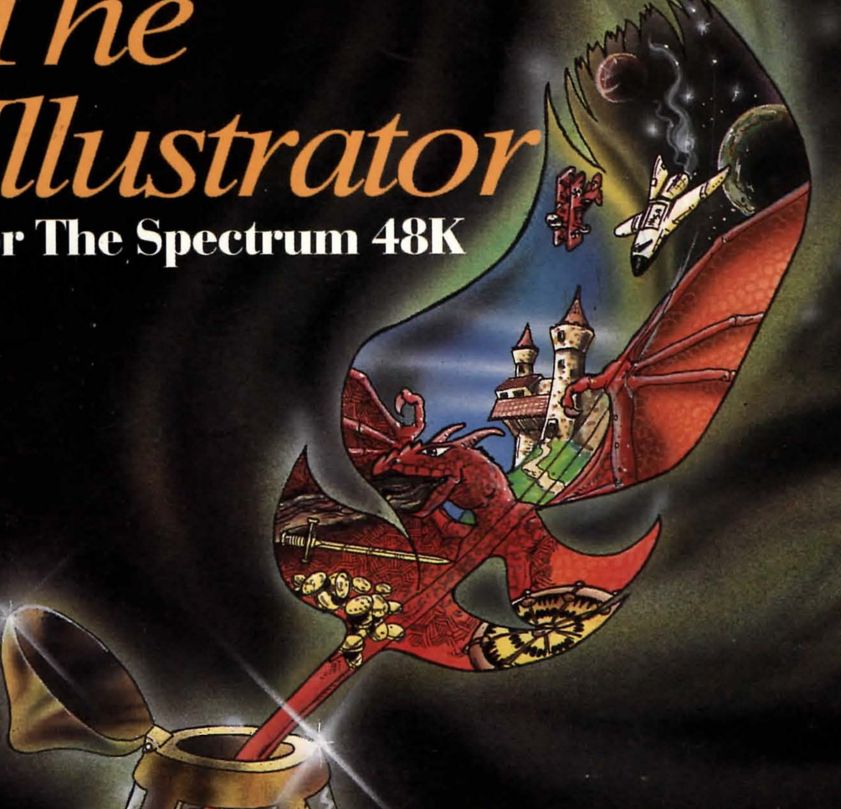
GILSOFT
Home Computer Software

The Illustrator

For The Spectrum 48K



For Use With

The Quill

Adventure Writing System

The Illustrator

A Graphics package for The Quill Adventure Writing System
on the 48K Spectrum by Tim Gilberts

Serial A

Acknowledgements

Many thanks to Graeme for advice and help in the use of Quill routines, Ziga Turk for his shade routine, Huw and Terry for their work on the graphic design and Pete for his corrections!

Contents

Getting Started

The Illustrator consists of four parts:-

a) A database which contains a string of drawing commands for each picture created.

b) A Menu system which allows tne database to be saved to and from tape plus a number of other options.

c) A Graphics Editor which allows commands to be inserted, deleted and tested within drawstrings.

d) An Interpreter which interfaces with the Quill and decodes the commands in a drawstring to produce a picture at the required location.

To load The Illustrator use LOAD "".

Part 1 of this manual will introduce you gently to The Illustrator, from creating a graphic database to creating a final graphic adventure. Part 2 contains a concise description of each section of The Illustrator for reference.

Part 1

The Main Menu

When The Illustrator has loaded you will be presented with the Editor's Main Menu. Some of the options, e.g., **Bytes Spare**, will perform a function and return to the Main Menu while others, e.g., **Graphics** will give you a sub-menu. **RETURN TO BASIC** is an exception to this as it executes the BASIC **NEW** command which destroys The Illustrator. **SAVE Adventure** does present you with a sub-menu but it does not allow you to return to the Main Menu; this is explained later so be careful!

The Input Routine

This is the same as that used in The Quill and you should already be familiar with its operation.

SAVE, VERIFY & LOAD Graphics

These three options on the Main Menu allow the graphic database to be saved to or reloaded from tape and in each case you will be prompted to "Type in name of file". When loading or verifying, the Spectrum will search for a file of bytes with the name specified and then load or verify it. If you enter a null filename (that is - just press ENTER) the Spectrum will load or verify the first file of bytes it finds on the tape. However, when saving, a null filename will produce the error message "Invalid file name". The BREAK key may be used to interrupt a SAVE, VERIFY or LOAD but if it is used to interrupt a LOAD, or a tape error is detected during a LOAD, then the graphic database will be cleared. Result: one blank location!

Creating a Graphic Adventure

The first step in creating a graphic adventure is to write the main adventure with The Quill. For an example we will use the adventure described in The Quill manual. In case you haven't kept a copy of the database we have provided one after The Illustrator on the enclosed cassette called "DEMO".

The Illustrator needs to know how many locations are contained in your adventure and also where the first free memory location is. This information is entered using **LOAD database** (Option G on the Main Menu). Use this option to load the Quill database called "DEMO" from the Illustrator cassette. **LOAD database** only needs to load the first few bytes of a Quill database no matter how big that database is, as these few bytes contain the necessary information.

When the Illustrator is initially loaded it has only one blank location contained within the graphic database. This is not enough for our sample adventure and so **LOAD database** will give you the option of Initialising (i.e. setting up) a suitable number of graphic locations. Reply "Y" to the prompt and The

Illustrator will create the required number of locations then return you to the Main Menu. (if you reply "N" then no change will be made to the graphic database - see the reference section for a detailed account of the **LOAD database** options).

Now your Illustrator should contain six blank (or 'null') locations. You can see if this is so by using **Graphics Start Table** (Option B on the Main Menu) and then using 'P' to PRINT the start options.

Due to the nature of The Illustrator it is very difficult to describe a picture in words so, in the greatest traditions of fast demonstrations, we have provided a file called "GRAPHICS" on the cassette - which is a part completed database!

Using **LOAD Graphics** (Option F) and the filename "GRAPHICS" load the database into The Illustrator.

Just to get a feel for the kind of pictures which can be created with The Illustrator select **Graphics** (Option A) and try 'P'rinting locations 1 to 5 in turn. Then use 'Z' to return to the Main Menu.

Now it's your turn to become an artist. You may have found that location 0 is blank. In The Quill manual this is meant to be the hall so let's create a picture of a hall:-

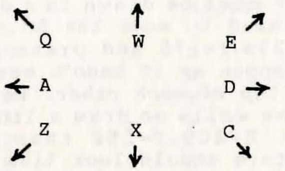
First of all, set up the background colours. This is done with the **Graphics Start Table** (option B on the Main Menu). Our hall requires a background PAPER of Yellow and an INK colour of Black so type: A 0 6 0 (remember the spaces) and press ENTER, then use 'P' to print the table. It should have an entry:

Location 0 PAPER:6 INK:0
Location 1 .... etc

If not, go back to the previous paragraph and try again. Otherwise use 'Z' to return to the Main Menu.

Now select Graphics again and amend location 0, i.e. A 0. You will be presented with a blank yellow screen with a flashing shape in each corner. This shape is actually two cursors in the very bottom left, but as they are so big they 'wraparound' and thus show up in each corner!

The cursors are the way the Illustrator Editor shows where you are on the screen. They can be moved using the keys around S as shown overleaf.



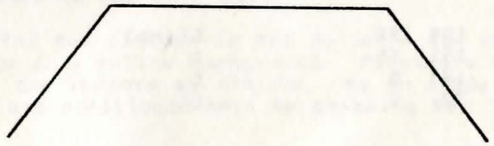
Try moving the cursor around a bit. You will notice that there is still a cursor in the lower left: this is called the Base Cursor (BC) and usually shows the last point plotted, also serving to mark a point for several of the drawing commands. The one you are moving is called the Rubber Cursor (RC) and is used to mark the other point you want a particular command to use. At the bottom of the screen you should see a rectangular blue block which displays a variety of information; this is called the Status Box. In the top left of the Status Box you will see the X and Y co-ordinates of the RC cursor displayed along with the number of the location we are currently editing. The bottom row displays the current values of INK,PAPER,FLASH,BRIGHT,INVERSE & OVER.

The movement of the RC cursor is quite slow as it moves a pixel at a time. This can be speeded up by pressing SHIFT at the same time as a direction. You can use this feature to get the cursor to the approximate area and then use the single pixel move to position it accurately.

If at any time you have pressed any other keys by accident you can delete their effect with SHIFT & 0 (i.e. the normal DELETE key).

Now we can start to draw. Position the cursor so X=55 and Y=71. The first thing we are going to draw is the outline of the walls. Press the L key for LINE draw and a line will be drawn from the Base Cursor to RC and the BC cursor will disappear. Now move to X=200,Y=71 (straight across) and you will notice BC is at the end of the line we just drew - quite a few operations cause the BC cursor to move to the RC cursor position. Now draw another line by pressing L again. If you make a mistake you can erase the last command using SHIFT & 0 (DELETE); but make sure you have shift pressed or you will enter an INK BLACK command. Now move RC to X=255,Y=0 and draw another line. The screen should now look like Diagram 1.

Diagram 1



The next part of the picture must be drawn in a different part of the screen so we need a command to move the BC cursor elsewhere. Position the cursor at X=255,Y=175 and press key M for MOVE! Probably very little will happen as it hasn't moved very far, but the two cursors will be on top of each other. We must draw three more lines for the top of the walls so draw a line to each of the following x,y positions; X=200,Y=152 then X=55,Y=152 and X=0,Y=175. Thus far the picture should look like diagram 2. Now the back wall needs defining so MOVE (i.e. M) to X=200,Y=152 (MOVE can be thought of as allowing the pen to be lifted from the paper - thus the BC cursor is moved to the RC position without drawing a line.) then draw a line to X=200,Y=71. Do the same for a line from 55,71 to 55,152.

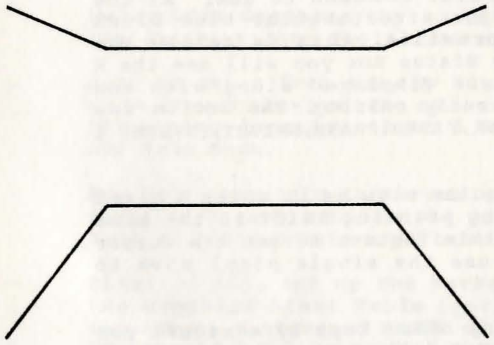


Diagram 2

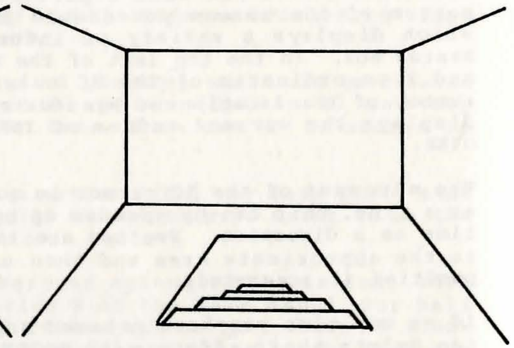


Diagram 3

The MOVE command causes the cursor to move to a specific X,Y position but this position is relative to the last position plotted so if you insert an extra command before the MOVE the position after the MOVE will be different. This can cause problems when editing, as the whole picture may change shape. It is thus a good idea to separate a picture into sections with the next command we are going to introduce. Position the cursor at X=72,Y=8 and press 'P' for PLOT. A set pixel will appear at the centre of the two cursors, which are now on top of each other. This point is absolute i.e. it is fixed in position no matter where on the screen the last point plotted was.

Next the steps descending into the cellar must be drawn. The following list of commands should be entered to do this:

Move Cursor To		Command
X	Y	
104	55	L(ine)
151	55	L
183	8	L
72	8	L
74	8	M(ove)
82	16	L
173	16	L
181	8	L
165	16	M
159	22	L
97	22	L
91	16	L
105	22	M
109	26	L
147	26	L
151	22	L
143	26	M
140	29	L
116	29	L
113	26	L

Now your picture should look like Diagram 3.

The picture is starting to take shape but it needs a bit of colour! We will use another absolute PLOT here to break up the picture. So position the cursor at X=57,Y=74. Now we do not want actually to set the pixel, so press 'I' and 'O' (Letters, not numerals) to set Inverse and Over, then 'P' to 'PLOT'; no actual dot appears due to the combined effect of plotting Inverse 1:Over 1. We do not want Inverse & Over any more, so press 'I' & 'O' again to cancel them.

In order to colour the back wall Red we will 'PAPER' it! So select PAPER RED by pressing SYMBOL SHIFT and 2, (INK and PAPER are both selected using keys 0 to 8 but PAPER is selected by pressing SYMBOL SHIFT as well). Now position the cursor at X=195,Y=148 and press 'H'. A block of Red should appear on the back wall. Key 'H' is the BLOCK command which allows the area between the RC and BC cursors to be filled with the currently selected colours - in this case Red PAPER.

The next command to introduce is a very powerful feature of The Illustrator which allows a standard picture to be drawn and then used in other pictures. The defined picture is called a 'Subroutine' and the command to call it is GO SUB.

To demonstrate this we have defined a subroutine ready for you; position the cursor at 72,96 and press M (i.e. MOVE the cursor to give a starting point for our 'Subroutine'). Now press 'G' for GOSUB and you will be asked for a location number. The example is location 6 (notice how subroutines are non existent location numbers). Next you will be asked for a scale value: for the moment use 0 (we will see examples of scale later). As if by magic a picture appears on the back wall.

Now the background for the picture is Red so let's use the BLOCK command again to give it a Yellow background. First use the MOVE command to position the cursors at 109,105. As an aside you can see why the cursors are positioned here by pressing key 'Y' which will give you a grid overlaid on the screen, and the area behind the silhouette man can be clearly seen. Cancel the grid by pressing 'Y' again.

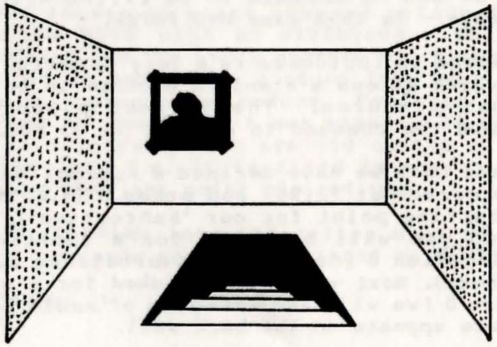
Next select Yellow PAPER (SYMBOL SHIFT & 6) and also Transparent INK (key '8') which prevents the BLOCK command from affecting the silhouette's colour. Then position RC at 84,131 and press 'H'.

Now a bit of shading: first of all PLOT the pixel at 0,0 to give us another absolute position in the picture. Then position the RC cursor at 128,32 and press 'F' - the area by the stairs will be FILLed in. This FILL command is very powerful and will find every nook, cranny and hole so ensure areas are fully surrounded with set pixels before using it (don't forget you can DELETE it if a mistake is made!).

Next we are going to give a bit of texture to the side walls, so select Ink Red (key '2'). Now position the cursor at 248,40 and press 'T'. This is the SHADE command which requests a shade value from 0 to 255, and controls the pattern which will be used to fill the area which RC is in. For our example we will use value 95. Experimentation with values is the best way of determining useful shading patterns but the reference section gives details of how to calculate your own. To make the walls match position RC at 48,80 and use a SHADE of 95 again.

Your picture should now look like diagram 4. It might be a good idea to save your graphics thus far if you haven't already done so. You can return to the Graphics submenu by pressing ENTER.

Diagram 4



The above section has introduced a lot of new ideas and commands the main points of which are presented below;

- Load database must be used to set up the correct number of locations before designing the graphics for a particular adventure.
- Graphics Start Table is used to set up the background colours of a location or to define one so that it cannot be drawn when the final adventure is running. i.e. a subroutine
- There are two cursors that are used to define the two points between or upon which a particular command will work.
- The Rubber Cursor is moved using the eight keys arranged around 'S' and accelerated by pressing CAPS SHIFT as well.
- The Status Box at the bottom of the screen displays the current position X,Y of RC and also the current values of the colours, Inverse and Over.
- Erroneous commands can be deleted using CAPS SHIFT and 0.
- A new position for the Base Cursor can be set using MOVE (M) (that is RELATIVE to the last position of BC) or PLOT (P) (which is a fixed (or 'absolute') positioning of the cursor)
- PLOT (P) commands should be used to separate a picture into fixed sections: you will find out how useful this is later. Note that a PLOT with both Inverse and Over at 1 leaves the pixel unaffected, only moving the BC cursor.
- The BLOCK (H) command can be used to fill the rectangle defined by BC and RC with the currently selected colours at high speed.
- INK colour is changed by pressing one of the keys from 0 to 8 and PAPER in a similar way but pressing SYMBOL SHIFT as well.
- The GOSUB (G) command can be used to draw a standard picture called a 'Subroutine' within another picture.
- FILL (F) can be used to fill completely a self-contained area with set pixels.
- SHADE (H) can be used to shade a self-contained area with one of 255 shading patterns.

All the commands introduced above plus the others provided by The ILLustrator Graphics Editor are defined fully in the reference section and summarised on the back page of this manual.

Adding a doorway

In order to demonstrate some of the other editing commands we will add a doorway to location 0 on the left wall. Make sure you are Amending location 0.

All the Illustrator commands are stored in the database in a long string in memory. The editor can split this string at ANY point along its length allowing quite advanced editing to be carried out. Press key 'S' for START and the screen clears but don't worry - you haven't lost the drawing. The START command moves what is known as the drawstring pointer to the start of the sequence of drawing commands (the drawstring) for the location we are editing. We can now step along the sequence until we reach the command we want; in this case it is the PLOT command which was used to start the steps in the floor.

You can step along using 'N' (NEXT). If you do this twelve times you will see the picture build up until the PLOT at 72,8. Now we want to insert our door drawing commands BEFORE this PLOT so that the PLOT will ensure the rest of the picture remains in its fixed position. We can step back through the picture using key '9' (PREVIOUS). (although if you have to go a fair way back it is faster to do a START (S) - then NEXT (N) until you reach the required point). In this case press '9' just once as we only need to step back past the PLOT.

MOVE (i.e. use 'M') the cursors to X=15,Y=19 - this is the starting point - and then draw lines to the following points;

15,131	39,128	39,50	31,50	31,129

this will give a display as in Diagram 5. You could use NEXT to step through all the way to the end of the picture but a fast trick is to press ENTER to return to the Graphics Sub Menu and then Amend the location again.

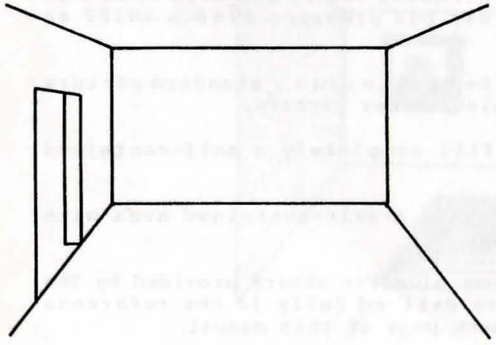


Diagram 5

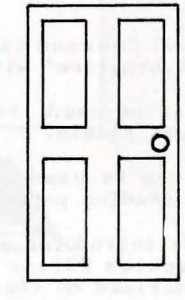


Diagram 6

Using Scale in subroutines

We have already looked at the use of subroutines in pictures; now we are going to see how to set up our own subroutines. For an example, we are going to use a door which will be added to the back wall of the hall. The type of door is shown in diagram 6.

First of all we must create a location containing the door; use Insert (I) on the Graphics sub-menu to create a new location. This is achieved by just typing I and ENTER, which will create a new location and automatically amend it. So now we have a blank screen on which to draw our door. Make a note of the Location number that has been created as you will need it later. As it is a subroutine we are going to draw it fairly large and Scale it when it is used. The following sequence of commands will create the door:-

They are in the form "X,Y;command" running across and then down.

0,144;L	80,144;L	80,0;L	0,0;L	8,8;M	8,64;L
32,64;L	32,8;L	8,8;L	72,8;M	48,8;L	48,64;L
72,64;L	72,8;L	72,80;M	72,136;L	48,136;L	48,80;L
72,80;L	32,80;M	8,80;L	8,136;L	32,136;L	32,80;L

Diagram 6 shows a handle on the door: this is left as an exercise for you! If you don't want to tackle the handle yet then continue without it and amend the subroutine later.

Now we must add the door to location 0. Press ENTER to return to the Graphics sub-menu and Amend location 0. A suitable point to insert our door is just before the PLOT which starts the steps in the floor so press S for START and step through the commands using N (NEXT) until you find it. Position the pointer just before it using 9 (PREVIOUS). MOVE the cursors to X=143,Y=71 as the starting point for our door and press G for GOSUB; the location number is the one you made a note of earlier.

Now as for Scale,the door we drew was far too large to use so we will make it smaller by using a scale value of 4. The scale value can range from 0 to 7; 0 means no scale and 1 to 7 specify the size of the subroutine in eighths. e.g. 4 means 4/8 = 1/2 , thus the door is drawn half size.

You could try deleting that GOSUB using CAPS & 0 and then using a different scale. If you use too big a scale you will get an 'Integer out of range' error. Don't worry - when you press a key it will redraw the picture to just before the GOSUB, thus using 'N' (NEXT) will cause the error again. To DELETE this erroneous command press CAPS SHIFT & N but be be careful you only press it once or you will end up deleting more than just The GOSUB.

Make sure you have a door of scale 4 drawn. Use NEXT to step through the picture until the PLOT at 0,0 (soon after the subroutine call for the picture on the wall). Use '9' (PREVIOUS) to move back past the PLOT then select PAPER CYAN (i.e. SYMBOL SHIFT & 5). MOVE the cursors to 147,75 then position the RC cursor at 180,140 and press 'H' for BLOCK. Finally select PAPER 8. Now use the trick of ENTER and Amend 0 again to see the final picture

Freehand drawing

There is one more function which will be introduced in the tutorial - the FREEHAND option. This allows you to add fine detail to pictures, but be warned it eats memory! FREEHAND is selected by pressing 'R' and an "F" will appear in the Status box to indicate FREEHAND mode. Every time you move the cursor the direction it was moved in is stored in the drawstring and the point it left PLOTted. Freehand mode can be cancelled using 'R' again. n.b. if there is a continuous run of more than three pixels in any part of your drawing use a LINE command to draw that part as it will take less memory than FREEHAND.

Important: FREEHAND is not subject to Scale so it may not be used in a subroutine which is drawn at anything other than full size (scale 0).

Summary

Again the above has introduced a variety of principles which are summarised below;

* All commands for drawing are stored in a long string which can be split at any point. The current position of the split is called the drawstring pointer.

* START (S) moves the drawstring pointer to just before the very first drawing command.

* NEXT (N) can be used to step through the commands one at a time, while PREVIOUS (9) can be used to step backwards.

* It is faster to START and use NEXT to reach a particular point than to use PREVIOUS several times.

* Additional sections to a picture should be added just before one of the PLOT statements which have been used to split the picture into sections. This ensures that the remainder of the picture is not upset by the additional part.

* If you wish to get to the last drawing command quickly press ENTER and amend the location again.

* The Scale input on GOSUB allows subroutine to be drawn in one of eight sizes; 0 means no scale (full size), 1 to 7 specifies a size in eighths.

The Final Stage

Now that we have created the graphic database suitable for our adventure we have to save it as a game. Ensure you have saved the graphics before proceeding to save an adventure.

SAVE adventure (Option H on the Main Menu) allows the graphics created with The Illustrator, the database created with The Quill and a machine code program to be saved to tape in such a way that using LOAD "" from BASIC will cause an autorunning graphic adventure to load into the Spectrum.

Because of the way The Illustrator works (i.e. it does not hold the entire Quill database in memory) **SAVE adventure** has to load in the Quill database for your adventure before it can save it to tape. This means The Illustrator will be overwritten and thus unavailable. So selecting **SAVE adventure** checks you actually want to continue - go no further unless you mean to!

You will be requested to enter a file name. This must be the name of the database (C series) which you saved from The Quill. Note it will also be the name that the final adventure is saved under. The Illustrator will then be locked in a loop until it successfully loads the required database (no going forwards or backwards). Once this is done you will be provided with a mini menu which will allow you to SAVE & VERIFY as many copies of the final adventure as you like. When you have done so, you can select RUN adventure which commits you to playing the adventure: there is no return!

Creating your own game

The following steps should be adhered to fairly closely in order to produce your own graphic adventure;

1/ Design the adventure using The Quill but try to give it a wide variety of different locations suitable for graphics. One other point you might like to include is to modify the normal REDESCRIBE entry in the Event table to read;

```
R  _    Conds
         Acts    SET  29
                 DESC
```

The picture for each location is normally drawn only when you first visit a location. Flag 29 is scanned by The Illustrator interpreter and if SET (that is it contains 255) the picture is drawn anyway and the flag cleared. This allows the picture to be recalled whenever REDESCRIBE is used by the player. Note that flag 29 is therefore unavailable for use in graphic adventures.

2/ Choose the locations which are to have graphics and draw out your ideas (appendix B may help here).

3/ Save a database from The Quill with the name you want to use for your final adventure.

4/ If you are an A series Quill user convert the Quill database to a C series one as described in Appendix A.

5/ Create a suitable blank graphic database using **LOAD database** on The Illustrator Main Menu.

6/ Draw out your pictures defining any locations which do not have graphics to be subroutines.

7/ Use **SAVE adventure** (once you have saved the graphics) to combine The Quill database and produce an independent game.

The above points should help you produce a professional adventure but if you intend trying to sell your adventure ensure it is thoroughly tested. We would repeat the request that you credit the use of The Quill and The Illustrator in your game and continue our offer of looking at at your games with a view to marketing them. If you are interested, please send a fully tested copy of the database, the final game, a map, a solution and one pound to cover administration costs to:-

GILSOFT
30 Hawthorn Road
Barry
South Glamorgan

Please be prepared to wait a while as we thoroughly check all the games sent to us and this can take some time. Happy Adventuring!

Part 2

Description of the Interpreter

The interpreter consists of a simple loop used to decode each command in the required drawstring one after another and also to take care of subroutine calls.

Description of the Graphics Database

The Graphics database contains two tables and an area of miscellaneous information that is number of graphics created and pointers to the rest of the database. Unlike a Quill database, the graphics database grows DOWN from the bottom of the interpreter. This allows the area normally occupied by The Quill to be used in the final adventure. The tables are:-

A The Location Flags

This table has a 1 byte entry for each picture, Bit 7 specifies if the picture can be drawn when that location is reached (i.e. if the location is not a 'subroutine') and Bits 0 to 5 describe the start PAPER & INK for the picture. Bit 6 is used by the interpreter.

B The Picture Table

Each entry in the table uses 3 bytes plus the length of the Drawstring. There are always at least the same number of entries as locations in the adventure, but the table can contain extra entries which are available for use via the GOSUB command. The Drawstring is encoded as a variety of various length commands which minimise the amount of memory needed to produce the drawing.

Detailed Description of the Editor

The Illustrator maintains a similar menu driver to The Quill allowing users familiar with that program to use The Illustrator easily.

A Graphics

Pictures may be inserted, amended, printed or have their length calculated:-

Insert I

The next available picture number is used and a null entry is made for it in the picture table. An entry of 'subroutine (PAPER 1, INK 7)' is also made for it in the location flag table. Processing then continues with an automatic call to the amend routine to allow the user to amend the null entry already set up in the picture table.

Amend **A picno.**

The graphic database is expanded to provide a gap at the end of the required picture. The main loop of the Graphic Editor described below is then entered. When return is pressed any gap still remaining is removed. n.b. unlike editing text on The Quill, the database itself is changed, thus you cannot abandon an edit with CAPS & 6.

Size S

The number of bytes between the start of the drawstring and the start of the next is calculated and printed on the screen.

Print **P picno.** or **L picno.**

The required picture is drawn on the screen and if **L** was selected the Spectrums COPY command is called. **picno.** must be specified.

Points to note:

a) Locations which are subroutines have PAPER 1, INK 7 as start up colours.

b) There is a limit of 254 pictures.

B Graphics Start Table

The status of a picture can be amended or printed:-

Amend **A picno. (paper ink)**

A flag is set to indicate that picture **picno.** is a subroutine unless PAPER and INK values are specified, in which case they are stored as the initial Global colours for the picture. <u>All locations in the adventure which do not require a picture should be Amended as a subroutine.</u>

Print **P** or **L**

Printing is either to the screen using **P** or to the printer using **L**. If the location is not a subroutine the Global colours are printed.

C Bytes Spare

The number of bytes between the end of the Illustrator or the end of the Quill database (whichever is higher) and the bottom of the graphic database are printed.

D SAVE Graphics.

Equivalent to the BASIC command SAVE f CODE m,n where m is the start address of the database, n is the length and f is the filename

E VERIFY Graphics.

Equivalent to the BASIC command VERIFY f CODE m,n

F LOAD Graphics.

Equivalent to the BASIC command LOAD f CODE, i.e. it will load any file of bytes back to the address it was saved from.

<u>**Very Important**</u>

If BREAK is pressed or a tape error detected during a load then the database held in memory would be corrupt, so a call is made to set a minimum database. This means your graphics will be lost, but unlike The Quill you haven't corrupted the database and can use any option available. If CAPS SHIFT & 6 is pressed while a name is requested the graphic database will be unaffected.

G LOAD database

This allows the information from a Quill database to be transferred to the graphic database. First the tape is searched for the required file and the first few bytes loaded in, (BREAK or a tape error will restore the existing Quill database). There are then three possibilities;

1) There is room to incorporate the new database and the number of bytes spare is amended if required.

2) The new database requires more pictures than are contained in the graphic database.

3) The new database would overwrite the graphic database.

In cases 2 & 3 the option of initialising a suitable graphic database is provided, if this is not done the original database is restored.

This option can be used to set up a graphic database suitable for a new adventure, or to amend the graphic database if you have made changes to the Quill database.

H SAVE Adventure

Important; because of the way The Illustrator overlays The Quill database once you commit yourself to loading a Quill database you will be locked in a loop until it is successfully loaded. Ensure you have saved your graphics as a file using option D before using SAVE adventure. Once the database has successfully loaded you will be presented with a mini menu allowing the final adventure to be Saved, Verified or Run. Note the files are saved with the name used on The Quill database and are designed to auto run when loaded from BASIC using LOAD "".

(c) Return to BASIC

Jumps to the BASIC NEW command.

Detailed description of the Graphic Editor

This section of the Illustrator allows a variety of operations to be carried out on a drawstring. When editing, the string is laid out in memory as follows;

END	The end of string marker
NEXT	Any commands still undrawn
SPARE	Available memory
END	Temporary end marker
DRAW	The main draw string

A two cursor system is used for editing; the Base cursor shows the last point plotted, moved to etc, the Rubber cursor shows the next position of the Base cursor or the point for a fill.

The Editor provides four groups of commands;

1) Drawing Commands

PLOT P Sets the pixel at the position of the Rubber Cursor (RC) according to Inverse and Over, then moves the Base Cursor (BC) to that position.

MOVE M Moves BC to RC without affecting the screen. This is coded as a relative offset from BC.

LINE L Draws a straight line from BC to RC according to Inverse and Over, then moves BC to RC. The line is coded as a relative offset from BC.

FILL F This causes the area which contains RC to be filled with set pixels (n.b. this will fill every nook and cranny and leak through ANY holes). No notice is taken of Inverse or Over and the position of RC is unaffected. The position of the fill is coded as a relative offset from BC.

All the above use 3 bytes in the database.

SHADE T This command, written by Ziga Turk allows areas to be filled with one of 255 shading patterns. The pattern is requested and the value to enter can be calculated by taking a 4*2 pixel grid and adding together the numbers shown in each square which are to be set;

128	64	32	16
8	4	2	1

Because of the way the routine works you cannot start a shade on X=0 or Y=0 and pressing T will be ignored. Note if SHIFT & T is pressed the shade will affect the boundary area (useful for shading clouds etc).

Shade uses 4 bytes in the database.

BLOCK H Causes a block of the currently selected colours to appear between the BC and RC cursors.

Block uses 5 bytes in the database.

FREEHAND R A toggle action (i.e. pressing 'R' will turn on Freehand if it was off and vice versa) causes all the movements of the cursor to be stored in the database. These are repeated when the string is drawn and each pixel plotted according to Inverse and Over.

Each cursor move uses 1 byte!

2) Colour Commands

INK 0>8 The current ink is set to the value of the key pressed. INK 8 as in BASIC causes all ink to be taken from the existing screen.

PAPER 0>8 & SYMBOL SHIFT sets the current paper to the value of the key pressed. PAPER 8 as in BASIC.

FLASH V The new value of Flash is requested (0,1 or 8).

BRIGHT B The new value of Bright is requested (0,1 or 8).

all the above use one byte in the database.

INVERSE I The state of Inverse (on/off) is toggled.

OVER O The state of Over (on/off) is toggled.

Neither Inverse nor Over use any memory but their state is encoded as part of each future instruction which is affected by them.

3) Subroutine Command

GOSUB G A picture number is requested which must be in the range 0 to max. picno. A scale value for the picture is then requested. This can be from 0 to 7 where the number indicates the size of the picture in eighths - 0 means 'no scale' (i.e.8/8).

Please Note;

a) Scale only affects certain commands, these are MOVE,LINE,FILL and SHADE. PLOT, BLOCK and FREEHAND commands will not be scaled or relocated and should generally not be used in subroutines (although they will work and can be used usefully sometimes)

b) You may only nest subroutine calls to a level of ten. (nesting means calling a subroutine from within a subroutine).

c) Scale does not affect GOSUB commands, i.e. if a GOSUBis used within a subroutine the string drawn will be at a fixed size and not scaled.

d) Calling the same routine you are drawing will cause an 'Out of Memory' error as the limit of 10 subroutine levels will have been reached.

4/ Editing commands

START S Puts the Drawstring pointer at the start of the drawstring.

NEXT N Executes next available drawstring command: if there isn't one the command is ignored.

PREVIOUS 9 Moves the drawstring pointer back one command and updates the screen.

DELETE SHIFT & 0 deletes the previous command in the drawstring and updates the screen.

DELETE NEXT SHIFT & N deletes the next command if there is one.

GRID Y Has a toggle action for a character grid of INK 0, PAPER 7 using BRIGHT. This allows exact positions of colour boundaries to be taken into account while drawing.

Editor Error Messages and their meanings

BREAK BREAK was pressed during a peripheral operation.

STOP in INPUT CAPS SHIFT & 6 pressed.

Tape loading error as BASIC. Note that a tape error during a load Graphics will set up a null database.

Database full There is not enough room in the database for what you were attempting.

Limit reached The maximum number of pictures is already present.

Integer out of range While drawing a picture a LINE command has gone out of range. This is usually due to a change of position of the starting point while editing.

Out of memory Ten nested subroutines have been used. This will usually be due to a recursive call to a subroutine, e.g. using GOSUB (G) on the picture you are editing will cause this error.

Note 1 After an error during editing, the Drawstring pointer is positioned just before the command which caused the error (i.e. a NEXT (N) command will cause the error again). If printing then a return is made to a menu.

Note 2 During input the Spectrum will emit a RASP if the screen and/or input buffer is full.

Appendix A: A-C Conversion

The Illustrator is designed to add graphics to a C series database, i.e. a database saved from a C series Quill. The serial type of the Quill is printed on its opening screen e.g. A03,C02. If you have an A series Quill you will have to convert any A series databases before using them with The Illustrator.

After the other programs on the accompanying cassette you will find a program to convert an A series database to a C series one. The program is loaded by typing:

```
LOAD "A-CCONVERT"
```

please note: the program is in two parts: the first part automatically loads the second.

If you have less than 990 bytes spare in your database, the conversion will fail with an error report. In which case you will have to reduce the database in size with an A series Quill.

Appendix B : How to Approach a Location Picture

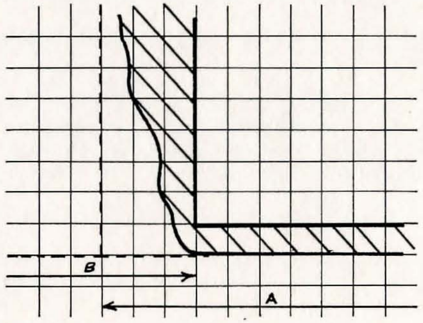
Just to give you a different viewpoint on The Illustrator we asked our graphic artist (who did the demonstration pictures) to give his advice on using it;

The most important thing to remember when approaching a location picture, is to think visually - a pencil and sketch pad are essential equipment for creating a graphic adventure. This not only saves time when coding the picture but will also form a useful source of visual information for future pictures.

Once an initial idea for a picture has been decided on, it may be transferred to one of the many commercially-available screen planners or to plain graph paper. This must be done with a ruler as accurate line-positioning is essential. When transferring the image to the graph paper it is useful to consider where each LINE, PLOT or MOVE command is going to be in the string of commands i.e. it is sloppy programming to draw a line at the top of the screen, then one at the bottom with a MOVE command in between if a line is to be drawn between these lines later in the command string. This is uneconomical on memory as it uses an unnecessary MOVE command.

Once the picture is drawn in line form, and an economical drawing sequence worked out, it is time to decide on the colours for the picture. Divide the Graph paper into 32 by 22 large squares, each containing 64 small squares (8 * 8) which form the Spectrum's ATTRibute grid. The Illustrator implements all of the Spectrum's colour capabilities.

It is worth mentioning the combined use of INK and PAPER to join two large areas of colour and fill them at speed at a point other than a colour boundary. The diagram shows the area we are considering: Area A uses a BLOCK command first with INK and PAPER both of the required colour, then area B uses BLOCK with PAPER of the second colour and INK 8 (Transparent), Finally a FILL with PAPER 8 of the shaded area finishes the effect. This is much faster than using FILL over the entire area of A.



When coding the picture it is best to draw as many of the outlines as possible first (split at suitable points by PLOT statements to fix positions), then any subroutines then any colour, FILL and SHADE commands (although colour commands sometimes need to be included in the line drawing section).

At first the SHADE command may appear to offer a bewildering range of options but after a bit of experimentation you will find a variety of patterns which appeal to you. These should be noted to make shading of future pictures easier (don't forget that shading clouds etc is made simple by pressing CAPS & T which causes shade to affect its defining border).

Any object which appears more than once in the picture, or more than once in the adventure, should be created as a subroutine, (e.g. Pictures, Doors, Bricks, Trees etc). Subroutines are constructed in the same way as a normal picture but be careful in coding them; if you incorporate a feature make sure it is required each time the subroutine is used i.e. a shadow may not be the same in every picture. Ensure that the subroutine is constructed so as to allow it to be easily positioned in the final picture, and make it large enough to draw correctly and give a range of sizes when used with scale.

The above-outlined methods will allow you to produce economical (on memory) and well-organised pictures, However it's up to you to produce original and interesting graphic adventures.

Huw Jones.

Summary of the Graphic Editor commands

NAME	KEY	DESCRIPTION
Start	S	Moves drawstring pointer to start of drawstring.
Previous	9	Moves drawstring pointer back one command.
Next	N	Executes next command in string. With CAPS SHIFT it deletes the next command.
Delete	0	& CAPS SHIFT: the previous command is deleted and the display updated.
Cursor move		The cursor is moved using the keys around S, thus: Q W E A D Z X C by one pixel at a time, if CAPS SHIFT is pressed movement is by 8 pixels.
INK	0-8	The current INK is set to the required value.
PAPER	0-8	& SYMBOL SHIFT will set the current PAPER.
FLASH	V	The state of FLASH can be set as 0,1 or 8.
BRIGHT	B	The state of BRIGHT can be set as 0,1 or 8.
INVERSE	I	Toggles the state of Inverse.
OVER	O	Toggles the state of Over.
PLOT	P	Plots the current position of RC, then moves BC to it.
LINE	L	Joins BC to RC with a straight line.
MOVE	M	Moves BC (relatively) to RC position.
SHADE	H	The defined area is shaded with the specified pattern.
FILL	F	The defined area is filled with set pixels.
BLOCK	B	The rectangle defined by RC,BC is set to the currently selected attributes.
GOSUB	G	Allows a standard shape defined in another string to be drawn at the cursor position.
FREEHAND	R	Allows cursor movements to be stored.
GRID	Y	Overlays the attribute grid on the screen.

GILSOFT
Home Computer Software

GILSOFT
☎: (0446) 732765