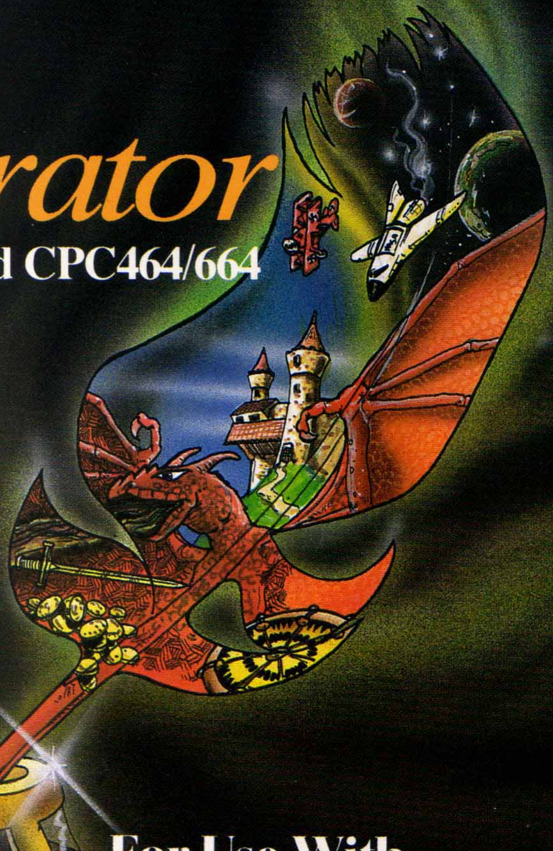
GILSOFT
Home Computer Software

# The Illustrator

**For The Amstrad CPC464/664**



**For Use With**

## The Quill
### Adventure Writing System

# The Illustrator

A Graphics package for The Quill Adventure Writing System
by Tim Gilberts, Amstrad CPC 464/664 version by Graeme Yeandle

Serial A

## Acknowledgements

Many thanks to 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 the database to be saved or loaded 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 reset the computer using CTRL, SHIFT & ESC and then enter RUN "I (RUN" if loading from tape).

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 similar to that used in The Quill and you should already be familiar with its operation.

### Disc / Tape

For those of you with a disc drive option H on the Main Menu allows you to choose between using disc and tape.

### SAVE & LOAD Graphics

These options on the Main Menu allow the graphic database to be saved or reloaded and in each case you will be prompted to "Type in name of file". When loading, the computer will search for a file of bytes with the name specified and then load it. A null filename may be used when you are using tape but not with disc. Care should be taken when using Load Graphics with a null filename because it will load any binary file. The ESC key may be used to interrupt a SAVE or LOAD but if it is used to interrupt a LOAD, or an error is detected during a LOAD, then the graphic database will be cleared. Result: one blank location!

### CAT

Option I on the Main Menu can be used to catalogue a tape or disc in the same way as from BASIC.

### 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 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 F on the Main Menu). Use this option to load the Quill database called "DEMO". **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" (after "DEMO" on cassette) - which is a part completed graphics database!

Using **LOAD Graphics** (Option E) and the filename "GRAPHICS" load the graphics 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:-

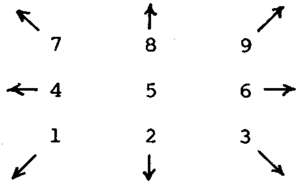
First of all, set up the colours to be used. This is done with the **Graphics Start Table** (option B on the Main Menu) which allows you to specify the colours to be used for each picture i.e. Inks 0-3 & Border. The colours we will use for the Hall are Yellow, Black, Red & Blue so type A 0 24 0 6 2 24 (remember the spaces) and press ENTER, then use 'P' to print the table. It should have an entry;

```
Locno. 0 I0=24 I1=0 I2=6 I3=2 B=24
Locno. 1 I0=...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 the bottom left hand corner. This shape is actually 2 cursors which are exactly in the corner of the screen. They do not look like cursors at the moment because the majority of them is off the visible screen.

The cursors are the way the Illustrator Editor shows where you are on the screen. They can be moved using the keys around 5 as shown below.



Try moving the cursor around a bit (start with the 9 key to move the cursor onto the visible screen) 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 the Status Box which contains the X & Y coordinates of the RC cursor, the number of the location we are currently editing and the current Pen. The BC cursor is an X while the RC cursor is a + .

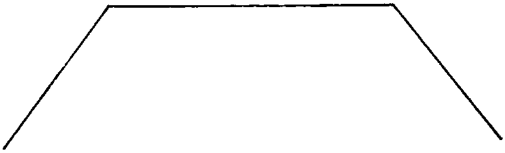
The movement of the RC cursor is quite slow as it moves a pixel at a time. This can be speeded up by pressing SHIFT at the same time as a direction. You can use this feature to get the cursor to the approximate area and then use the single pixel move to position it accurately.

Note that the cursors can only be positioned on even numbered pixels. The visible screen is from x=0 to 638 and from Y=0 to 366. The 5 Key can be used to centre the RC even if it is offscreen.

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

Now we can start to draw. Position the RC cursor so X=138 and Y=150. 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 join the RC. Now move to X=500, Y=150 (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 DEL. Now move RC to X=638, 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=638, Y=366 and press key M for MOVE! 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=500, Y=320 then X=138, Y=320 and X=0,Y=366. Thus far the picture should look like diagram 2. Now the back wall needs defining so MOVE (i.e. M) to X=500, Y=320 (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=500, Y=150. Do the same for a line from 138,150 to 138,320.

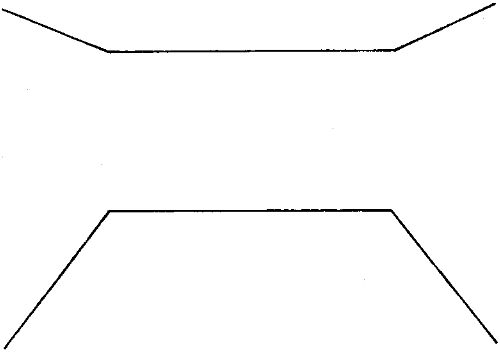


Diagram 2

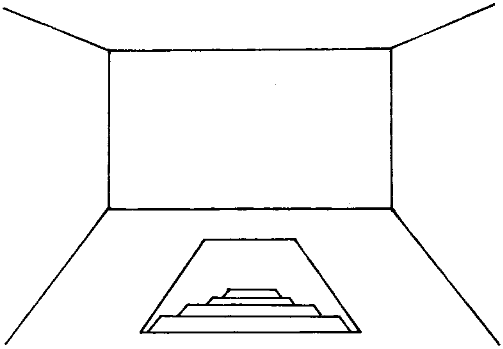


Diagram 3

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=180, Y=20 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 X | Cursor To Y | Command |
|---|---|---|
| 260 | 116 | L(ine) |
| 378 | 116 | L |
| 458 | 20 | L |
| 180 | 20 | L |
| 186 | 20 | M(ove) |
| 204 | 34 | L |
| 434 | 34 | L |
| 452 | 20 | L |
| 412 | 34 | M |
| 398 | 46 | L |
| 242 | 46 | L |
| 228 | 34 | L |
| 262 | 46 | M |
| 272 | 54 | L |
| 368 | 54 | L |
| 378 | 46 | L |
| 358 | 54 | M |
| 350 | 60 | L |
| 290 | 60 | L |
| 282 | 54 | L |

Now your picture should look like Diagram 3.

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. First of all we need an absolute position at which to start the subroutine so position the RC at X=180, Y=202. Now we do not want to plot this point so press 'A' for MOVEA (Move Absolute). 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.

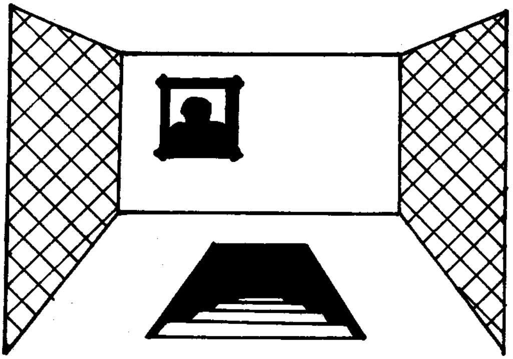
The picture is starting to take shape but it needs a bit of colour! We will use another absolute move here to break up the picture. So position the cursor at X=138, Y=150 and press 'A' for MOVEA. The cellar is rather dark so we will fill in the area around the stairs in Black. To do this position RC at X=298, Y=110 and press 'F' for FILL. The picture of the man would be better in Blue so press CTRL & 3 to select Pen 3, position the RC cursor at X=218, Y=230 and press F again.

In order to colour the back wall Red we need Pen 2 so press CTRL & 2, position the RC cursor at X=298, Y=310, and press F. The fill routine first fills in a downward direction & then fills in an upward direction. To make it as fast as possible, when it is going up it doesn't look down and vice versa. thus the position you start the fill from may affect how much of the area is filled. In this case the area under the picture frame has to be filled separately. To do this position the RC cursor at X=202, Y=206 and press 'F' again.

Next we are going to give a bit of texture to the side walls, so position the cursor at 58,310 and press 'S'. This is the SHADE command which needs to know which pattern(s) 0-15 to use to fill the area. For our example we want to use pattern No. 3 only, so answer 3 for the first pattern and 3 for the second pattern. (The reference section gives details of the 16 default patterns) To make the walls match position RC at 518,150 and shade with pattern No. 3 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 sub-menu 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 declare the 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 keys around '5' and accelerated by pressing SHIFT as well.

* The Status Box at the bottom of the screen displays the current position X,Y of RC and also the current Pen.

* Erroneous commands can be deleted using DEL.

* 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) or MOVEA (A) (which is a fixed (or 'absolute') positioning of the cursor)

* PLOT (P) or MOVEA (A) commands should be used to separate a picture into fixed sections: you will find out how useful this is later.

* The current Pen is changed by pressing CTRL & one of the keys from 0 to 3.

* 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 a self-contained area.

* SHADE (S) can be used to shade a self-contained area with one of numerous 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 '↑' 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 '→' (NEXT). If you do this twelve times you will see the picture build up until the PLOT at 180,20. 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 '←' (PREVIOUS). (although if you have to go a fair way back it is faster to do a START (↑) - then NEXT (→) until you reach the required point). In this case press '←' just once as we only need to step back past the PLOT.

MOVE (i.e. use 'M') the cursors to X=38,Y=42 - this is the starting point - and then draw lines to the following points;

38,276    98,268    98,106    78,106    78,270

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.

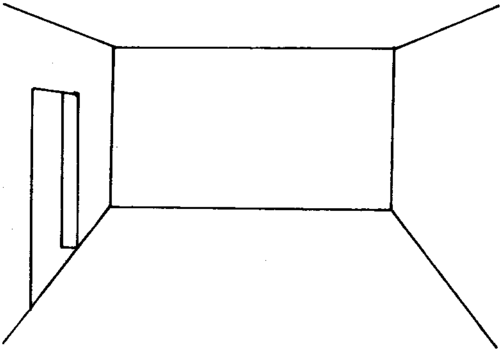


Diagram 5

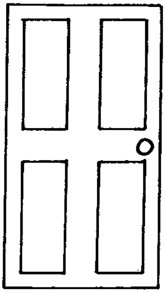


Diagram 6

### 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. Notice the S in the Status Box signifying subroutine. 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.

| | | | | | |
|---|---|---|---|---|---|
| 0,304;L | 192,304;L | 192,0;L | 0,0;L | 16,16;M | 16,136;L |
| 80,136;L | 80,16;L | 16,16;L | 176,16;M | 112,16;L | 112,136;L |
| 176,136;L | 176,16;L | 176,168;M | 176,288;L | 112,288;L | 112,168;L |
| 176,168;L | 80,288;M | 16,288;L | 16,168;L | 80,168;L | 80,288;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 ↑ for START and step through the commands using → (NEXT) until you find it. Position the pointer just before it using ← (PREVIOUS). MOVE the cursors to X=360,Y=150 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 DEL and then using a different scale. If you use too big a scale part of the door will be drawn off the screen. If the RC is off the visible screen then the '5' key may be used to centre it.

Make sure you have a door of scale 4 drawn. Now we want the door to be in Blue so we are going to use another command to do this. Position the pointer using the → (NEXT) & ← (PREVIOUS) keys so that it is immediately before the GOSUB for the door (ie X=360,Y=150) and then press CTRL & 3 to swap to PEN 3. Now position RC at 454, 302 and press 'B' for BLOCK, then press CTRL & 1 to change back to Pen 1. The BLOCK command fills the rectangle between BC & RC (approx) using the current pen. Make sure you have changed back to Pen 1, then the → (NEXT) key should execute the GOSUB command to draw the door on top of the blue rectangle. Finally 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 CTRL & COPY 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 CTRL & COPY 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 (↑) moves the drawstring pointer to just before the very first drawing command.

- NEXT (→) can be used to step through the commands one at a time, while PREVIOUS (←) 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 or MOVEA 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 subroutines to be drawn in one of eight sizes; 0 means no scale (full size), 1 to 7 specifies a size in eighths.

- * The BLOCK (B) command be used to fill the rectangle defined by BC and RC with the current Pen. Note that the width of the rectangle is always rounded to a multiple of 8 pixels!

**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 G 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 in a form that will auto-run when loaded using the RUN command.

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. 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 for the database saved from The Quill and a name for the final adventure. 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 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. Change system message 32 to read something like "Type in name of file", this is because game positions on a final adventure saved from the Illustrator are now saved to and loaded from a named file (to disc if a disc drive is available). In order to provide control over the graphics, entries similar to the following should be inserted in the Event table (having of course inserted suitable words in the Vocabulary):-

```
PICS ON   Conds                        ;allows graphics constantly
           Acts     LET  29 64
                    OK
```

```
PICS OFF  Conds                       ;graphics are never drawn
           Acts    LET   29 32
                   OK

PICS NORM Conds                       ;normal (first visit only)
           Acts    CLEAR 29
                   OK

LOOK _    Conds                       ;allows picture to be seen
           Acts    PLUS 29 128        ;in addition to text (i.e. R)
                   DESC
```

The picture for each location is normally drawn only when you first visit the location. Flag 29 however is used by the illustrator as follows:-
If bit 7 (128) of flag 29 is set the picture is drawn. ie. LOOK
If bit 6 (64) of flag 29 is set the picture is drawn ie. PICS ON
If bit 5 (32) of flag 29 is set the picture is not drawn. ie. PICS OFF. Whether or not the picture is drawn bit 7 of flag 29 is cleared. Note that flag 29 is otherwise 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

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

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

6/ 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 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 top of memory. 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 5 byte entry for each picture specifying if the picture can be drawn when that location is reached (i.e. if the location is not a 'subroutine') and the colours to be used for the picture.

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.

C The Shade Table

This contains the 16 patterns that are used by the SHADE routine.

## 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'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 ESC.

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 a subroutine (SCDUMP) is called. **picno.** must be specified.

Points to note:

a) Locations which are subroutines use INKS 1,24,18 & 26 as start up colours.

b) There is a limit of 254 pictures.

c) Due to the variety of printers/plotters available the SCDUMP subroutine provided saves the screen to disc/tape with a filename of PICxxx. This file may be reloaded from BASIC using the supplied program "SHOWPIC" and you may then use any of the published or commercially available screen dump routines to produce a hard copy.

Alternately the address of the machine code SCDUMP subroutine is given on The Illustrator start up screen. If you wish you can write your own printer driver and patch it into The Illustrator using the **Load Graphics** option on the Main Menu. The subroutine should end with a RET instruction, it should preserve all registers, it should not exceed 1K bytes and obviously it should not 'mess up' the firmware.

B **Graphics Start Table**

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

Amend **A picno.(ink 0 ink 1 ink 2 ink 3 border)**

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

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**

Saves the graphics database to tape or disc.

E **LOAD Graphics**

Loads a graphics database into The Illustrator. It will however load any binary file back to the address it was saved from. e.g. you would use 'E' to load an SCDUMP routine.

**Very Important**

If BREAK is pressed or an 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 ESC is pressed while a name is requested the graphic database will be unaffected.

F **LOAD database**

This allows the information from a Quill database to be transferred to the graphic database. Only the first few bytes are loaded in, (ESC or an 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.

G **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, or Run. Note the files are designed to auto run when loaded from BASIC using the RUN command.

H **Disc / Tape**

Allows you to choose between disc & tape.

I **Cat**

Catalogues a disc or tape in the same way as from BASIC.

**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 the current Pen then moves the Base Cursor (BC) to that position. The position plotted is an absolute position and only positions on the visible screen can be plotted.

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

MOVEA A Moves BC to RC without affecting the screen. The position moved to is an absolute position and must be on the visible screen.

LINE L Draws a straight line from BC to RC according to the current Pen, then moves BC to RC. The line is coded as a relative offset from BC.

FILL F The area below RC (relative) is filled using the current pen. Fill works by passing a pattern to the SHADE routine so the notes on SHADE apply also to FILL.

All the above use 3 bytes in the database. Relative distances are limited to ± 1022 in the X direction and ± 510 in the Y direction.

SHADE S The area below RC (relative) is shaded with one of a large number of patterns. The database contains 16 patterns (0-15), the default patterns are shown in Appendix A which also details how they can be changed.

The pattern used for shading is determined as follows:-
a) You are asked for 2 pattern Nos in the range 0 to 15. If you only want to specify 1 pattern then specify the same No. for both patterns.
b) The 2 patterns specified are OR'd together i.e.they are placed on top of each other.
c) Any Ink 3 in the resultant pattern is changed to the current Pen. (If the current Pen is Pen 3 then this has no effect).
d) If SHIFT S was pressed the resultant pattern is inverted ie Inks 0 & 3 are swapped and Inks 1 & 2 are swapped.

Notes on the shade (and fill) routine:-

1) The shade first works in a downward direction and then in an upward direction. For speed, when it is going down it doesn't look up and vice versa. Any areas the shade misses must be shaded seperately although careful choice of the start position for the shade will minimise this problem.

2) The area to be shaded must be in Ink 0 ie background. Any other Inks or the edge of the visible screen mark the edge of the area to be shaded.

3) The start point of the shade must be on the visible screen.

4) If the area to be shaded is 'too complex' then the shade will be abandoned. It has to do this to enable it to detect when it comes across an area which has already been shaded. Thus an area can only be shaded once as an already shaded area will be 'too complex' to shade again. You should not shade an area and then try to fill in the background with a fill command - patterns 14 & 15 can be used in combination with the other patterns to avoid this.

5) N.B. The firmware line drawing routine on the 664 is slightly different to that on the 464 such that a line draw from A to B may not plot exactly the same pixels on the two machines. This is unlikely to cause problems but when selecting a start point for a shade you are advised not to go immediately next to any lines already on the screen.

Shade uses 4 bytes in the database.

BLOCK B The rectangle defined by BC & RC (Absolute) is completely filled using the current Pen. The left hand edge will be rounded down to start on a pixel which is divisble by 8. The right hand edge will then be rounded up so that the width of the rectangle is a multiple of 8 pixels. This command uses the firmware routine SCR FLOOD BOX so it is faster than FILL but much less flexible.

Block uses 5 bytes in the database.

FREEHAND CTRL/COPY A toggle action (ie CTRL/COPY will turn on free hand if it was off and vice versa). When Freehand is on all movements of the cursor are stored in the database.

Each cursor move uses 1 byte!

**2) Colour Commands**

PEN CTRL/0-3 Changes to the Pen specified.

Pen uses 1 byte in the database.

**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 the relative commands, these are MOVE,LINE,FILL and SHADE. The other 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) Scale works by multiplying the relative distance by the scale value, dividing by 8 and rounding down. Thus only relative distances which are multiples of 8 will be scaled precisely.

c) Subroutines cannot be nested. If a subroutine includes a GOSUB command it will cause a 'Limit reached' error. If this happens, pressing any key will redraw the picture to just before the GOSUB, then the CLR key (which deletes the next command) can be used to delete the erroneous GOSUB.

## 4/ Editing commands

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

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

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

DELETE DEL The previous command is deleted and the screen redrawn.

DELN CLR Deletes the next command if there is one.

## Editor Error Messages and their meanings

BREAK

ESC was pressed during a peripheral operation or while editing, or a disc error occured.

I/O Error

An I/O error has occured. Note that an 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.

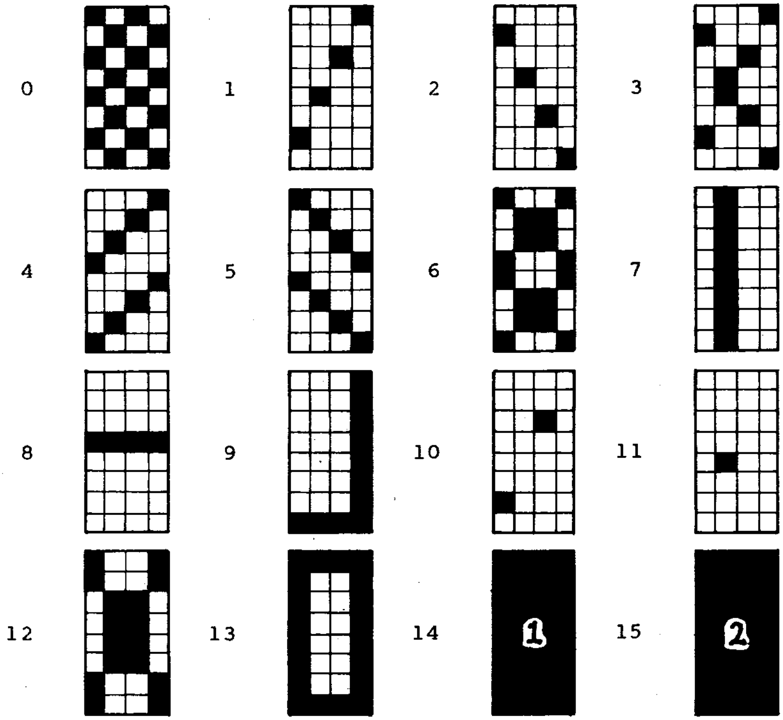
Limit reached

The maximum number of pictures is already present or a nested GOSUB has been found.

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

## Appendix A: The Shading Patterns

The default shading patterns are:-



Pattern 0-13 all use Ink 3

Pattern 14 is entirely Ink 1 & Pattern 15 is entirely Ink 2

These patterns were set up using the supplied BASIC program SHADES. If you wish to change the shade pattern then change the data statements in the BASIC program, run the program to save the file STABLE which can then be loaded into a graphics database using the Load Graphics command.

Note that the data statments contain the bit patterns which will be copied to the screen memory map so you will need to understand how the bits are used to encode each pixel (see the firmware manual).

## 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 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. When coding the picture it is best to draw as many of the outlines as possible first (split at suitable points by PLOT & MOVEA 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.
>
> 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

| | | |
|---|---|---|
| START | ↑ | Moves drawstring pointer to start of drawstring. |
| PREVIOUS | ← | Moves drawstring pointer back 1 command. |
| NEXT | → | Executes next command in string. |
| DELETE | DEL | The previous command is deleted and the display updated. |
| DELN | CLR | The next command is deleted. |
| Cursor move | | The cursor is moved using the keys around 5 thus; 7 8 9<br>4 6<br>1 2 3<br>by 2 pixels at a time. If SHIFT is pressed movement is by 20 pixels at a time. |
| CENTRE | 5 | The RC is moved to the centre of the screen. |
| PEN | 0-3<br>& CTRL | Changes the Pen which is being used for drawing. |
| PLOT | P | Plots the current position of RC, then moves BC to it (absolute) |
| LINE | L | Joins BC to RC with a straight line (relative) |
| MOVE | M | Moves BC to RC (relative) |
| MOVEA | A | Moves BC to RC (absolute) |
| BLOCK | B | The rectangle defined by BC,RC (approx) is completely filled using the current pen (absolute) |
| FILL | F | Fills the area below RC using the current pen (relative) |
| SHADE | S | Shades the area below RC with the pattern(s) selected which may be affected by the current pen and the use of SHIFT key (relative) |
| GOSUB | G | Allows another string to be drawn at the BC position. |
| FREEHAND | COPY<br>& CTRL | Allows cursor movements to be stored (relative) |

GILSOFT
*Home Computer Software*

GILSOFT
☎: (0446) 732765