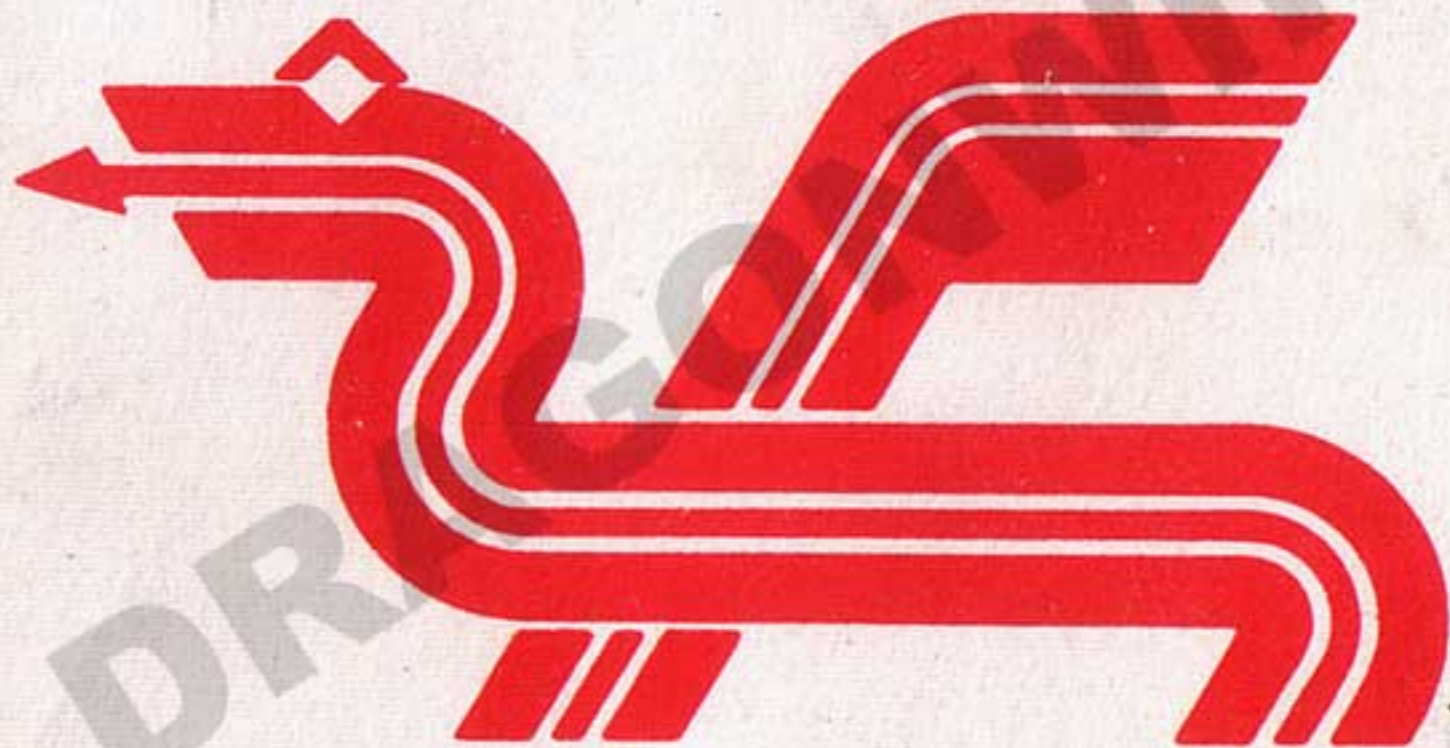


PIXEL EDITOR

AO520

CONTENTS



DRAGON

DATA LIMITED

WELCOME TO DRAGON

Dragon Data Limited welcome you to your new software for your Dragon computer. We hope that you enjoy using it as much as we enjoyed producing it for you.

Look out for new titles in the Dragon software range.

PIXEL EDITOR

AO520

CONTENTS

1	INTRODUCTION	Page 2
2	HOW TO LOAD THE PROGRAM.....	Page 3
3	GENERAL INFORMATION.....	Page 4
4	THE PIXEL EDITOR — HOW TO USE IT	Page 8
	AN EXAMPLE	Page 14
5	THE "GETPUT" ROUTINE.....	Page 16
6	THE MULTI-COLOUR PRINT ROUTINE — MCPR	Page 18
7	TECHNICAL APPENDICES — A — GETPUT	Page 21
	— B — MCPR	Page 24

LICENCE

Pixel Editor, in all machine readable formats and the written documentation accompanying them are copyrighted. The purchase of Pixel Editor conveys to the purchaser a licence to use Pixel Editor for his/her own use and not for sale or free distribution to others. No other licence, expressed or implied, is granted.

INTRODUCTION

The Pixel Editor can be used to create and edit graphics shapes. These could be anything from a letter in an alphabet, to a picture of a multi-coloured umbrella. We have used it to help produce the graphics in this program, and the screen that is shown while the program is loading is an example of what can be done. The graphics shapes that you create with the program can be saved in a cassette file and later used in either BASIC or machine code programs.

Pixels are the dots on the display screen, and the Editor allows you to create multi-coloured characters at this level without the need to know how the display works or even how the colours are stored within the machine. In addition, there is a BASIC routine on the tape that can be inserted into your programs, so that using the shapes you create is as easy as possible.

For the more experienced programmer, a second version of the routine is provided in relocatable machine code to enable faster and more compact programs to be produced.

You will also find a set of characters on the tape that can be loaded into the program and added to, or edited as desired. One of these sets is a lowercase alphabet — a, b, c, etc., — which can be used on the graphics screen.

2 HOW TO LOAD THE PROGRAM

- 1 Switch on your Dragon or, if you have been running another program, press the RESET key on the side of your Dragon. In both cases the letters OK will appear on your T.V. screen.
- 2 Insert the cassette into your cassette unit.
- 3 Next, type in CLOADM and then press the ENTER key.
- 4 Rewind the tape if necessary, and then press PLAY on the Cassette Unit. An S in the top corner of your T.V. screen shows that the machine is searching for the program.

- 5 The S on the T.V. screen will soon be replaced by F PIXED and shortly afterwards a message will appear giving you the title of the program and telling you how long it will take to load.
- 6 If the sound on your T.V. is turned up sufficiently you'll get an alarm call when everything is ready.

HAVING PROBLEMS? Please read on.

If an error message is displayed, or if the title of the program does not appear within a minute or so, something has gone wrong. Check the connections to your tape recorder, or, if they are correct, adjust the volume control setting on your tape recorder. Rewind the tape and start again from the beginning of the loading procedure.

3 GENERAL INFORMATION

You will find four items on the tape:

- 1 The Pixel Editor itself called PIXED.
- 2 The GETPUT routine.

- 3 The Multi-Colour Print Routine.
- 4 Collection of character sets.

We shall be looking at how to load and use each of these in the following sections. Before we do that, let's define some of the terminology we will be using.

The Dragon reference manual will give you the details of the different graphics modes and the memory layout. We will be dealing with the blocks of store known as graphics pages, specifically pages 5 to 8 in the Dragon memory map. We use these pages of store to hold the shapes we are editing.

A **pixel** is the term for each of the dots on a display screen; they are combined together in varying brightness and colour to make the picture. On the Dragon they can be addressed at different levels depending on the resolution required and the number of colours being used.

We will be using Modes 3 and 4, and in these, it takes four pages to make up a screen. While you are editing shapes you will see that the page being edited takes up a quarter of the screen.

It is important to note before using the pixel editor that whereas in normal counting one would expect to count from 1 upwards, in computing one counts from zero. This state has arisen because at the start of any counting in computing there is nothing, i.e., zero, so that is the first number. If, therefore, you want to have a block of memory accessed that is 12 columns wide, it would be addressed from 0 - 11, rather than 1 - 12 as you might expect.

The pixel editor enables up to three sets of shapes to be edited at any one time, and holds each set in a separate page.

The number of **addressable elements** in a page will depend on the mode in use. Mode 3 has 48 rows x 128 columns, giving 6144 addressable elements. Mode 4 has 48 rows x 256 columns. It therefore, has twice the resolution, but has only two colours available.

These addressable elements are grouped together into "grids" which are used whilst editing each character or shape. The **grid size** is the number of rows and columns used by each shape. For example, the grid size of the character sets supplied on tape is 8 rows x 6 columns. On a page, one can therefore fit 21 characters across (128 columns per page divided by 6 columns per character) with 6 down (48 rows per page divided by 8 rows per character line). This is shown in the figure on the next page.

0	1	2	→	18	19	20
21	22	23		39	40	41
105	106	etc.		123	124	125

The numbers on the grid are the numbers used to reference each shape.

If you define a grid size that does not fit the page exactly, there will be some room left at the edge. If, for example, you set a grid size of 24 rows of 12 columns, they will be held on the page as shown overleaf.

0	1	2	3	4	5	6	7	8	9
10	11	12	13	14	15	16	17	18	19

4 THE PIXEL EDITOR — HOW TO USE IT

If you have not already done so, load the Pixel Editor using the loading instructions set out in Section 2 and, when loading is complete, read on.

The pixel editor control works on a standard method of repeated choice. On running the program the user is presented with an immediate choice of whether to load, edit, or save a character set and this is the Base Menu. Throughout this system of menus it is only necessary for the user to follow the instruction. The second selection, edit, will lead on to another menu — the Main Menu — and this has the eight basic selections available within the editor. In each of these eight options there will be further options and so the whole program structure can be thought of as a series of "levels" through which the user descends to arrive at a specific action. To return back up through the layers it is simply necessary to press the # key for each level ascended.

BASE MENU — Load, Edit, Save

The Base Menu has three options available and these are as follows:

- ★ Load an old character set.

If there are three consecutive character or graphics sets on the cassette then the Auto Load function can be employed. This takes the first character set and places it into page 6, the second into page 7 and the third into 8. If only one character set is required, or if the sets are not consecutive, the manual option can be used to take named sets from the tape into a specified page of memory.

If you have a tape with three sets on it, insert it into the cassette unit and, after selecting the Auto-load option, press PLAY. The sets will be loaded in one by one, and the program reverts to the Main Menu when finished.

- ★ Edit

The use of this option takes the user straight through to the Main Menu — see below.

- ★ Save

This is used when a character set has been modified or created and it is now desired to save it on cassette for later use.

Any of the three graphics pages can be saved under any name on the tape, but it is important to name sets sensibly and to save them in an appropriate place on the tape.

Insert the tape to which you wish to save your character set, into the cassette unit. After specifying the name of the set, start the tape by pressing the RECORD and PLAY keys and start the program saving the data by pressing the SPACE BAR. If you save more than one set, make sure you leave an adequate space between each.

MAIN MENU — By selecting the Edit Option on Base Menu

★ Select Graphics Mode

The Graphics Mode selected will define the size of each addressable element in a shape. Mode 3 has larger elements and therefore less definition than Mode 4, but it has the advantage of being able to use four colours. If Mode 4 is chosen, the number of colours is reduced to two. The program starts in Mode 3.

★ Select Colour Set

There is an interdependency between the colours available and the graphics mode specified. Colours available for each combination are shown in the following table.

SET 0

MODE 3	MODE 4
Green (1)	Black (0)
Yellow (2)	Green (1)
Blue (3)	
Red (4)	
Buff (1)	Black (0)
Cyan (2)	Buff (1)
Magenta (3)	
Orange (4)	

SET 1

The number in brackets indicates the key to be used to select this colour.

The program starts using Colour set \emptyset .

★ View Character Sets

This option enables the user to look at the current state of any set of shapes, and after viewing enables any other set to be used until the # key is pressed.

★ Set Working Graphics

There are three graphics pages holding shapes that can be edited. In this option you can select the page you wish to use. The program starts using page 6.

★ Set Background Colour

When it is intended to create a new character or graphics set, the first thing to do is to choose a page and select a colour for it. This is done using this option.

The keys for each colour are shown under "Select Colour Set" above.

★ Set Character Grid Size

In the editor it is possible to vary the size of the shape being edited. Choose the number of rows and columns for each shape. The editor then splits the page into

grids of the size you have chosen. The program starts with a grid size of 8 rows x 6 columns.

★ Copy or Move Characters

It is possible, if required, to copy a shape being edited to another position. It is important to remember when doing this that the size of the area being transferred is the grid size chosen earlier and so, if it is required to transfer a different sized area the grid size must be changed first.

★ Edit Characters

Having set the various parameters described above it only remains to actually edit the shape of the characters or graphics. The editor will ask which character number is required and this will depend on the grid size. On selection, the program shows the page being edited across the bottom of the screen and a large version of the current grid above this. One square, the cursor, is flashing and can be moved around the grid, using the four arrowed keys on the left and right of the keyboard. The colour of the flashing cursor can be changed by using the keys 1 - 4 or \emptyset , 1 if in a two colour mode only.

The colours corresponding to each key are shown under "Select Colour Set" above. A square is painted by pressing the ENTER key. If a square is the wrong colour then it is simply necessary to repaint it.

★ An Example

Having seen the explanations you will want to try it out. If you have not already done so, load the program using the loading instructions set out in Section 2.

Unless you have a character set you wish to modify, type 2 to get to the Main Menu. As we are creating a new character set the first thing to do is to set the background colour (option 5). Now set the background colour of page 6 to yellow (value 2). There is now an odd display on the screen. What we see is pages 5, 6, 7 and 8. Page 5 is used as a palette by the program. Page 6 is the second quarter of the screen and is yellow which is what we wanted. Pages 7 and 8 are still unassigned and should be multi-coloured. Any key will now return us to the Main Menu.

Page 6 is now ready to draw on as it is a blank yellow page. This can be validated by choosing option 3 to view a page. As usual typing # will return you back up towards the Main Menu — type # until you are there. Now try editing a character (option 8). The editor has split the page into grids, each of 8 rows x 6 columns. They are numbered from 0 to 125 and character 21 beneath it and so on. Type Ø to edit the first character. There should now be a copy of page 6 at the bottom of

the screen and an expanded version of the character above it. Now try drawing a box around the outside of the square in blue by selecting 3 as the colour and using the ENTER and arrow keys as described in Edit Characters. You should now see this blue box in the top left hand corner of the copy on the bottom of the screen. When you've finished drawing one character type # and the editor will invite you to edit another character.

Try drawing an X in a different colour in character number 22 and then 0 in character \emptyset . To do this just paint in the centre four squares and you have an 0. Obviously you could go on drawing characters like this as long as you wanted; with this grid size there is room for 126 characters in a page.

We have drawn two characters:- Character \emptyset and character 22. Now let us see how to use these characters in another program. We shall first save them on a cassette.

Use the # key to return to the Base Menu. Now select Option 3 to save the character set.

First make sure that the tape in the cassette unit is not the one with the Pixel Editor on it. It is best to use a blank cassette which can be later used as a library of character sets.

The Pixel Editor is now telling you that you have been working with page 6. You may have been working on several pages so you have the choice of which page to save. We want to save page 6, so type 6 followed by the ENTER key. It is now necessary to type a filename, and it is a good idea to try and use a meaningful filename. In this case it would therefore, be sensible to try CHR-OX so type this and press the ENTER key. Now press the Record and Play buttons and make sure that the leader tape is past the tape heads. Now press the ENTER key and you should hear the page being saved (provided that the volume on your T.V. is turned up).

5 THE GETPUT ROUTINE

The way the character set can be used in BASIC relies on two of the Dragon's Graphics commands GET and PUT. It will help you to understand the GETPUT listing (see Appendix A) if you have read Chapter 10 of the Dragon User Guide which deals with the GET and PUT statements.

Firstly load the GETPUT program in much the same way as the Pixel Editor, but in this case it is a BASIC program so you will type CLOAD "GETPUT" rather than CLOADM. Follow with the ENTER key, and press PLAY on your cassette unit. When the program has loaded you will receive the OK message. Insert the tape with your characters into

the cassette unit. Now type RUN and press the ENTER key. Start the cassette unit playing and press the SPACE BAR. You should now see your character set being loaded into page 5 at the top of the screen. When it has been loaded you should press the SPACE BAR to continue. You will be asked to type in your grid size — if you want the same size shapes as shown then type Y; if you want to change the size type in the new values, following each with the ENTER key.

The GET and PUT areas now have to be selected and these are done as follows:

The GET position is the position within page 5 to get the shape from. This is calculated in addressable elements and you have to remember that in Mode 3, page 5 will be made up of 48 rows of 128 columns. The top left hand element is referred to as (0,0) and will match up with shape \emptyset . If your grid size is 8 rows of 6 columns, position (0,8) would give you the left hand character in the second row.

The PUT position is worked out in the same way, but is the position in the graphics screen that the shape will be displayed.

After you have typed in your GET and PUT coordinates, the character selected should appear on the screen. If you press the SPACE BAR, you can then move further areas into the graphics screen.

Try building a house with the 0 and X characters you created earlier.

By looking through the listing of the GETPUT routine you will see how to use the shapes you have created, in your own programs.

6 THE MULTI-COLOURED PRINT ROUTINE

The Multi-Coloured Print Routine — MCPR — uses Mode 3 character sets created with the Pixel Editor. It is provided in relocatable machine code which normally gets loaded into memory at \$2400. It provides a quicker method of displaying custom built characters on the screen, and gives a straightforward interface to the user. Having understood how to use the subroutine the user can access the internal parameters and extend its facilities still further. For a full description of MCPR and its parameters, their defaults etc., see Technical Appendix B.

Firstly, type PCLEAR8 to prevent the basic program overlapping MCPR and the character set. Then follow the standard loading procedure to load the routine into memory using a CLOADM"MCPR" command. The system will reply "OK" when it has loaded.

Having loaded the multi-coloured print routine, load one of the character sets provided on the tape into graphics page 5 with the following command: CLOADM'name of character set',&H600. Now try typing in the following:

For example:

```
10  REM MCPR TEST
20  PCLEAR8
30  A1=&H2400:PP$="": VPT=VARPTR(PP$)
40  POKE A1+4,INT(VPT/256): POKE A1+5,VPT AND 255
50  POKE A1+13,ASC("<")
60  PMODE 3,1: SCREEN 1,1: PCLS:COLOR 2,1
70  POKE A1+2,0: POKE A1+3,0
80  PP$="Would you like to <play<<@3A@2:Game 1<":EXEC A1
90  PP$="@3B@2:Game 2<":EXEC A1
100 GOTO 100
```

For better understanding of this program refer to the Dragon User Guide, and the Technical Appendix B.

If you run this you will see the power and speed of the MCPR. The text is displayed very quickly, the colour of the text changes and lower case characters are displayed. Once you understand how the routine is used within BASIC programs you can experiment with the various parameters shown in the appendix. Try changing, e.g.,

A1+13 — the line feed/carriage return key

A1+2,A1+3 — the X and Y start co-ordinates,

The strings and colours printed etc.

The MCPR routine, although slightly harder to use, is much faster than anything that could be produced using BASIC alone and is ideal when lower case etc, is required for fast text displays on the Dragon. Larger characters than these can be used by changing other parameters.

APPENDIX A

The GETPUT Program

GETPUT is a general program designed to take an area of the graphics pages and output it on to the screen. The program centres around two Basic commands in the Dragon's Graphics facility — GET and PUT. GET command gets an area of the current graphics page defined by two pairs of co-ordinates (X1,Y1) and (X2,Y2). (X1,Y1) pinpoints the top left-hand corner of the area to be transferred and (X2,Y2) pinpoints the bottom right-hand corner. In the program then, X1 and Y1 are input by the user and the size of the area input earlier is used to calculate (X2,Y2). The PUT statement works the same way — see chapter 10 of the Dragon User Guide for further information on the technical aspect of these two statements.

Between the GET and PUT statements there is a statement changing the current graphics screen from page 5 to page 1. This causes the GET statement to get the information from page 5 and the PUT statement to put the information out on to page 1, and using the SCREEN command to output page 1 on to the screen. In this way, whole pictures can be built up from graphics and character sets defined earlier, and, by changing the character grid size the size of the area being moved can be changed.

THE GETPUT ROUTINE

```
1Ø CLS:Pmode3,1:PCLS2:PCLEAR8
2Ø INPUT"TYPE THE NAME OF THE CHARACTER SET YOU WISH TO
LOAD (N FOR NONE)";Q$
3Ø IF Q$ = "N" THEN 1ØØ
4Ø PRINT@128,"LOADING CHAR SET INTO PAGE 5",,,
"START CASSETTE PLAYING",,,
5Ø PRINT "PRESS SPACE BAR WHEN READY"
6Ø IF INKEY$ = "" THEN 6Ø
7Ø Pmode3,5:SCREEN1,Ø
8Ø CLOADM Q$,1536
9Ø IF INKEY$ = "" THEN 9Ø
1ØØ DIMGP (59,39)
11Ø GX=12:GY=8
12Ø CLS:PRINT"SAME GRID SIZE(";GX"★";GY;)" :INPUT "";Q$
13Ø IF Q$="Y" THEN 16Ø
14Ø INPUT "HORIZONTAL (MAX 6Ø)";GX
15Ø INPUT "VERTICAL (MAX 4Ø)";GY
16Ø REM GET COORDS...
17Ø CLS:PRINT"CHOOSE GET COORDS":INPUT"X1";X1;INPUT"Y1";Y1
18Ø Pmode3,5
```



```
19Ø GET (X1,Y1)-(X1+GX-1,Y1+GY-1),GP,G
20Ø REM CHOOSE PUT COORDS
21Ø PRINT "CHOOSE PUT COORDS":INPUT"X1";X1:INPUT"Y1";Y1
22Ø PMODE3,1:SCREEN1,Ø
23Ø PUT(X1,Y1)-(X1+GX-1,Y1+GY-1),GP,PSET
24Ø IF INKEY$="" THEN 24Ø
25Ø GOTO 12Ø
26Ø PMODE3,5:SCREEN1,Ø:GOTO26Ø
```

APPENDIX B

MULTI-COLOUR PRINT ROUTINE

Dragon 32 High Resolution Graphics Text Print Routine

Version 1.1 Multi-colour
 In string colour switch
 Graphics mode 3 only

1 **Technical Summary**

This machine code routine displays text in high resolution graphics mode by reading a BASIC string and transferring characters (from a character set held somewhere in RAM) to the graphics screen area.

1.1 Facilities

The text string can be displayed starting from any point on the screen. Parameters DX and DY give the coordinates of the top left corner where the string starts.

The BASIC 'COLOR' command can be used to specify the foreground and background colours of the text.

The colour of the lettering (foreground colour) can be changed at any point in the string with a user-definable escape character (normally @) followed by the desired BASIC colour number (1-8). This greatly simplifies the printing of multiple coloured menu screens.

A user-specified character (normally '<'), at any point in the string will cause a return-newline on the screen: the remainder of the string is displayed one character-line down, starting from the left border of the screen.

The 'configuration parameters' (which determine character size, spacing, and character set type) have default values loaded with the routine. They are set for use with the standard character set supplied, and can be changed for use with other character sets.

1.2 Restrictions

In its present form the routine will only work in graphics mode 3.

No range checking is performed on any of the routine parameters, except the 'hidden parameters' which are checked by BASIC.

The colour change escape character (normally '@') may not appear in the string except when followed by a colour number.

1.3 Code

The routine is supplied in fully re-locateable object code from \$2400 (graphics page 6). It can be moved to any position in store.

Where A is the base address of the (relocated) routine, A is also the entry point and A+2 through to A+15 is the parameter area.

2 Parameters

2.1 Hidden Parameters

The hidden parameters are taken from the BASIC system storage in page 0 and hence do not require special attention by the programmer.

(a) Target Screen

Parameter name	address	system use
DBP	\$B7,\$B8	top of current graphics screen

Characters are printed to the screen area specified by the most recent PMODE command — as with other graphics functions such as PUT, LINE etc.

(b) Target Colours

Parameter name	address	system use
DFCL	\$B2	current foreground colour
DBCL	\$B3	current background colour

The character colouring is determined by the BASIC foreground and background colours (unless the colour change character is encountered in the string). Where a normal single colour character set is being used, COLOUR F,B will cause the string to be displayed with characters colour F on a background strip colour B. Remember the 'C' command used in a DRAW string also changes the current foreground colour. Any other colours in a non-standard character set are unaffected by the colour switching software.

2.2 Functional Parameters

The functional parameters are passed in the first part of the parameter area. Values must be poked in before the routine is called.

Name	Address	Use
DX	A+2	Target X coordinate (top left corner). DX is altered by the routine such that on exit it points to the next character position following the string; consecutive calls to the routine will cause strings to be displayed one following the other. Beyond the right border the screen DX wraps around in a non-useful way. An end-of-line character in the string causes DX to be reset to zero.
DY	A+3	Target Y coordinates. DY is not altered unless an end-of-file character is encountered in the string, in which case DY is incremented by RSP (see 2.3). There is no range checking on DY which which may exceed 191 — as if the screen extended downwards.
VPT	A+4 A+5	String variable pointer. VPT must contain the 2 byte address of the basic 5 byte string descriptor. The name PP\$ is recommended for all calls to the routine so that the following initialisation code can be executed once at the start of the program.

PP\$="" :VPT=VARPTR(PP\$):POKEA+4,INT(VPT/256)
:POKEA+5,VPT AND 255.

2.3 Configuration Parameters

The configuration Parameters all have default values loaded with the object code. These are set up for use with the standard character set supplied, held in graphics page 5, coloured blue on yellow, with CR as the end-of-line character and '@' as the colour switching character.

Name	Address	Default	Use
SBP	A+6	\$1E00 (page 5)	Character set base address. The character set may be located anywhere in RAM.
GL	A+8	8	Character height.
GN	A+9	6	Character width in mode 3 addressable elements.

3 Recommendations for use: See Section 6 for an example of its use

3.1 Other Character sets

When using other character sets the following considerations should be borne in mind:

- (a) The characters are indexed by the ASCII code, minus 32; thus space is normally the first character in a set. Where non-ASCII characters are to be printed one at a time this routine should not be used. Use the GETPUT Program instead.

- (b) Where more than two colours are used in the set, only the Character Set Foreground Colour and Character Set Background Colour will be detected and switched to the foreground or background colours; other colours are unaffected. Where multi-coloured characters must be faithfully reproduced, select the same target foreground and background colours as are set in the Character Set Foreground Colour and Character Set Background Colour.

IN GENERAL, for non standard character sets which fundamentally differ from ASCII (graphics sets etc.) the GETPUT program method should be used.

3.2 In-String Colour Switching

The Character colour (foreground) can be changed mid-stream with the colour switch escape character, normally '@', immediately followed by a single digit giving the BASIC colour number required. This change is only for the duration of one call, the BASIC system foreground colour is not altered. See the example program in Section 6 for a typical application. To disable the facility change the Colour Switch Escape Character to some character which is never used.

TAPE CONTENTS

- (i) PIXED PIXEL EDITOR (M/C)
- (ii) PIXEL ED ..
- (iii) GETPUT BASIC A
- (iv) MCRK M/C
- (v) ASCII 3/1
- (vi) ASCII 2/1
- (vii) ASCII 3/1

WARRANTY STATEMENT

Dragon Data products sold by authorised dealers are offered under the provisions of the Supply of Goods (Implied Terms) Act 1973. In order to provide a satisfactory service to our customers, Dragon Data Ltd. warrants the following:

- I. All faulty components due to defective manufacture will be replaced free of charge for a period of 12 months from the original date of purchase.
 - II. All labour and/or services will be provided free of charge to repair your Dragon Data product which fails in its specified performance due to manufacturing defects for a period of 12 months from original date of purchase.
- NB (a) The guarantee is restricted to the original purchaser.
(b) Claims will not be accepted if any unauthorised modification is made to the product or if the serial number or guarantee labels have been removed or defaced.
(c) Dragon Data's liability is limited to the cost of repair or replacement (at Dragon Data's discretion) of the defective product.

This warranty is offered as an extra benefit and does not affect customers' statutory rights.



WWW.DRIVENRACING.COM