

Issue 0
Sept 87

Script

The LocoScript
Newsletter

Contents

News	2
What's been going on that concerns LocoScript and the PCW.	
LocoMail	4
Labels made easy with LocoMail.	
Changes	7
A quick look at some of the more subtle changes from LocoScript 1	
Layouts	8
How LocoScript 2's Layouts work	
Letters	12
Extracts from Locomotive's Customer Support postbag.	
Printers	15
Solutions to some of the problems with different printers.	
PostScript	16

Script is Published by:
Locomotive Systems Ltd
Allen Court
Dorking
Surrey RH4 1YL

© 1987 Locomotive Software Ltd

Welcome to *Script*

Script is a newsletter for users of LocoScript, written and produced by the people who created LocoScript. It will be published every two months.

Through *Script*, we will be explaining areas of LocoScript which appear to be causing problems, describing enhancements you can make to your word processing system, and generally looking at ways you can make better use of LocoScript.

Each issue will feature one or two items in depth. In this issue, we look at Layouts – a complex concept for people new to word processing. Users are also finding LocoScript 2's Layouts difficult as they are not exactly the same as LocoScript 1 – they do the same job, but are used in a different way.

In future issues, we'll be looking at imaginative uses of templates, special printer features and defining your own characters.

In addition, we'll be including some shorter explanations of areas that are puzzling LocoScript users (and here we'll feed back advice from our customer support department), plus a selection from the letters we have received. For example, in this issue we explain how and why the definition of a paragraph has changed, and some of the aspects of using different printers that aren't covered in the User Guide.

And, of course, we'll be covering LocoMail and LocoSpell too. LocoMail will have its own regular feature – in this issue, how to print names and addresses on different types of labels.

In future issues we plan to include items which will make it easier for you to use LocoScript 2, starting in the next issue with some quick reference keyboard charts, to help you type all the characters LocoScript 2 supports.

We hope you like *Script*.

This special introductory issue is being sent to everyone who bought LocoScript 2. If you're one of the many who have already subscribed, don't worry – you'll get a full six issues after this one. If you haven't subscribed so far, we hope you will – the more subscribers we have, the better we can make *Script*!

Script costs just £9.95 for a year (six issues):
to subscribe, just send back the enclosed order form .

News

The biggest news since LocoScript 2 is of course Amstrad's release of the PCW9512. Although not officially available until late September, the new PCW is no longer a dark secret.

The PCW9512 has a number of new features – a new shape, new keyboard layout (with the same keys), single 720K disc drive and dedicated daisy wheel printer. From Locomotive Software's viewpoint, the most important feature of the 9512 is that it comes with LocoScript 2 included as standard, plus LocoMail and LocoSpell.

The versions of these programs supplied with the 9512 are fully compatible with the versions we supply for the 8256 and the 8512. So you will be able to transfer documents from an 8256/8512 to a 9512 simply by moving the disc. Equally, you'll be able to move documents from a 9512 to an 8512 that's running LocoScript 2.

What you won't be able to do is move documents back to an 8256, as this only has a 180K disc drive and so cannot read anything written on the 9512's 720K drive.

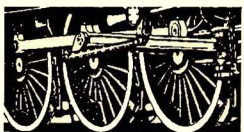
It's also just not possible to run the original LocoScript on a 9512. All documents prepared on a 9512 will be LocoScript 2 documents, and to read them on an 8512, you'll have to have LocoScript 2.

One possible drawback of the new machine is that it doesn't have a matrix printer. Whilst daisy wheel printers are very good when you want really high quality letters, they do have a very limited repertoire of printing characters. So, if you've got used to printing LocoScript 2's 'fancy' characters – Cyrillic characters, special mathematical symbols, etc – on your 8256/8512, you'll find that you can't do this on the 9512.

LocoScript 2 stranded in Gulf War – Untrue News story of the year

When one of our customers asked Dixons' Wimbledon branch for a copy of LocoScript 2, he was told that there were no supplies as the ship carrying LocoScript 2 was stranded in the Gulf War!

But don't worry, this rumour is entirely false – LocoScript 2 is made in England. The discs (admittedly of foreign manufacture) are duplicated in Northampton and the manuals printed in Woking.



**LOCOMOTIVE
SOFTWARE**

&

**The British
Computer Society**



The British Computer Society has asked Locomotive Software to give a presentation to its North London Branch in November. Whilst this is a BCS event, it is open to anyone interested.

Locomotive Software's Howard Fisher will be talking about what goes on behind the scenes at Locomotive. He will outline the process of the development of Locomotive's various products, chiefly with reference to the Amstrad computers. The talk will involve some demonstrations of the various products and there will be an opportunity to meet some of the people who work behind the scenes at Locomotive.

The joint event with Locomotive will be held on the evening of Wednesday November 4th at the Hendon Hall Hotel, London NW4. For a free ticket, don't write to Locomotive; instead please send a stamped addressed envelope to:

Howard Gerlis, MBCS,
The British Computer Society
Hon Secretary, North London Branch
56 Courthouse Road
North Finchley
London N12 7PJ

LocoScript 2 Upgrades and New Products

Printers

The printer files supplied on the LocoScript 2 disc support Diablo 630 daisy wheel printers, Epson FX80 matrix printers, and other printers which obey the same commands as either of these. We now know of a number of popular printers which don't fit into these categories and so have added support for these to the latest version of LocoScript 2.

The following .PRI printer files are now available to support these extra printers (or in some cases, eg Juki, to improve our existing support).

Daisy wheel printers

D630.PRI	Diablo 630 Panasonic KX-P3131 Brother
JUKI6100.PRI	Juki 6100
QUME.PRI	Sprint 11 LetterPro
D1610.PRI	Diablo 1610 Silver Reed EXP range
GAKKEN.PRI	GPR2000 Uchida Quendata

Matrix Printers

FX80.PRI	FX80 Draft
FX80NLQ.PRI	FX80 NLQ
OKI.PRI	OKI Microline290 series

Special Characters

The other improvement we've made to LocoScript is to add a feature that lets you define your own special characters. We've actually done rather better than we had initially announced: there are actually *sixteen* characters which you can design yourself for display on the screen and in both Draft and NLQ forms on the built-in printer. We also supply a program which you can use to modify the characters easily.

How to upgrade your LocoScript 2

The support for new printers and the handling of special characters are new features in version 2.12 of LocoScript 2. You can upgrade to this version in the usual way, by returning your LocoScript 2 Master Disc together with the upgrade handling charge of £5.

The Printwheel Constructor Disc and the LocoScript Keyboards Disc are two new products, each costing £14.95. These require LocoScript 2 version 2.12 (or later) – a free upgrade is available when you order these new products.

To order your upgrade or these new products, please use the order form enclosed with this issue of Script.

Printwheels

The problem with a daisy wheel printer is that the printwheels you use don't necessarily have the same characters on them. Elsewhere in Script (page 15) we discuss the problems this can cause when using unusual characters (like £!).

We have developed a special computer program – the Printwheel Constructor – which you can use on your PCW to configure LocoScript 2 to use the characters on your particular printwheel. After using this, you should never get # when you mean £ again!

Keyboards

There have been many requests in the past to use say an English LocoScript on a French AZERTY keyboard. Until LocoScript 2, that was not possible.

We have now changed the way LocoScript works so that you can mix and match different nationality keyboards with different language LocoScripts. We can now support the Danish, Finnish, French, German, Italian, Norwegian, Spanish, Swedish and USA keyboards, regardless of the language which LocoScript's messages are written in.

If you simply want to use the English version of LocoScript on a foreign keyboard, help is at hand immediately. You just need to get the LocoScript Keyboards Disc for the PCW8256/512. We will also be making available a Keyboards disc for the PCW9512 in due course.

When the full range of national versions of LocoScript 2 is released, these too will be able to work with other keyboards if you have the Keyboards Disc.

Other Languages

We are now producing versions of LocoScript 2 in a variety of different languages. These will become available over the next few months. As each language becomes available, this will be announced in Script.

LocoMail – Labels

LocoMail and LocoScript together provide a very powerful set of tools for producing labels automatically. LocoMail selects the addresses from your mailing list: LocoScript looks after how they look on the page. In this first LocoMail feature in *Script*, we look at the different parts to this job.

Producing labels automatically which are correctly positioned on your labels stationery is essentially a team effort by your mailing list, your labels Master document, LocoMail and LocoScript.

- The mailing list provides the names and addresses.
- The Master document provides the outline for each label that is produced.
- LocoMail processes the Master document, inserting the information specified from the mailing list. At the end of each ‘pass’ it produces a complete LocoScript document which it then hands to LocoScript for printing.

Obviously, it is important that LocoMail selects the right information from the mailing list. This means setting up the data file in such a way that LocoMail is able to extract the correct information

and putting the correct LocoMail commands in the Master document. We describe both these tasks in this article.

However, this only ensures that the correct information is inserted into each filled document. It does nothing to ensure that the labels themselves are positioned correctly on the paper.

The key to getting this aspect of the job right is to remember that each pass through the Master document produces a separate document. So a Master document set up to produce labels will produce large numbers of separate documents – each containing just one name and address. And after printing each one, LocoScript will tell the printer to move to the top of the next page.

The distance the printer moves in order to get to the top of the next page depends purely and simply on the page length of the paper the printer is currently set up

for. This is true for any document you print. So the problem becomes how to set up the printer for your labels stationery.

In LocoScript 2 the answer is very easy: you simply set up the document you want to print for the right type of paper. With LocoScript 1, it was less straightforward because you had to go into Printer Control State every time you wanted to change paper and set up the printer for this paper by hand.

The documents LocoScript prints when you are merging a data file with a Master document are essentially copies of the Master document, so for LocoScript 2 in particular, the crucial part of the operation is setting up the Master document for the right kind of stationery. This is covered for all types of different stationery in Session 19 of the LocoScript 2 User Guide: what we shall look at here is the specific case of setting up a document for labels stationery.

Working with labels stationery

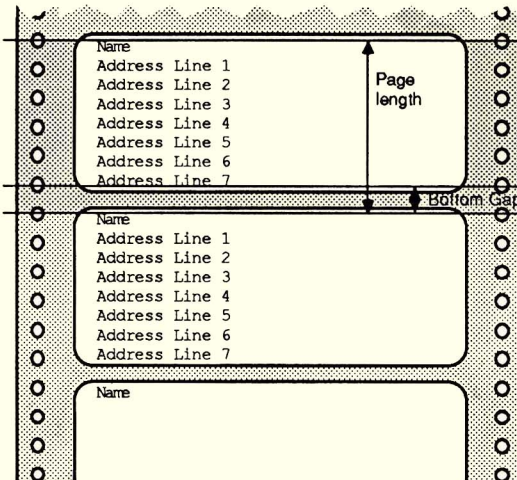
The key to producing labels on labels stationery is to set up the Master document for this kind of stationery.

The first thing to consider is the dimensions of this type of paper.

In particular, you want to know the repeat distance – ie. the distance from the top of one label to the top of the next – and the size of the gap between the labels. The repeat distance is the page length you need so that the printer moves to the top of the next label after printing each name and address; the gap is needed to ensure that the names and addresses aren't printed on the backing paper.

The problem with continuous labels is that because of the sprocket holes down the side and the position of the tractor feed mechanism on your printer, you can't always position so that printing starts at the left edge. Use the Left Offset feature to adjust the left edge to fit your labels and printer.

Trial and error is usually the best way of coming up with a suitable Left Offset the first time round. But once you have



found this, it's a good idea to always use the tractor feed set in the same position or to mark the printer so that you can readily find this position again.

When you create a Master document, it is automatically set up for the same type of paper as the template (TEMPLATE.STD document) for the group you are working on. Unless this template is itself set up for your labels stationery, you will need to go into your new document's Document Set-up and set up the dimensions of the paper.

Setting the paper type

Press **[F1]** for the Actions menu, and select Document Setup. As usual, the function key menu options change to those appropriate to setting up the document. Choose the Page option by pressing **[F5]**. The Page menu now pops up to give you the chance to change various aspects of the paper and how the text is placed on it.

So, what do we need to set for our labels? First, we have to specify the page size. Select Paper Type from the menu. This then displays a further menu of standard paper types. Unless you have already set up a paper type for your labels, it is unlikely that you'll have one that is suitable – so you'll have to create a new one. Choose the paper type which is nearest to your labels (if you don't have anything better, choose 11continuous because, like labels, it is a type of continuous stationery) and press **[E]**. Now move the cursor down to Show Paper Type and press **[ENTER]**. This will show the details and you should change them to match your labels as follows:

First, type in the new name for your paper type, say 1½" Labels, then move the cursor down to set the other

items. With labels, the Height is simply the repeat distance in lines (ie. the distance in inches from the top of one label to the top of the next multiplied by the number of lines per inch (6)). If we have 1½ inch labels our page size is simply 1½ times 6, giving 9. We also have to set Top and Bottom Gaps to mark out the section of the page you want to avoid printing on. It's a good idea to have a Top Gap of zero lines, as then you can position the labels in the printer exactly where you want the first label to begin. Set the Bottom Gap to the number of lines you must leave blank to avoid printing on the space between the bottom of the label and the start of the next label.

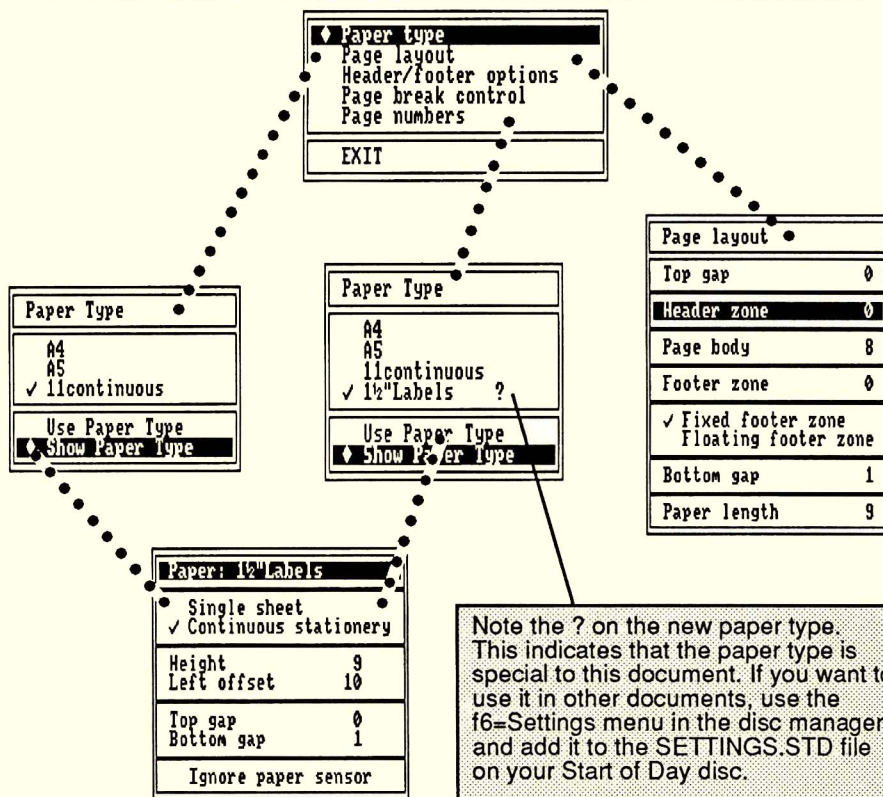
Finish by pressing **[ENTER]**, moving to Use Paper Type and pressing **[ENTER]** again. This returns you to the main Page menu. Now you need to adjust the page layout to fit within the new paper type you have just created. (You may sometimes get a message telling you to do this.) Move down to the Page layout entry and press **[ENTER]**. The menu is replaced by the Page layout menu from which you can set the necessary items (but note how the paper length and top and bottom gaps you've just set up are already shown). For straightforward labels you don't need a header or footer, so set the Header zone and Footer zone both to zero to give the maximum number of lines to the details you will need on the label. Once again, press **[ENTER]** to return to the Page menu.

You have now set up the page to match your labels, so press **[EXIT]** to move down the Page menu to the exit option and then press **[ENTER]**. This takes you back to the Pagination screen with the header and footer text displayed. Delete any header and footer text you have here, because otherwise LocoScript will try to print it. It doesn't ignore it simply because you have set the Header and Footer zones to zero. Finally press **[EXIT]** to leave Document Set-up and return to editing the document proper.

That's set up the document to fit onto the continuous labels – essentially a LocoScript task. Now you need to type in the LocoMail commands to take the details from your address list and print them onto the labels.

The Mailing list ...

Names and addresses are potentially quite complex objects to have in a data file because of the number of different ways in which you may want to present the information in the finished document. Addresses, for example, are typically laid out in quite different ways in the address part of a business letter, on a label, on one line in a list of contacts and so on.



LocoMail gives you a lot of flexibility in how you represent an address in a data file. If you only want the address in one particular form, you can simply use a single LocoMail data item "Address" to store the whole address. Or you can use a separate data item for each part of the address and use carriage returns in your master document to arrange each address neatly. You just need to define a record pattern to match what you want.

For example, you might split up an address into the following components: Apartment/Office address, Building, Street, Village, Town, County, Postal code, Country. You actually need a long list to cover all addresses adequately. Using a separate LocoMail name for each of the items gives a record pattern like:

```
Name ←
Rm ←
Bldg ←
Street ←
Village ←
Town ←
County ←
Postcode ←
Country ↓
```

Then you will have to ensure that each of the names and addresses in the following data records matches your (rather long) pattern. For example, William Smith's data record might look like:

```
William Smith ←
Flat 10 ←
West Lodge ←
223 The Broadway ←
Ameswell ←
Gloucester ←
Glos ←
GL23 7HE ←
England ↓
```

... and the Master document

Having taken the trouble to impose this structure on your names and addresses, it is very easy to produce a master document to print them out.

As with all LocoMail master documents, first produce a sample document using real data. As we want to fit the name and address into 8 lines on our inch and a half labels, a typical label might look like:

```
William Smith
Flat 10 West Lodge
223 The Broadway
Ameswell
Gloucester
Glos
GL23 7HE
England
```

When you are happy with this, turn each part into a LocoMail command using the names you used in the record pattern for the data items. This gives the required master document suitable for Merging with the names and addresses data document:

```
(+Mail)Name (-Mail) ←
(+Mail)Rm (-Mail) (+Mail)Bldg (-Mail) ←
(+Mail)Street (-Mail) ←
(+Mail)Village (-Mail) ←
(+Mail)Town (-Mail) ←
(+Mail)County (-Mail) ←
(+Mail)Postcode (-Mail) ←
(+Mail)Country (-Mail) ←
```

LocoScript handles how the information is arranged on the paper and LocoMail takes care of the actual information.

More than a simple label

Having set up the data with this flexibility, you can now go on to produce more sophisticated LocoMail master documents. One very common requirement is to use labels which are arranged in groups of two or three across the backing paper. We'll now take the simple master document above and see how it can be extended for use with "n-across" labels.

The first step as always is to set out a sample document using typical data. Here, with labels three across the paper, we might have:

William Smith Flat 10 West Lodge 223 The Broadway Ameswell Gloucester Glos GL23 7HE England	John Freeman Suite 210 Geers House 53 Fen Rd Litton Newmarket Cambs CB4 6TR England	Andrew Garden 1st Floor Old Hall 5 High St Westerham Sevenoaks Kent SS5 7XX England
--	--	--

Now consider how LocoScript will print these labels out. It will print all three names, then all three flat and building lines, three streets and so on – with tabs (set up in the Layout) between the three items so that the three labels are lined up neatly. This is the clue to the LocoMail commands that will be needed to print the addresses. The LocoMail 'program' works with groups of three names and addresses, only printing out a line when we have all the information (or have reached the last address!)

What you need is a number of variables, each representing a line of information, and a LocoMail "Program Unit" to insert the information into these lines. (If you are not sure what a Program Unit is, look at Chapter 8 of the LocoMail Guide.)

So the first task for LocoMail commands in the Master document is to set up some names to hold the information lines, and make each one blank. (It also sets up some names with fixed information for use later.)

```
(+Mail) Nameline="" ←  
Addline1="": Addline2="": Addline3="": Addline4="":  
Addline5="": Addline6="": Addline7="" ←  
Tab="→" ←  
Spaces=" " (-Mail)
```

Then it needs to repeat three times (once for each label) the program unit to append the various details from the current name and address and move on to the next name and address. The LocoMail command to do this is:

```
(+Mail) @Loopcount %Loop (-Mail)  
where Loopcount is used to ensure that Loop is executed three times. What actually happens is that the LocoMail instructions stored as the program unit Loop are obeyed while Loopcount is non zero. So one of these instructions has to make Loopcount count down 3,2,1,0 – otherwise it would never finish!
```

The instructions stored as Loop are:

```
(+Mail) Nameline=Nameline & Name & Tab  
Addline1=Addline1 & Rm & Spaces & Bldg & Tab  
Addline2=Addline2 & Street & Tab  
....  
Addline7=Addline7 & Country & Tab  
Loopcount=[Loopcount-1]  
$( -Mail)
```

The first of these adds the Name of the current name and address to the righthand end of the item we're using to hold the first line and then adds a tab character to space out ready for the next item. The second does a similar operation on the first address line, but it also adds some spaces and the building name before finishing with the tab character. This continues until all the items from the name and address have been recorded in the appropriate line. The penultimate LocoMail command decrements the counter we're using to go round three times and the final command \$+ moves on to the next name and address (see page 101 of the LocoMail Guide).

Repeating this three times has the required effect of creating the lines of the labels. These instructions are stored with the name Loop by the LocoMail command:

```
(+Mail) Loop=" ... " (-Mail)  
where ... stands for the required commands.
```

Now all that is left is to print out the lines. This is done by a mixture of LocoMail commands and LocoScript formatting characters as follows:

```
(+Mail) Nameline (-Mail) ←  
(+Mail) Addline1 (-Mail) ←  
(+Mail) Addline2 (-Mail) ←  
...  
(+Mail) Addline6 (-Mail) ←  
(+Mail) Addline7 (-Mail)
```

You now need to repeat this with the next three addresses and so on. LocoMail essentially does this for you because it automatically moves onto the next record before starting to fill the master document again.

But, there's a problem! The stored commands in Loop process a record and finish by moving onto the next record. When LocoMail starts again to process the master document it too moves to the next record. Together, this means that LocoMail would miss out a record in going from one set of three labels to the next. To overcome this, you need to add another LocoMail command, \$-. This stops LocoMail moving on to the next record when it starts again with the master document.

The complete LocoMail 'program' with all these features is shown below.

The final thing to consider is what will happen at the end. There might not be an exact multiple of three names and addresses. In fact this doesn't matter, since when a \$+ command moves past the last record, it substitutes a dummy record with all items blank. So the last one or two times round the loop, blanks will be inserted.

The complete LocoMail program

```
(+Mail) Nameline="" ←  
Addline1="": Addline2="": Addline3="": Addline4="":  
Addline5="": Addline6="": Addline7="" ←  
Tab="→" ←  
Spaces=" " ←  
Loop="( +Mail) Nameline=Nameline & Name & Tab ←  
Addline1=Addline1 & Rm & Spaces & Bldg & Tab ←  
Addline2=Addline2 & Street & Tab ←  
Addline3=Addline3 & Village & Tab ←  
Addline4=Addline4 & Town & Tab ←  
Addline5=Addline5 & County & Tab ←  
Addline6=Addline6 & Postcode & Tab ←  
Addline7=Addline7 & Country & Tab ←  
Loopcount=[Loopcount-1] ←  
$( -Mail) " ←  
Loopcount=3 ←  
@Loopcount %Loop ←  
Nameline (-Mail) ←  
(+Mail) Addline1 (-Mail) ←  
(+Mail) Addline2 (-Mail) ←  
(+Mail) Addline3 (-Mail) ←  
(+Mail) Addline4 (-Mail) ←  
(+Mail) Addline5 (-Mail) ←  
(+Mail) Addline6 (-Mail) ←  
(+Mail) Addline7  
$- (-Mail)
```


LocoScript 2 Changes

Positioning the Margins

The way LocoScript measures margins and tabs on the ruler line has changed. In a LocoScript 1 document, margins and tabs were measured in the "Base Pitch" for the document. In the absence of any explicit Layout codes, the text in the document was also in the Base Pitch. Consequently, if you changed the pitch for the document you also changed the pitch for the margins and tabs and so changed their positions.

In LocoScript 2 this has changed. Now, the margins and tabs are measured in a special pitch, the Scale Pitch. So, if you use Change Layout to change the pitch of the characters in your document, you will not change the position of the margins etc.

Ideally the Scale Pitch of your Layouts should match the pitch used for the majority of the text in the document. To set it, select Change Layout as usual, but

select the special Scale Pitch option to set the Scale Pitch to the value you require. We recommend that if you are using 12 pitch or proportionally spaced text, you should set the Scale Pitch to 12, and if you are using 10 pitch text, set it to 10.

Once set correctly, you should not need to adjust the Scale Pitch any more. Of course, if you are using a number of different Layouts in a document, you should give them all the same Scale Pitch. It is unwise to mix different values of the Scale Pitch in a single document.

When is a paragraph not a paragraph?

LocoScript 2 contains a number of subtle changes. Generally, the users of LocoScript 2 haven't even noticed these. However, one change a few people have picked up is the difference in the way LocoScript 2 treats paragraphs.

A LocoScript 1 paragraph had to contain lines which had been wrapped at the right margin. This meant that sections of letters such as addresses, or the date, or even "Dear Sir" were not in fact treated as paragraphs. This surprised a lot of people. Also, when LocoMail was introduced, it proved rather difficult to use LocoScript to prevent individual addresses being split across page boundaries.

A LocoScript 2 paragraph is simply text up to the next "gap", where a gap could be two adjacent carriage returns, a carriage return with carriage return extra greater than zero, a form feed etc. This definition is more intuitive than the LocoScript 1 definition, and in fact, as most people tended to use two carriage returns when using LocoScript 1, most people have not noticed the difference.

The new definition means that the `PARA` key will not always behave the same as

on LocoScript 1, but usually the movement it now makes is the one that was wanted in the first place!

But there is unfortunately a side effect to the change. The change of definition can affect the way in which a document is split into pages. This seems to have chiefly affected journalists (a naturally shy and retiring breed of LocoScript user), many of whom lay out their text by separating paragraphs by a single carriage return and indenting the following line.

Under the old scheme this was fine since the original LocoScript treated text wrapping round followed by a carriage return as a paragraph. These paragraphs were then dutifully not split across a page boundary, or were split to avoid widows and orphans. The new definition unfortunately does not see these as paragraphs! Consequently, the paragraphs are split quite differently from LocoScript 1.

If you have been suffering from this problem, the simplest thing to do is to Exchange `↵` for `↵↵` throughout your document. This will ensure that the paragraphs are again treated as paragraphs, but will make the documents

take a little more room. More sophisticated solutions involve using LocoScript 2's line spacing features.

Suppose you are using single line spacing at the moment. With just a single carriage return at the end of each paragraph, each new paragraph starts on the next line. What LocoScript needs to recognise the end of the paragraph is either two carriage returns or a non-zero CR Extra Spacing. So we need a way of giving it one or other of these, while still starting the new paragraph on the next line.

As it happens, you get exactly the same position for the new paragraph, both on the screen and when it is printed, by having a CR ExtraSpacing of $\frac{1}{2}$ and changing line spacing to half a line for just the last line of each paragraph. LocoScript 2 recognises this as the end of a paragraph because of the non-zero Extra spacing.

So, the solution is to set CR Extra spacing to $\frac{1}{2}$ and to replace the carriage return at the end of the paragraph by:

```
(+LSpace½)↵  
(+LSpace1)
```

To make it easy, simply put this sequence of codes and characters into a phrase and paste it in instead of pressing the carriage return key.

Of course, if you normally use line spacings other than one, you should vary the codes accordingly. For example for LSpace 2, set Carriage Return extra to 1, but change the line spacing to 1 for the carriage return.

- Our thanks to Mr Roger Taylor on whose suggestion we based the second solution to this problem.

We realise that this is not an ideal solution, and are investigating better solutions.

LocoScript 1 defined a paragraph in the following rather backward manner: "A *paragraph* starts to end when a *pseudo-line* is terminated by a carriage return, form feed or (UniT) code. A *pseudo-line* is a line following a line which is not terminated by a carriage return/form feed or (UniT) code. A *paragraph* actually ends when a line does not contain just a carriage return."

LocoScript 2 defines a paragraph simply as: "A *paragraph* is a group of lines separated from other groups of lines by some *vertical white space*. This *vertical white space* can be a form feed, any number of lines containing just carriage return or (UniT) codes, or a single carriage return or (UniT) code with CR extra spacing greater than zero."

Layouts

Layouts are one of LocoScript's most powerful features – they control how documents look. They are also one of the areas where we have made some big changes in LocoScript 2. This article covers the use of Layouts in some detail – including why you might use Layouts as well as how.


Layouts is a big enough subject to fill a whole issue of Script! So we've broken it down into a number of individual articles. This one gives you the basics of what Layouts do for you, plus details of how to use them. In future issues we'll be covering their advanced use and giving some examples of how to use them most effectively.

What layouts do

Put simply, Layouts describe how your text should be laid out on the page. You can use one Layout for the whole document, to give it a uniform look, or use several Layouts to add some variety. So for example, a business letter might just use one Layout, whereas a book or an article might use several, say to lay out literary quotations or scientific formulae in a different way to the narrative text.

Each Layout in LocoScript is a complete description of how text should be laid out. The simple parts of a Layout are much the same as the settings you make on a typewriter to control the look of the text. The other settings of the Layout come from the fact that a word processor is far more powerful than a typewriter!

Revalo Designs
Unit 7 Woodfield Industrial Estate
Dryden Bell West Yorks YT5 7XJ



Ms J Wilkin
Widget Products Ltd
34-36 High Road
Faldale
Lancs FY5 6TT

21st August 1987

Dear Ms Wilkin

Re: The Self-adjusting Widget Holder

Thank you for your letter of the 17th August concerning the problems we are having with your self-adjusting widget holders.

Your suggestion that we try placing the pointed end of the self-adjusting widget holders into a vat of ice cream immediately prior to inserting the widget has not proved very helpful. You must understand that we are using your products in an industrial environment which is not conducive to the keeping of ice cream. We have, however, tried your suggestion as an experiment, and it did work.

Unfortunately, the side effects were rather serious. In our case, the final position of the self-adjusting widget holders is uncertain, and some ended up upside down. By this time, the ice cream adhering to the pointed end of the self-adjusting widget holders had melted, and so started to run down to the blunt end. As you know, this is where the widget is held. The ice cream then flowed into the widget itself, causing it to fall intermittently.

I am sorry, but your solution cannot be considered to be a workable proposition for our company. We must now insist that you supply replacement self-adjusting widget holders which work at all angles, or provide us with a full refund of the money we have paid to date.

I look forward to your prompt reply.

Yours sincerely

William Greenough
Widget Sales Supervisor

Page 1

2.1. Equipment Requirements

The proposed survey runs through ground of varying degrees of difficulty. In particular, some of the high ground is particularly rocky and subject to extensive and hazardous hazards.

The new methods of construction allow us to quote for some specialist machinery to perform some very difficult tasks. You will, however, need to supply some operators with the appropriate skills for this job. The equipment and the corresponding qualifications needed are:

Equipment	Staff	Contract	Qualification
High slope grader	2 men	Industrial Hqs	2 years
Strip excavator	1 man	C & G Sec	1 year
Flying leveler	1 man	Pilot	2 years
	1 man	Machina ops	6 months

Please note that you can acquire staff with this experience, there should be no reason why we cannot provide you with necessary equipment. This will be at our regular rates less your normal Government discount, currently 8%.

Please note that all our rates are quoted exclusive of VAT and delivery charges. There is a flat rate charge for delivery of 6000 within the UK mainland.

The charges are:

Equipment	Rate	Week	Year	Insurance
High slope grader	4.23	21.00	210.00	***
Strip excavator	100.78	800.00	4000.00	**
Flying leveler	1000.00	7000.00	64,000.00	***
Delivery services	50.00	50.00	500.00	*

We can of course arrange insurance for your staff, but you may find that your standard policy covers the requirements. Please consult Fred Westhead of our contracts department for more detailed information.

In addition to the above we would like to confirm that our equipment is subject to inspection every 7 months and complete overhaul once per year. If you have the equipment on hire at the time of the year when an inspection is due, we will of course carry out the work on your own site. This is free of all charges.

Confidential

The Character Pitch Characters are displayed at this pitch (ie number of characters per inch)	CR Extra Spacing The extra movement down the page after each Carriage Return	Justify/Italic You can choose to Justify the text and whether to use Italics	Zero Whether to show zeroes with or without a slash	
The Layout Name Useful to help you remember which is which	The Line Spacing How far to advance the paper at the end of one line before starting the next	The Line Pitch The number of lines printed per inch	Decimal Whether to use a full stop or comma for decimal points	The Scale Pitch Margins and tabs are measured in characters of this pitch.
<p>Layout 1 Pi12 L51 CR+0 LP6 Justify Italic Decimal=, Zero=0 ScalePitch12 f1=Margins f3=Tabs f4=Size f5=Stock f7=Name f8=Options CANCEL/EXIT</p> <p style="text-align: center;">0 1 2 3 4 5 6 7 8</p> <p style="text-align: center;">→ ****→ ****→ ****→ **, **↓</p>				
The Left Margin How far from the left of the page to place the first character on the line	 Tabs Mark special positions to align text	A Simple Tab The first character after the tab is placed just to the right of the tab position	A Right Tab The characters after the tab are right aligned to the tab position	A Centre Tab The characters after the tab are centred about the tab position
		A Decimal Tab The characters after the tab are aligned so that any decimal point is at the tab position	The Right Margin The maximum distance from the left of the page the last character on the line can appear	

With all these items specified, LocoScript is able to work out what each character should look like and where to place it under all possible circumstances. In other words, the Layout defines all the rules that LocoScript has to obey in positioning each character.

Layouts in a document

In many cases, a document just requires all these details to be defined once – a set of rules to apply to the whole document.

The initial set of rules you work with are supplied by LocoScript. LocoScript takes these from the template for the group that you create the document in or, if there is no template, it gives you its very simple default Layout which gives you margins an inch in from either side of A4 paper.

Naturally, the details of this Layout can be changed so that your document has the margins, tabs, line spacing etc. that you want. But before we describe how this can be changed, we will look at why you might want more than just this Layout in a document.

LocoScript lets you specify further Layouts so that you can use a different set of rules to lay out the text in one or more parts of the document. The classic example of text requiring a different Layout is a table for which you need a particular set of tabs that you are unlikely to be using in the rest of the document.

In the main, the differences between the initial layout and these further Layouts will be ones of margin and tab position, rather than simple changes of Character Pitch, Line Spacing, CR Extra Spacing, etc. which can be achieved straightforwardly using (Pitch), (LSpace) and (CR) codes.

Where you want to start using this different definition, you insert a (Layout) code containing all the new Layout details. Where you want to change again to another set of rules, you insert another (Layout) code – this one containing the next set of rules you want to apply.

These (Layout) codes can be inserted by using the New Layout option in the f2 Layout menu. You position the Text cursor where you want to move over to using the new Layout; press [F2] to display the Layout menu; check that the Menu cursor is on New Layout (the top option); and press [ENTER].

This puts you into the Layout Editor so that you can set up the details of the Layout you want. As you're likely to change just a few details, LocoScript starts by giving you the details currently in force at the Text cursor. When you leave the Layout Editor and return to the document, LocoScript inserts the (Layout) code you need – complete with all the details of this new Layout definition.

At each point you want to change the Layout, you need a (Layout) code. It is perfectly possible to set up each of these by taking the New Layout route described above. But that is taking a harder route than is necessary when you are using the same few Layouts throughout the document, and especially when you also want to use these Layouts

in more than one document. Setting up the details afresh via New Layout and the Layout editor at each point you want to change the layout is a waste of effort.

Using the Stock Layouts

LocoScript 2 makes setting up identical Layouts easy by keeping a stock of standard Layouts for you to copy – the Stock Layouts. These are stored as part of the Document Set-up and when you want a Layout code with a particular set of characteristics, you create a New Layout that is a copy of the Stock Layout that has this set of characteristics.

There are two ways of copying a Stock Layout. You can either use the New Layout option as before but, when you go into the Layout Editor, you simply press [F5] to display the Stock menu, move the Menu cursor to the Stock Layout you want as the pattern and press [ENTER]. Then when you return to the document, the (Layout) code that is inserted contains all the same details as the Stock Layout you selected – ie. it is an exact copy.

The other, slicker, way of copying the details into a new (Layout) code is to press [F5] and then type LT followed by the number of the Stock Layout you want to copy. So for example if you knew that the set of details you wanted were set up in Stock Layout 2, you could insert the new (Layout) code by typing [F5]LT2.

Two of these Stock Layouts have a special use. As we explained earlier, LocoScript starts by using a supplied set of Layout rules which is stored in the Document Set-up. The Layout rules it uses are the ones in Stock Layout 1. LocoScript also needs a Layout ready to use for the documents Headers and Footers – the Pagination text: in this case it uses Stock Layout 0. Whenever you want to use the initial Layout again further down the document, you simply need to insert a new (Layout) code that's a copy of Stock Layout 1 – by typing [F5]LT1.

(LocoScript '1' used one Layout – the Base Layout – for both of these jobs, which had the effect that any changes made to the Base Layout for the benefit of the document were also made to the Header and Footer text. This had the disadvantage that if you set Line Spacing 2 so that your document was double spaced, you instantly doubled the number of lines of Header and Footer text you had, which could push the document text down the page and the Footer onto the next page – and you had to be quite alert to spot what had gone wrong!)

Using the Layout Editor

Whether you are adding a new Layout, or changing an existing one, you'll use the Layout Editor. This gives you all the tools you need to change the elements of your Layout. Page 8 shows the Layout Editor's display and the items you can change.

Typically, you'll just want to change the Margins, Tabs and Character Pitch. You can change these through the f1, f3 and f4 menus, or by using the following special shortcuts. To move a margin, position the cursor on the margin and press [F1] or [F3] to move it right or left. To insert a tab, position the cursor where you want to set the tab and press [F4] repeatedly until the correct sort of tab symbol is shown. You can also use the Space Bar to move quickly to the Right Margin and [TAB] to move to the next tab.

If you want to change the setting of Justify, Italic, Zero, Decimal or Scale Pitch, you will need to use the f8 Options menu. The Scale Pitch is the one element of the Layout you should be careful about. It is one of LocoScript 2's new features. We recommend that you set the scale pitch to 12 if you are using proportional spacing or a Character Pitch of 12; otherwise set it to 10. Once you've made sure that it is set to a suitable value, you should not need to change it again.

Finish using the Layout Editor by pressing [EXIT].

(continued)

Changing Layouts

Neither the Layouts described by the (Layout) codes nor the Stock Layouts are fixed. Both can be changed – but in different ways and with different results.

The Layouts used in the document itself can be changed by using the Change Layout option in the f2 Layout menu. The Layout this changes is the one that affects the part of the document that the Text cursor is in: no other (Layout) code either above or below it in the document nor any Stock Layout it was copied from is changed.

For example, suppose you were working on a document in which the first (Layout) code was created by copying Stock Layout 3 and that copies of this code had been used in elsewhere in the document.

To change the Layout set up by the first (Layout) code in a document, you move the Text cursor to some position below this (Layout) code but before any other (Layout) codes; press **[f2]** to bring the Layout menu onto the screen; move the Menu cursor to Change Layout; and press **[ENTER]**. You are then put into the Layout Editor, just as you are after selecting the New Layout option, and you have the same array of tools to help you make the changes you require. *(But don't change the Scale Pitch at this stage unless you are proposing to change the Scale Pitch in all the Layouts. Having Layouts with different Scale Pitches in the same document is just confusing.)*

When you return to the document, the details associated with this particular (Layout) code and only this (Layout)

code have been changed. The copies of this (Layout) code that were used in other parts of the document won't have been touched: (Layout) codes are quite independent once they've been inserted.

Note particularly that your changes *haven't* affected the Stock Layout from which the (Layout) code was copied – Stock Layout 3 in this instance. The (Layout) code was simply a copy of the Stock Layout as it was when the code was created and so shares the same name, but is otherwise quite independent.

If the Text cursor is between the beginning of the document and the first (Layout) code (if any) when you use the Change Layout option, the Layout you change is the document's initial Layout. Changing this Layout has all the same feel to it as changing a (Layout) code – the only obvious difference is that there is no (Layout) code involved.

But, in fact, there is another much more important difference. As we explained earlier, the initial Layout is Stock Layout 1, so in changing the initial Layout you actually change Stock Layout 1 itself. Similarly, when you are working on Header and Footer text on the Pagination Screen and select the Change Layout option from that screen's Layout menu, you actually change Stock Layout 0.

Of course, this also means that any change that you make to Stock Layout 0 changes how the Header and Footer text is laid out, and any change to Stock Layout 1 changes the initial Layout used in the document itself.

If your document uses the same Layout throughout, then there's no need to think of Changing the Layout as doing anything other than changing the Layout you are using – the one for the document when you are displaying the document; the one for the Header and Footer text when you are displaying the Header and Footer text.

But if you have carefully set up the Stock Layouts in your template to have particular jobs, it may well be worth remembering that changes that you make to the initial document Layout are made to Stock Layout 1 and that changes made to the Header and Footer Layout are made to Stock Layout 0.

Setting up Stock Layouts

Every document you create has ten Stock Layouts stored in its Document Set-up - Stock Layouts 0..9. If these are to be any use to you, they have to be set up as you want them. Stock Layouts set up wrongly are no more use than no Stock Layouts at all.

Starting with the Layout you want

The way to have the Layout you want in a document when it is created is to use LocoScript's Template facility.

A template is a LocoScript document that you set up to act as a model for the real documents you produce.

When you create a document, LocoScript first tries to find a Template to copy. That is a document called TEMPLATE.STD in the group you are using. (If it can't find one it looks in group 0 of the same disc, and if it still can't find one it looks on drive M where it will have placed copies of templates from your Start-of-day disc. If LocoScript can't find a Template anywhere, it invents a very basic one for you).

If you set up Template(s) with the correct Layout, then all your documents will have this Layout. So first create your Template!

To create a Template, move the group cursor to the group you want to work with (or group 0 for the Template to apply to all groups on that disc), and press C. When LocoScript gives you the document name menu, type TEMPLATE.STD and press **[ENTER]**. Now you need to make this document the model you want for all future documents, both in its Layout and in its initial text. For now, we'll assume that we don't want any text and just set up the Layout. (If you have been given any text, use the Cut facility to delete it.) That just leaves the Layout to be set up.

LocoScript will have already given you a Layout, either one from the Template it used or the default if there wasn't a Template to copy. To set up the Layout you require, simply change this Layout to have the margins, tabs etc. that you want – for details see Changing a Layout.

When you have set the Layout to what you want, you have created a Template which doesn't have any text in it, but is set up so that any text will be laid out according to the rules you've just made. Press **[EXIT]** **[ENTER]** to save the Template permanently.

From now on, all the documents you create in that group (or disc if you used the group 0 option) will use your new Layout. If later you decide that you need to change your standard layout in any way, simply Edit the TEMPLATE.STD document and change the Layout.

Typically, setting up Stock Layouts is a job to do when you are preparing a template document (TEMPLATE.STD) for a group. The Stock Layouts you set up in the template are automatically the Stock Layouts set up in the Document Set-up of any new document you create using this template. So if you think out the basic Layouts you need for the documents in this group and set these up as the Stock Layouts in the template, these will be quickly and easily to hand when you are preparing the documents themselves.

To set up the Stock Layouts, you go into Document Set-up (by selecting the Document Set-up option from the f1 Actions menu); press **[F2]** to display the Layout menu; move the Menu cursor to Change Stock Layouts; and press **[ENTER]**. LocoScript then displays a menu of the Stock Layouts.

Move the Menu cursor (and the diamond) to the Stock Layout you want to change, press **[ENTER]** and you are put into the Layout Editor so that you can set up the details you want. As you do this, remember to set the same Scale Pitch in every Layout and to give each Layout a suitable descriptive name.

We recommend setting up names for the Stock Layouts at this stage for two reasons:

– Firstly, all the menus that list the Stock Layouts offer you a list of their names. Picking out the one you want when it has an appropriate name is very much easier than picking out Layout *n*.

– Secondly, giving copies of your Stock Layouts a consistent set of names is very much harder once you have started using the template and inserting (Layout) codes in your documents.

When you press **[EXIT]** to leave the Layout Editor, you return to the menu of Stock Layouts with the cursor picking out the Stock Layout you have just changed. This makes it very easy to just work down the list of Stock Layouts changing them one by one.

We will be looking at Stock Layouts, Layout Exchange and Layout Replacement in a future issue of Script.

The effect of Layout codes on the document

To understand the real effect of (Layout) codes, it's best to understand how LocoScript works out how to lay out a document all the way from the top to the bottom.

LocoScript works by starting with the prescription for a Layout given by Stock Layout 1 and then modifying this prescription according to the word processing codes that are inserted in the document itself. Starting from the top of the document, each of these codes is taken in sequence and the changes the code makes to the Layout rules are applied, giving a new set of rules. For example, a (+Italic) code tells LocoScript to ensure that the rules now include the rule that characters should be given the Italic styling. If the rules already included Italic styling, then of course the (+Italic) code has no obvious effect. Similarly, a (-Italic) code tells LocoScript to ensure that the rules include that characters should not be given the Italic styling.

Thus moving through the document sees a succession of changes to the rules LocoScript is working to caused by word-processing codes, each change resulting in a new set of rules which is itself changed by the next code in the document.

The rules that apply at any point in the document are the result of all the rule changes up to and including the preceding word-processing code. These rules apply to all text between that code and the next code that will change it: any text inserted here will also be laid out according to these rules.

Inserting text higher up the document will move this text, possibly changing where the line breaks are but it has no effect on the rules that are used in laying out this text. Inserting a word-processing code somewhere higher up the document, however, can potentially change the rules applied to every piece of text that follows it. These changes are sorted out and the text laid out afresh as you move your working position down the document, with the final adjustments being made just before the document is saved on disc.

This way of working enables LocoScript to respond to changes made higher up the document without losing any of the positioning instructions and styling instructions you have lower down the document. A change of Character Pitch, for example, in LocoScript won't change the right margin of your text or lose the fact that you want a paragraph indented. In many other word-processing systems, it will!

The change to a different Layout is similarly a change to the current set of rules, but whereas each (Italic), (Pitch), (LSpace) etc. code just changes one rule, the (Layout) code resets the lot. The rules in use immediately after a (Layout) code match the details in the (Layout) code – with the one exception that the new margin positions and tabs only come into effect on the next line because there is no sensible way in which these can change in the middle of a line.


One effect of this is that the preceding 'history' of word-processing codes is effectively forgotten. That is why you have, for example, to insert another (+Italic) code after the (Layout) code if you want to continue using Italic characters but Italic is not defined in the Layout.

The other important effect is that you have to put in another (Layout) code when you want to return to the original set of Layout rules. Until it processes past such a code, LocoScript will continue to lay out the text according to the rules set out in the last Layout code, with any modifications made by subsequent word-processing codes.

The key to returning to the original Layout is to remember that the original Layout is defined by Stock Layout 1 – so to return to this Layout, you just need to put in a (Layout) code by typing

[F2]LT1.

Letters


In the  pull down menu there is the code (SiC) and I cannot for the life of me remember whether I read about it in the User Guide or not. What is certain is that I can find no mention of it in the index, and careful searching through likely chapters has given me no clue. Could you please put me out of my misery and tell me what it does?

Mr DF, Loughborough

The (SiC) code is used with LocoSpell. You can mark words that you do not want checked by LocoSpell with this code and then when you run LocoSpell it will just ignore the marked words.

With LocoSpell I have problems when hyphenating a word to split it over to a new line. It then is picked out as a spelling mistake. Can you fix it please?

Mr PW, Shepperton

There is a simple way round this problem, and one which will make your life easier in future! If you want to split a word at the end of a line across two lines, don't use a hyphen, use a soft hyphen. Then, LocoSpell will check the word as a whole word rather than two halves. Also, if you later change your text so that the word no longer needs to be split, LocoScript will combine the two halves on one line and remove the hyphen automatically. You type a soft hyphen by pressing the  key, then a hyphen.

I intend to add a second disc drive, most probably a 5¼" low current 40/80 track switchable, "BBC compatible" drives from Watford Electronics if reasonably possible (they are accessible and technically literate). Is there a LocoScript 2 snag?


Mr DH, Rickmansworth

Provided you make your add-on drive behave electrically the same as a standard 3" drive add-on – and make it double-sided double-track – then LocoScript 2 (and indeed LocoScript 1 and CP/M Plus) will use your disc drive correctly.

I would prefer to have the LocoScript 2 User Guide in a loose leaf binder or wire bound. Is there any chance of this?


Mr AG, Cambridge

We have had a number of requests for this, so we are thinking of producing a special edition. If we get enough requests, we'll go ahead.

We have recently purchased from you LocoMail for LocoScript 2. In general, we have no problem with converting to this, but we have not been able to discover the alternative used for the following mathematical character. The bar in LocoMail 1 which is placed in front of the right number of decimal places and is typed by the  key and full stop.


TI Ltd, Sheerness

LocoMail relies on the features of LocoScript for its underlying operation. With LocoScript 1, you had to work in a LocoScript 1 way, and with LocoScript 2 you have to work in a LocoScript 2 way.

So, to get the vertical bar character you have to use the keystrokes  and \$. This is described in the LocoScript 2 User Guide starting at page 257. (Also, we'll be including a keyboard layout chart with the next issue of Script).

I am having trouble with LocoMail for LocoScript 2. Before, I could Fill or Merge documents without any problems, but now I just get the message "Not a suitable document".

Mrs EK, Chelmsford

To Fill or Merge a document with LocoMail for LocoScript 2, it has to be a LocoScript 2 document – if it isn't you get the message you describe. So to Fill or Merge a LocoScript 1 document you have to first convert it to LocoScript 2 format. Do this by moving the cursor to the document in question and press E . You will then get the familiar "Converting ..." message. Check the document is OK and Save it. Thereafter, Fill and Merge will work.

I notice that if I set up a layout with Italics and Justification on, when I use that layout the Italic reminder is displayed in the status window at the top of the screen, but the Justify reminder is not. Why? (Both effects work successfully on the printout).

Mr CB, Warlingham

There is not enough room at the top of the screen to indicate all the possible effects without squashing the messages unduly. So we restrict the items shown to those that cannot generally be seen on the screen. Also, we feel that Justify is on/off for whole documents, whereas the fields displayed show more localised changes which cannot be shown on the screen (Subscript, Superscript, Italic etc).

With LocoScript 2 if the SHIFT key is held down and two keys pressed simultaneously (I am an inaccurate two fingered typist so this happens frequently) then the FIND/EXCHANGE menu appears. It happens a lot with the K and L keys.

Mr JR, Kenilworth

The keys on the keyboard are connected together by a grid of wires (called the Scan Matrix). When a key is pressed down, it connects some of these wires together. Consequently, electrical signals sent down one wire are detected on another. Because of the grid arrangement, this works rather like a map reference and allows the task of deducing which key is pressed to be left to a little dedicated microprocessor.

This keeps the keyboard simple to manufacture, but does have one problem. Think of a square made by two crossing pairs of parallel lines. You know if you pass a current down one and pick it up on another, that the corner where these meet must be touching. So, if that were a key on the PCW's keyboard you could deduce that it was pressed down, similarly for 2 keys. But, if 3 keys at the corners are pressed simultaneously, you cannot detect any difference from all four being pressed! This "ghost" fourth key is precisely the effect you have noticed.

This is why we never use three key combinations except with ALT – because in the case of ALT all the "ghost" keys are in an unused part of the scan matrix.

You will in fact find exactly the same effect with CP/M and with the original LocoScript.

One of the documents I converted from LocoScript 1 has now become inaccessible. When I try to edit it, I get the message "Not a suitable document". Fortunately, it is not a very important document, but it would be extremely awkward – not to say costly – if this happened very often. Have you had any other reports of this problem, or is it a one-off? And is there any way of getting into this document?

Mr MC, Southampton

I'm afraid that the solution to your problem is rather mundane. You are mistaken when you say that you converted this document. The document is in fact a LocoScript 1 Saved Block (and not a document at all). LocoScript 2 cannot process these – use LocoScript 1 to insert it into a document and then convert the document.

Letters

As a user of the French version of LocoScript 1 (ie AZERTY keyboard, rubrics in French) I am wondering when and from where the French version of LocoScript 2 (disc and manual) will be available.

Dr GE, Norwich

We are now working on foreign versions of LocoScript 2 (and LocoSpell and LocoMail) and will be making these available as and when they are ready.

We have changed the way LocoScript 2 works so that you don't have to work with the version of LocoScript that matches your keyboard. So if you have a French keyboard, you can equally well use an English version of the software (ie. with English messages), or a French version or... See the News pages of this issue of Script for further details.

However, we should point out that it will not be possible to use foreign LocoSpells with an English LocoScript.

I would find it very useful to be able to make an ASCII-version of text originally typed in Russian and then of course to be able to transfer it back into LocoScript. Is this in fact possible?

Also, in his preview of LocoScript 2, Ben Taylor (8000Plus, May 1987) states that in LocoScript 2 "there is scope for you to define up to 10 characters of your own design". I have not found this in the manual so far, and should be grateful if you would explain to me how this can be done.

Mr LB, Halesworth

At the moment, there is no way of making an ASCII version of Russian text that retains the Cyrillic characters. The problem is that making an ASCII version isn't just a matter of stripping out layout and styling codes - it also involves making it compatible with other software, and at present, as far as we are aware, there is no recognised standard for Cyrillic text. (Actually it isn't clear what you'd use the ASCII for anyway, since virtually no other software can cope with the Cyrillic alphabet).

The facility to create user defined characters will in fact be to create 16 special characters. It is not in the manual because it requires an additional piece of software with which you will be able to design your own characters (for the screen, high resolution printing and draft printing). This additional software is now available (See the News page).

My problem is that after making a back-up copy of the master disc, using DiscKit, the copy fails to run, also using the disc copying system within LocoScript 2 the back-up copy doesn't work. After it has almost loaded (using DiscKit to copy it), it prints:

```
ERROR in: loading SCRIPT.JOY
Checksum is wrong
```

Using the LocoScript 2 method the program only runs as far as clearing the screen of the black and green lines.

I am using an upgraded PCW8256.
NC, Bristol.

I have copied the disk (both sides) on to two different disks, formatted with CP/M and have had similar results each time. Although with the original LocoScript disk I reach the disk manager stage without trouble, this does not happen with the copies.

With the copies I cannot reach the disk manager stage but as soon as the disk is inserted I stick at:

```
Error with file: SCRIPT .JOY
File checksum error
Cancel operation
```

But if I press the 'cancel' key it has no effect.

I hope I do not have the trouble I had with the PCW8256 in November 1986 when I had to change the machine twice because of hardware faults.

Mr JS, London

Your problems are symptoms of faulty memory.

When discs are copied by LocoScript 2, they are read and their contents stored in the PCW's memory to be written back to the new disc. If the memory is faulty, the copy can be corrupted. LocoScript 2 uses different parts of the memory when copying discs to those used by DiscKit, and so it is actually possible for the DiscKit copy to succeed but the LocoScript copy to fail.

[Editor's Note: Locomotive have agreed with Amstrad to distribute a special disc which tests the memory. In both these cases, and in several others, we have lent out copies of this disc, and found the memory to be at fault. - as the following letters show. Unfortunately, they also show that some service engineers do not thoroughly understand the equipment they are servicing!]

Thank you for the copy of the PCW self-test program.

This did show that the memory in my upgraded PCW had a problem and I was able to get my PCW to my supplier at the beginning of the week. He was able to replace the memory board and now I can successfully make copies using f2.

Rev JBM, Downpatrick

Please find enclosed the test disc which you very kindly sent me in response to the problems I was having with LocoScript 2. You will be pleased to hear that these problems have now been solved and that eventually, after two visits from the service engineer, the cause was found to be in the CPU.

Certainly I would not have convinced the engineer that there was a hardware problem without the disc and am very grateful for your help and service.

Mr RN, Caterham

I had not really used the disc copying facility previously to make my working discs, and instead used the DiscKit program with CP/M. This time I decided to use the LocoScript version and found that it told me that discs were in the drives when the drives in fact were empty! This is not a great problem as if you decide to proceed, as it encourages you to do, it ultimately decides that there are no discs to work from and says so.

Mr RA, Broadstairs

*I think you have misunderstood the "Discs are in the drives" message. Like other options picked out by arrows or diamonds, this message is the description of the command you want LocoScript to carry out when you press **ENTER**: it is not LocoScript telling you that this is the state of affairs.*

In fact, LocoScript cannot tell whether there are discs in the drives without starting the disc motor, which is the last thing you want to happen while you are changing discs. When it really matters, LocoScript 2 checks that the discs are there and that they are the right type and so on, as you have observed.

Letters

Can you tell me please whether the LocoSpell word-counter counts every word? Even two-letter words? Prospell doesn't and so is totally useless for that job, though it is pretty fair as a proof reader, once you cotton on that it corrects only in the B drive.

Mr AE, Belsize Park

LocoSpell defines a word as "black stuff surrounded by white stuff", and counts all items fitting this description. This means that it will count one letter and two letter words, and indeed will count a hyphenated word as a single word. Only words which are alphabetic are checked (with hyphenated words being checked as two halves). Unfortunately, all definitions of a word have their faults, and the problem with this definition is that isolated punctuation (a "." or "-" say) surrounded by spaces will count as a word.

LocoSpell works as an integral part of LocoScript's editor and so any changes you make (or LocoSpell makes) are automatically included in your finished document.

You may recall that I phoned you in panic about 10 days ago when I could not get LocoScript 2 to work with a Panasonic printer. The problem has been solved, so in case others are as ignorant as I was, it seemed a good idea to tell you what had gone wrong.

The solution lay in getting the correct interface. Originally I had bought an Amstrad 8256/512 Centronics Printer Adapter made by Computer Related Services Ltd. It needs a short program to make it run, and worked with CP/M software. On changing the interface to the Amstrad Centronics Parallel/RS232C Serial interface, all the problems were solved! I had run all the programs correctly, but as far as the computer was concerned, there was no printer at all, hence the funny messages. Hope this helps someone else, as the dealer had sold me the first interface in good faith, and in fact took it back when we realised it wasn't suitable.

Dr CE, Truro

I am experiencing difficulty in getting LocoScript 2 to drive my daisy wheel printer.

I have a Silver Reed EXP400 printer and an SCA interface. I have worked through Session 20 of the manual several times but there is a total lack of response from my printer, although the Printer Supplement lists the EXP400 as being suitable for support by the D630.PRI and INSTALL.DRV files which I have copied into Group 0 as instructed.

If I go through the printing sequence and then press **[PTR]** I get the message "waiting for paper". I have tried pressing **[P]** to "resume printing" but although I get the message "printer ready", when I press the **[EXT]** key I still get no response. Pressing **[E]** in the printer control state shows that LocoScript wants to use the EXP400 printer.

You will probably say that this is a matter for the printer manufacturers but since I cannot be the only person who wishes to use LocoScript 2 with the EXP400 I think you should know of the problem.

Incidentally, I experienced a similar lack of response attempting to print ASCII files with CP/M version J14CPM3, but got round the problem by using version M14CPM3 which I also had to hand, so I know the printer and interface are not faulty.

Mr SC, Knowle

Your problem is not with LocoScript, nor is it with your printer. The problem is the SCA interface. We believe this interface is non-standard and is not compatible with the Amstrad CPS8256. LocoScript 2 only supports interfaces which are fully compatible with the Amstrad product. We assume that this is why you could not use the interface with CP/M (and assume that M14CPM3 is some hacked version of Amstrad's software which will drive this interface).

You will need to replace your interface with a "standard" interface. Then, you will be able to drive the EXP400. Note that we have now produced a new Printer File for printers like the EXP range. Although your printer will work with the standard D630.PRI file, you will get better results by getting the new version.

[Editor's note – see the News Pages for details of the new Printer files]

The capacity of Drive M is proving a little small for moving the very large documents which I normally handle, and I should be pleased to receive some details of the PCW8256 Memory Upgrade (in particular how much more memory does it give and how much time does fitting take).

Mr TT, Winchmore Hill

The Memory Upgrade adds 256K to Drive M, and as well allowing you to move large documents, it will let you copy single sided discs in one go and (if you use LocoSpell) it will let you use the large dictionary from the Memory disc.

It should take about 40 minutes to fit the upgrade. It entails plugging some extra chips into sockets on the PCW's main board and changing some wire links. Unless you have a very old PCW, no soldering is required to change the links.

Will LocoScript 2 arrange entries in the order of the Esperanto alphabet? – or have you the software that will do this? – as I need this for various purposes.

Mr ER, Tal-y-Bont

The only place where LocoScript arranges items in alphabetical order is the Disc Manager Screen. The names here are restricted to ASCII characters to be compatible with the CP/M file names.

LocoSpell presents the user with an alphabetical list of words in its Consult Dictionary menu. We have no plans to produce an Esperanto spelling checker.

The Letters column will be a regular part of Script. We will take letters of general interest from Locomotive's Customer Support mailbag, and of course from letters to the editor of Script.

Printers

Installing Printers

LocoScript 2 comes ready to print on the PCW matrix printer. However it can print on a wide range of additional printers. Some of these are already supported as standard, but many more will become available over the coming months.

The standard printers supported are those which are compatible with the Diablo 630 daisy wheel printer or with the Epson FX80 matrix printer. If your printer is compatible with one of these it is a relatively simple matter to modify LocoScript to use it. The document PRINTER.SUP on your LocoScript 2 master disc contains the Printer Supplement to the User Guide, and gives details of using the supplied extra printer software.

The problem is that whilst all "compatible" printers are equal, some are more equal than others. For example, we now know of several supposedly Diablo 630 printers which are not 630 compatible

at all! Also, even if a printer is compatible with the Diablo 630 or Epson FX80, it might actually print different characters.

Whilst many modern matrix printers are broadly compatible with the FX80, there are a number of different daisy wheel printers. Also, some of the printers already supported can be driven in a better way. We are producing new printer files for these printers, and so you will be able to use LocoScript with Qume printers, with Amstrad DMP3000 printers using their IBM character set, and so on.

Even if your printer is not one of those which we support as standard, all is not lost. We are producing a full printer support kit, to make it possible for you to produce your own printer files. This is intended chiefly for printer dealers. It will not be cheap (as we expect to have to give more support to its users) and will require a thorough knowledge of the printer being installed. We hope to have more details in the next issue of Script.

Daisy wheel printers

For each daisy wheel printer you need one or two printer files – one to control the printer, and one defining the print wheel you have fitted if it is different to the one expected as standard.

All LocoScript can tell a daisy wheel printer to do is (in essence) print the character on petal number *n*. The print wheel files tell LocoScript which character is on which petal, and so enable it to print the correct characters. Thus, if you haven't got the correct print wheel file you may well print gibberish.

There are many different print wheels, and whilst a fair number of these have the same arrangement of characters on their "petals", there is still a wide range of distinct wheels. It is not practical for Locomotive to support anything other than the most popular of the wheels. So we have produced a special program which runs under CP/M that you can use to create a print wheel file for yourself. All you need is the program, its user instructions and the printwheel and you should have no trouble getting the right characters printed.

Single sheet paper and LocoScript 2

If you use "single sheet" paper with LocoScript 2 then you will find that LocoScript 2 understands that you have to load fresh paper at the end of each page and will help you out as far as it can.

While it is printing LocoScript 2 keeps track of its position on the paper. When LocoScript knows the sheet of paper is finished it stops printing and waits for you to insert the next sheet. Only when LocoScript knows that the next sheet is available will printing continue. By keeping track of what is going on in this way LocoScript 2 avoids printing onto the platen or otherwise losing part of your document.

If you are using the built-in matrix printer this is all nice and simple. LocoScript will detect you pulling on the bail bar to load the next sheet. LocoScript then automatically enters Printer Control state and, because it knows you are loading paper, it automatically cancels its "Waiting for Paper" state. Once you are sure that the paper is straight, you type **EXIT** and LocoScript continues printing.

All of this is no different in principle with external printers. Unfortunately it is not possible for LocoScript to know when you have put the next sheet of paper in. Not all printers can sense paper; there is no standard for signalling paper or bail bar state and the CPS8256 interface does not implement the Centronics (Parallel) "PAPER OUT" signal anyway!

Thus, when an external printer is being used with "single sheet" paper the following sequence happens.

1. LocoScript 2 keeps track of its position and stops printing at the end of the page.
2. LocoScript 2 enters "Waiting for Paper" state. The top line changes to display `Paper Please`. Note that if your printer stores text in an internal buffer before printing it, this stage may be reached before the entire page has been printed.
3. Operator loads new paper.
4. Operator types **PTR** and enters Printer Control state.
5. Operator types **F1** and then **ENTER** to "Resume Printing"
6. Operator types **EXIT** to leave Printer Control state.
7. LocoScript 2 will now continue printing.

Note that this is just like the built-in matrix printer except that actions 4 and 5 must be done "by hand" because LocoScript cannot detect you performing action 3.

If your printer is capable of detecting whether it has paper in it and automatically prevents itself from printing on its platen then we recommend that you set LocoScript up to expect **Continuous paper**. This shifts the task of detecting what's happening to the paper from LocoScript 2 to the printer itself.

If your printer doesn't have this facility then you will have to use the procedure outlined above.

Note that for LocoScript 2.03 and earlier, the message `Paper Please` is not displayed; instead you will see the word `Printing`.

PostScript

We were late (for which we apologise), but we did finally finish LocoScript 2 and “get it out the door”. Well, I say get “it” out; by the time we were ready to dispatch there were over 5000 anxious customers waiting so we had more than 5000 “it”s to dispatch. Also, by that time we had a fair sized “priority box” of people with the best excuses to jump the queue.

None the less, we tried hard. LocoScript 2 came into our offices at 2.30pm on Tuesday 23rd June, and by 6pm that day we had packed and posted the first 1,200 orders. By Thursday we’d cleared the lot! We even caused chaos at the Post Office. On the second day, they sent along their usual Transit van for the post, but you can’t fit 2000+ copies of LocoScript into a Transit! So they ended up sending a ten-ton lorry.

During our planning for sending out LocoScript one thing we had underestimated was how mobile you all are! As the date of June 23rd came closer we had one member of staff occupied almost full time in keeping our orders up to date. Many of you moved house, a fair number got married and, as it turned out, some simply disappeared. Only a few of you had stopped your cheques.

We’re glad to say that most of you like LocoScript 2 – not everyone, but the enthusiasts far outnumber the grumblers. We’re now trying to ensure that we keep you happy and that means running our Customer Service Department as efficiently as possible. In order to give you the best product at our low

price, one thing we *don’t* offer is telephone support. That’s not because we don’t want to help you, but because the phone is very inefficient when it comes to sorting out problems. If our Customer Support Staff don’t know the answer to a phone query, they can’t always find a suitable “expert” at that particular time. With a written question, time can be scheduled to ensure that even the most difficult of questions are answered promptly and correctly.

Every support letter we get is logged when it is received and tracked as it is passed round to the appropriate person for reply. We aim to turn round all letters within a week, and most within a day or two. If you don’t get a reply within that time, it probably means we never received your letter. If you keep a note of the date you wrote to us, let us know this date when contacting us to find out why we haven’t replied.

What is important for us is that you supply good details of what your problem is. The User Guide discusses the information we need to know (on the page just before the Contents List). If you are having trouble with a particular document, one thing which would be more help than anything else is if you could send us a copy on disc of that document. And here, we ask you to break the golden rule. Copy the troublesome document onto your LocoScript 2 Master Disc and return it to us. When we reply, we’ll make sure you get a master disc returned, so you won’t be without a master disc – but it won’t always be the same master disc.

In future issues:

What we cover will in part depend on what you want. Of course, we’ll be continuing with the series on Layouts - moving on to replacing and exchanging Layouts - and we will be continuing the series on LocoMail. We’ll also be looking in depth at some of the other areas where we answer a lot of questions, and once again, publishing some of the more interesting letters we receive.

So, do write to the editor of Script and say what other areas you’d like to see covered.