

**8000
PLUS**

TIPOFFS

The collection: issues 1-6

PART I

TIPOFFS

The full collection taken from the first
six issues of 8000 Plus

For detailed contents refer to index on page 44

TIP-OFFS

The pages which give you the lowdown on LocoScript, CP/M, life, the universe and avoiding the PCW hair-tear syndrome.

Screen printout

It's often handy to be print out an exact copy of the screen on the printer. If you have lots of discs, maybe you could keep with each one a copy of the LocoScript main menu screen showing all the document names so that you know what's on it.

To get a "screen dump" like this at any time, whether you're in LocoScript, CP/M or an application program, just press [EXTRA] + [PTR]. **WARNING!!** If you are still using LocoScript version 1 (supplied with early PCW 8256's), then doing a screen dump while editing a file in LocoScript will work fine, but will crash the machine afterwards. Send off your version 1 to Amstrad and get your free LocoScript update now! (The version number is displayed on the initial title screen.)

Control and Escape

Some of the key labels on the PCW keyboard are not conventional. If you buy some software, you might be frustrated by not being able to find the keys it refers to as "CONTROL" and "ESCAPE" or ("CTRL" and "ESC"). On the PCW, you will find that the "CONTROL" key is labelled "ALT", and the "ESCAPE" key is usually labelled "EXIT", or sometimes "STOP" – it varies between software packages.

The 8512 second disc

If you've invested in a PCW8512, or bought a second disc drive for the 8256, then you might get caught out if you haven't realised that there are crucial differences between the two disc drives on the PCW.

Discs on the single density drive (drive A) have two sides; each side has to be formatted separately, and can be write protected separately. On the double density drive (drive B), which has twin disc heads, the PCW uses both sides of the disc at once. Formatting the disc formats both sides, and the disc can only be used one way around in the drive (the way that it was when you formatted it). So don't try to format side 2 of a double density disc, thinking you can store more files there, because you will only erase everything that is on side 1!

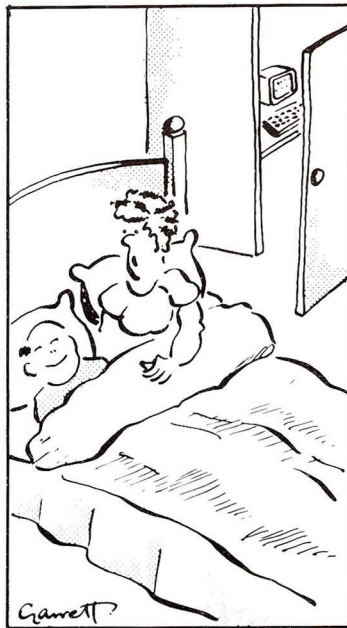
Copying to printer

Apart from simply printing out a copy of a file that you have on disc, the printer can also be made to echo everything that is written to the screen while you're in CP/M. To do this, type [ALT] + P at any time; the PCW will beep once, and then the printer will reproduce all subsequent text that goes onto the screen. Typing [ALT] + P again turns off the printer (but won't beep at you).

This can be really useful for example, you can store with each disc

a paper copy of its directory. Type DIR [ALT] + P [RETURN], and the directory will appear on the printer. When the printer stops, type [ALT] + P again. Or, you can print a copy of the output from a program to use as instructions for someone else; type [ALT] + P before you run it, and again after it has finished.

This only works for simple text printing – the printer won't copy any fancy screen graphics. But don't forget you can get a full screen dump by pressing [EXTRA] + [PTR].



"JUST WHO IS THIS JOYCE YOU KEEP MUTTERING ABOUT?"

Caps lock key

The SHIFT LOCK key on the PCW does not work quite in the conventional way. Most computers have a CAPS LOCK key which puts all letters into upper case, but leaves the number keys unshifted, unlike SHIFT LOCK which alters both.

KEY CONVENTIONS

The TipOffs in this booklet are taken from the first six months' issues of 8000 Plus. Over that time the ways we have indicated how you should type text have changed.

- 1 Text you are supposed to type in at the keyboard is either in **BOLD** or in a **SPECIAL TYPEFACE**.
- 2 When referring to a single key, like the 'Return' or 'Extra' key, it is put in square brackets. So when you read DIR [RETURN] you should type the letters D, I and R, followed by the Return key. The special 'plus' and 'minus' keys either side of the space bar are referred to either as [+]/[-], or as ⊕/⊖.
- 3 Some command require the use of the [ALT] or [EXTRA] keys in conjunction with another key. When you are told to type [ALT]+P (or sometimes just [ALT]P), hold down the [ALT] key and press P. Use of the [SHIFT] key is normally obvious, so we would refer to the [f2] key, rather than [SHIFT]+[f1].

CAPS LOCK is normally a good deal more useful – it would allow you to type PCW 8256 instead of PCW **%!

But all is not lost; buried in the depths of the PCW manual there is a way of getting CAPS LOCK: just press [ALT] + [ENTER]. The same combination again releases CAPS LOCK mode.

Using phrases

LocoScript has a quick and handy way to save you typing out regularly used phrases again and again – you can save up to 26 short phrases so that at a single keystroke the words are automatically typed for you.

To set this up, first type the phrase normally into a document as you are editing. Place the cursor at the start of the phrase, and press [COPY]. Move the cursor to the end of the phrase, press [CUT], and choose a letter (A to Z) to store the phrase by. To insert the phrase later on in the document, type [PASTE] followed by the phrase's reference letter.

You can go on using the same set of defined phrases in any document you edit during a given session with the PCW. If you want to turn off the machine but still use the same phrases when you next edit a document, then you will have to save them. To do this, select the "Save all phrases" option from the "Blocks" menu [F8] while editing any document. This creates a file PHRASES.STD in the first group on the M: disc. Now replace the PHRASES.STD file on your start-of-day disc with this new version using the Copy command [F3], and these new phrases will be available whenever you start up LocoScript using that disc.

The Unit key

There is a little mentioned facility in LocoScript, hinted at by the mysterious presence of a key marked "UNIT". The "Set" menu also has an unit option. Units are a way of storing place markers in LocoScript text, so you can easily move around.

Suppose you are writing an instruction manual, with a summary section. Every time you write an instruction, you want to update the summary section too. At the place where you are inserting an instruction section, set a unit mark (type [+]UT), and also set one in the summary section. Now pressing [UNIT] while in the instruction section will take you straight to the summary section, and then [ALT] + [UNIT] takes you straight back. You can set any number of unit marks – each press of [UNIT] takes you to the next (or last) mark.

Making up a NewWord boot disc

NewWord is a word processing package that is becoming increasingly popular with PCW owners. Although it comes with a very meaty manual, that doesn't infact address Amstrad specifics at all, and the additional sheet and READ.ME file that NewStar distribute are only of limited help. One of the most useful things to do is to make up a disk so that NewWord will automatically start when CP/M is booted up.

There's an easy bit and a hard bit to this. First the easy parts:

- ▶ Find a blank side of a single density floppy disc, and using PIP copy from your CP/M master disk onto your blank disc the files J14CPM3.EMS, PIP.COM, SUBMIT.COM, SETKEYS.COM, and KEYS.WP.

- ▶ Copy from the NewWord delivery disc the files NW.COM, NW.OVR and NWMSG.SOV.

Now for the hard bit. There isn't enough room to store the final file that NewWord requires, NWPRINT.OVR, without first "installing it" for the PCW. This allows for most of the file to be discarded since some parts are irrelevant to the Amstrad. Here's what you do.

- ▶ Copy from the NewWord delivery disc to the M drive the files NWPRMAKE.COM and NWPRINT.OVR.

- ▶ Type **M:** to set the default drive to M:, and type **NWPRMAKE NWPRINT.OVR**.

- ▶ You will see a host of names of printers on the screen and although none of the options mention Amstrad or PCW, the right one is number 7 (unless you are using a different printer). Press **7**

- ▶ **[RETURN]**
- ▶ Now copy NWPRINT.OVR from M: onto your boot disc.

- ▶ Finally, you have to create a file

to tell CP/M what to do when it starts up. Put your boot disc in the A: drive and run NewWord from it by typing **NW**. As described in the NewWord delivery sheet, create a non-document file (type **N**) called PROFILE.SUB with the following lines in it:

```
SETKEYS.WP
PIP
<M: = NW.COM
<M: = NW.OVR
<M: = NWMSG.SOV
<M: = NWPRINT.OVR
<
M:
NW
<LA
```

Make sure that the last line ends with a **[RETURN]**, and finish editing. Note the trick in the last line: because of the way SUBMIT files work, **LA** is passed through to NewWord as the first command to run, so it automatically changes the logged drive to A:, where your data files are, (if you have an 8512 you might prefer to make the B: drive the default drive by substituting **<LB**).

You can take this further and add other initialisation commands of your own at the end of PROFILE.SUB, each line preceded by the **<** symbol. However, if you're doing a lot of initialisation commands you ought to try and customise your NewWord permanently using the NWINSTALL program - more of this below, and in the NewWord manual.

Now with your boot disc in the A drive, if you reset the PCW with **[SHIFT] + [EXTRA] + [EXIT]**, NewWord will automatically start up and run from the M drive, but use A as the drive on which files are to be saved.

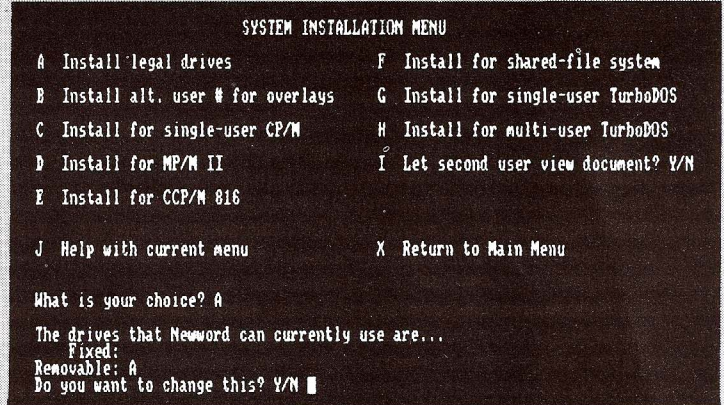
You will notice that when running NewWord as it is delivered that you cannot change the logged disc drive to M:. This is because it comes

preinstalled without M declared as a legal drive (the reason being to prevent you holding data files on M). If you want to override this, it can be done fairly easily by "re-installing" the program.

To do this first of all ensure that you have the NewWord files on your M disc and that M is your default drive, (i.e. the CP/M prompt is "M>"). Now put your NewWord master disc in the A drive and type **A:NWINSTALL [RETURN]**. The name of the file you need to install is **NW.COM**, and the file to hold the

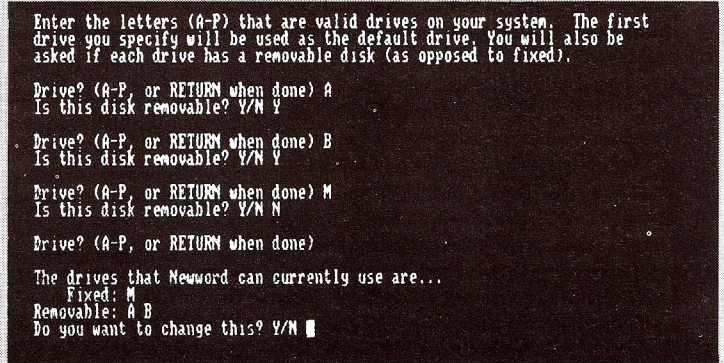
new version to is also **NW.COM**.

At the first options menu you see, type C for "Computer related items", and then A at the next menu for "install legal drives". Now follow the screen prompts to declare all three of A, B and M as legal drives - A and B are "removable", and M is "fixed". Exit back through all the menus, and test out your change by running NewWord. When you are sure everything is OK, you can copy the amended NW.COM file to your boot disc, and that will make your changes permanent.



▲ To allow NewWord to recognise drive M, select option A from the System Installation Menu.

▼ That produces this screen. Using our method, the order in which you specify drives is not important.



Mysterious happenings inside PIP

Most of the utilities that come free with CP/M such as PIP, SUBMIT and so on are not renowned for being overly helpful when things go wrong.

Suppose you want to use PIP to transfer a single file from drive A: to drive B:. As you well know, you type **PIP B: = A:MYFILE.DOC** cross your fingers and hope everything goes smoothly. But if it doesn't you'll probably never work out what has gone wrong by reading the error messages that CP/M produces. Here are some of them explained in human being-style terms:

ERROR:DISK WRITE NO DAT BLOCK-B:MYFILE. \$\$\$ means that the destination disc B: is full, and doesn't have enough room for the new file.

ERROR:MAKE FILE R/O DISK - B:MYFILE. \$\$\$ means that the write protect tabs are activated on the B: disc. Take the disc out, unprotect it and try again.

ERROR:OPEN FILE NONRECOVERABLE - B:MYFILE.DOC
ERROR:OPEN FILE INVALID DISK SELECT - B:MYFILE.DOC
 both these mean that either you haven't got a disc in the disc drive (idiot!), or the disc is unformatted or has been corrupted. You can format the disc using DISCKIT, but this will erase anything already on it.

Problems with DISCKIT

Copying discs with the aid of DISCKIT is kids' stuff, so (naturally) it can be rather boring!

Imagine that you are running a large application from a full RAM disc (drive M) with your priceless new data safely on one of the floppy discs, A or B. The time has come to make a backup copy, so you leave the application in the approved manner, write-protect your data disc and run DISCKIT from a CP/M system disc containing DISCKIT.COM.

Following the instructions on the screen, everything goes as you expect until you eventually press **Y** to go ahead with the copy. Then a message appears on the screen "Copying in 20 parts" (if you are copying a double density disc in B) - yes, two-zero parts, which is a gentle way of breaking the news to you that you are in for FORTY disc changes.

This is caused by a "feature" in

DISCKIT that it uses the M drive as temporary storage while copying discs. Therefore, if there's not much room on your M drive the copying process will be in several parts. So for best effect, M must be empty before copying discs, or you're in for a lot of disc swapping.

If you find yourself faced with the "copying in 20 parts" message, you can break out of DISCKIT; at the stage that it says "press Y to continue", remove all discs from the drives, and press **Y**. Then an error message appears saying that the drives are empty and asking you to "R-etry or C-ancel?". Press **C** for cancel, and the process is aborted. Now you can exit from DISCKIT normally, clear out your M drive and start over. Now, with a clear RAM disc, you will see, "Copying in 2 parts" - so you might reach the pub before closing time after all! ▶

A different way of using letter templates

You will by now have discovered that you need several different headings for writing paper, depending on whether you are writing to your mum, your bank manager, or whoever.

You have probably set up various TEMPLATES.STD and alternative layouts in different LocoScript groups to suit all your needs, but there are other ways of storing templates that are more economical on space and possibly easier to use. You could set up headlines for letters using the "Edit Header" option from the "Modes" menu [F7] – but don't – it wastes paper and a large blank area at the top of following pages looks wrong. Try this technique first:

Firstly, create a letter exactly as you want it, headings and all. Now

while editing it press [ALT] + [PAGE] (to take the cursor to the top of the page), [COPY], then [EOL] repeatedly until you have highlighted the area of the letterhead that you want to re-use in other letters.

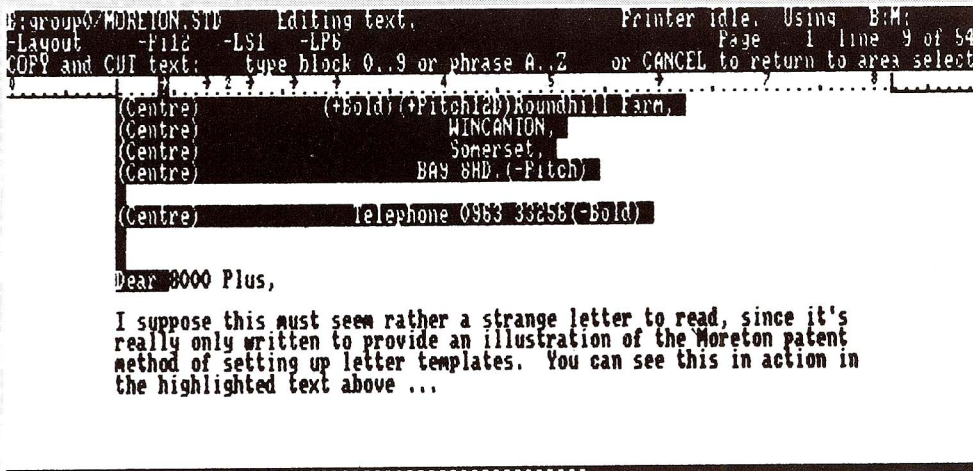
Press [COPY] and then a letter from A to Z by which you wish to store the template like "F" for "Family" letters. What has happened is that your whole-heading has been saved as a LocoScript phrase, and to re-use it in other letters you just type [PASTE] followed by the letter it is stored by (F, as we used) at the top of the new letter.

When you create a letter, you are first faced with contents of the document TEMPLATES.STD of the current group. If you are using this

phrase template method, you will need to totally clear your template file to give yourself a clear work page at the start: to do this, edit the template file normally, type [CUT] at the top, move to the bottom, [CUT] again to delete it all, and then save it normally.

A final note: as explained in the tip titled "Using Phrases", you'll have to store your new PHRASES.STD document on your startup disc so that you can still use it next time you start LocoScript from cold.

(Thanks to JOHN MORETON for both this tip and those entitled 'Stopping a Printout' and 'Problems with Diskit'.)



▲ How the screen should look when you CUT the letter heading for storing as a Phrase – the highlighted area can then be pasted instantly into future letters.

Command line editing

The PCW has a way to allow you to edit the previous command line that you typed in CP/M. This is very useful if you mistyped a long command since it saves a lot of retyping.

At the CP/M "A>" command prompt, and even in some programs like PIP, if you press the [PASTE] key, the last line you typed will appear with the cursor at the end of it. Then you can use the cursor left and right keys and the two delete keys to edit the line. Press [RETURN] to run the amended command when the line looks correct, and if it's still wrong you can use [PASTE] again to repeat the process.

Stopping a printout

Have you ever been in the position when you've finished a LocoScript edit with the "Save and Print" option, only to realise too late that the document is still wrong, and you have to sit and wait while 37 useless pages churn out?

In case you haven't mastered the ins and out of the printer, then there is a way to cancel printing very fast. From anywhere in LocoScript, just press [PTR], then [F7] (for "Reset"), then [ENTER] to confirm. Finally, of course, [EXIT] to leave the printer menu.

Foot trouble

Although the current version of LocoScript (version 1.2) has fixed many of the old bugs in using headers and footers, there are still many frustrating pitfalls for the unwary. Here's a word of advice if you are wondering why you can't get footers to appear properly.

If you choose the option of having the header and footer on page 1 different to all the other pages, then the footer will not in fact appear unless there is a second page to print. So, to get a footer at the bottom of a single page document, either set the header/footer pagination to "All pages the same", or finish your document with [ALT] + [RETURN] to force a second page into existence.

Quickfire BASIC

The suggested way of loading or running BASIC programs is to type BASIC at the CP/M A> prompt, wait for the system to get ready and say 'Ok', then to type RUN "WRDCOUNT", or whatever your program is called.

There is a quick way of doing both these in one. Provided your program is on the same disc as BASIC.COM, just type CP/M BASIC WRDCOUNT (note there are no quote marks), BASIC will start and run the program WRDCOUNT.BAS automatically. When the program finishes, you will be returned to CP/M instead of the BASIC prompt.

More on multicolumn LocoScript

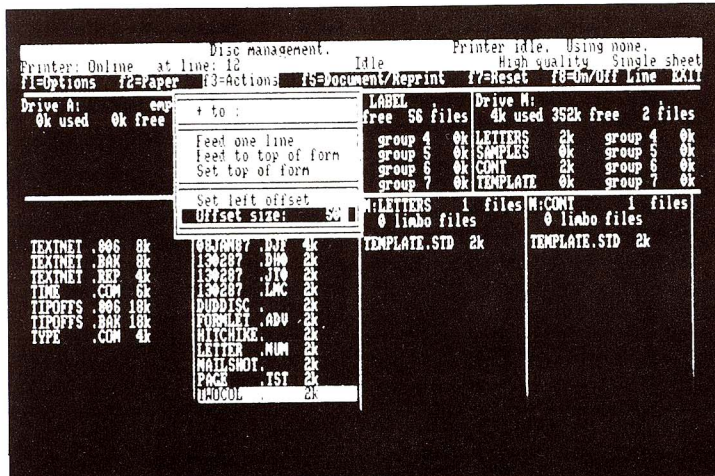
A tip printed on page 15 explains how to get two-column print from LocoScript, by setting up half-

width columns and changing Layouts every page.

This works, but makes editing

the file very tedious since you have to move all the layouts around if you insert or delete any lines – also it royally confuses the AnsibleIndex program. An alternative way is to use a single layout for the whole document, again half a page wide (eg. with a left margin at 0 and a right margin at 36).

To get two column print, you just change the 'left offset' of the page, using the [PTR] and [F3] keys, prior to printing a page. An offset of 10 prints the left half of the page, and an offset of 50 prints the right half on the same sheet. Admittedly you have to print every page separately with the 'Print some pages' option of the print menu, but editing the document is quite straightforward – no messing with continual layout changes. Eric Fenster, Oxford



▲ Setting the offset for a right-hand column

Overprinting

One feature that LocoScript lacks is a backspace facility to allow you to create composite characters by overprinting on the printer. However, there is a way round this, albeit longwinded.

The solution is to put the characters you want to superimpose on separate lines one below the other on the screen while editing your document, and set the line spacing to zero between those lines. Here's an example: suppose you want to show a word having been crossed out with hyphens.

- (1) Make sure you are not using proportional spacing – change to 12 pitch if you are.
- (2) Type the basic line of text and press [Return].
- (3) Type the strike out hyphens on the next line down, in the same columns as the characters that they are to cross out in the line above.
- (4) At the start of your basic text line, set the line spacing to 0.
- (5) At the start of the hyphen line, set the line spacing back to 1 (or whatever it was originally).
- (6) Carry on using LocoScript as normal

```

PRINTERS/OWAPRINT Editing text Printer idle Using M
-Layout +F1P -L1 Page 1 Line 11 of 54
F1=Show F2=Layout F3=Emphasis F4=Style F5=Lines F6=Pages F7=Mode F8=Blocks EXIT
-----
The following line shows a correction:
( )Space( )The Amstrad PCW is a truly awful awesome machine!
( )Space( )
and now it's back to normal ...
    
```

- How the LocoScript screen should look (showing both "Codes" and "Effectors").

The following line shows a correction:

The Amstrad PCW is a truly ~~awful~~ awesome machine and now it's back to normal ...

- The printed result.

Two jobs at once

The PCW has several pairs of hands – you can print a file from LocoScript at the same time as editing another. But you can go further than this; if you're editing a long file, you might want to print another file without having to save your current edit and restart all over again.

While editing your file, press the [F7] key and select the "Disc management" menu option. The editing screen will vanish and you will see the same file directory screen

as when you aren't editing a file. Now you can print, move and delete files in the normal way. To get back to your edit, just press [EXIT]. This might come in handy if you realise while editing that your disc is full and you want to make some space to save your edit. Go to the disc manager and move enough files from the floppy disc to the M: disc to make space. When you've saved the edit you can juggle the files around onto fresh discs.

Typing ahead

If you're working with documents of five pages or more in LocoScript, you will by now have discovered that it takes a long time to scroll from the start to the end if you want to add some text at the bottom. There's a new version of LocoScript due out soon which will avoid this, but you can still partly get round it now.

The PCW stores up keystrokes if you get ahead of it – this is called a "type-ahead" facility. So if you want to get to page 13 of a 20 page document, just type [PAGE] 12 times in quick succession (12 not 13, because you start on page 1, not page 0!). OK, this doesn't get you there any faster, but at least you can go away and make a cup of tea while the PCW does it all for you.

You can use the same principle in many other parts of LocoScript. For example, wherever there are sequences of menus you go through regularly, you can type the relevant sequence of key-strokes all at once without actually waiting for the relevant menus to show on screen.

Using foreign characters

Maybe you're having trouble writing to your pen pals in Outer Mongolia. Here's a quick guide to using accented foreign characters on your PCW.

Look on page 10 of the LocoScript manual – you will see three diagrams. The bottom one shows the characters you get by holding [EXTRA] down and then pressing each key. The keys that are ordinarily the number keys (1 to 0) show a selection of accent marks, accute, grave, umlaut, circumflex etc. To get an accented character, first type the accent (e.g. [EXTRA] + 6 for acute); the cursor will not move on a column on the screen like it normally does. Now type the character (e.g. 'e'), and you will see it properly printed on the screen. It can be deleted, moved and printed like any normal character. Note that c-cedilla has a key ali to itself, [ALT] + [comma].

By the way, these accents are only available in LocoScript, not in other word-processors you might have such as Wordstar or NewWord. ▶

Business stationery

You're running a small business and you use your PCW to produce letters with, but you don't have any preprinted stationery. How do you make letters look a bit snazzier just using LocoScript and the PCW printer? Here are a few ideas involving some use of headers and footers, and changes of print size. If you aren't familiar with altering these settings, don't worry – read our upcoming LocoScript features.

At the top of the page:
 ● Use bold, double width printing for your company name, and centre the text too

● Print your address centred all on one line below the company name, using superscripted full stops as bullet-type separators

At the bottom of the page:
 ● Put your registered address, or owner's name, in the footer of the first page of the letter. Do this by altering the template so that the footer for the first page is set to be different from all the other pages, and then put the names in the footer section of "first page only" of the template. Also, you could print this text in 17 pitch subscript type to make it smaller, as is common on business stationery.

```

PRINTERS/REJECT Editing text Printer idle Using M
-Layout +F1P -L1 -LPB Page 1 Line 11 of 54
F1=Show F2=Layout F3=Emphasis F4=Style F5=Lines F6=Pages F7=Mode F8=Blocks EXIT
-----
(Centre) (*Pitch10) (*Bold)8000 PLUS MAGAZINE
(*PitchPS) (-Bold)
(Centre) (*Super) (*Super) Somerton (*Super) (-Super) Somerset TA11 5AH
(Centre) Telephone Somerton (0458) 74011
( )
( ) (date)
( )
( ) (Addressee)
( )
Dear (name),
    
```

- The template for the letter heading showing Codes and Effectors.

```

PRINTERS/REJECT Editing pagination Printer idle Using M
-Layout +F1P -L1 -LPB Page --- Line -- of 54
F1=Show F2=Emphasis F3=Style F4=Style F5=Lines F6=Pages F7=Options F8=Blocks EXIT
-----
---end of header 1 : used for only the first page
(Centre) (*Sub) (*Pitch17) Directors: F. Bloggs, M. Vole-Strangler, MBE.
---end of footer 1 : used for only the first page
---end of header 2 : used for all pages except the first
---end of footer 2 : used for all pages except the first
    
```

- The commands to set up the footer in the Editing Pagination screen. (Codes and Effectors are switched on).

8000 PLUS MAGAZINE

The Old Barn · Somerton · Somerset TA11 5AH
 Telephone Somerton (0458) 74011

18th September 1986

Mr. B. Spender,
 94 Hopeful Road,
 Brentwood,
 Essex.

Dear Mr. Spender,

Many thanks for your application to edit our new title 8000 Plus. We have had a huge response to the advertisement, and I have seen a large number of very well qualified candidates. Unfortunately, I have to say that you were not among these, but I wish you good luck in your job hunting -- you'll need it.

Yours sincerely,

Anne Person

Directors: F. Bloggs, M. Vole-Strangler, MBE.

- The final letter printed out.

TIPOFFS

Sorting Cardbox files

A correspondent in the November 8000 Plus was advised to buy another database in order to alphabetically sort Cardbox files. She needn't, since there are 'public domain' programs which will do the job just as well.

If you 'WRIte' a WordStar mode file from within Cardbox, you will get the records output to disc in their usual unsorted order. The program SORTV13.COM can then be used to order the records, which can then be used as input to Wordstar for mailmerging, or be printed out directly, or (with a little manipulation) read back into

Cardbox to form a new, sorted, database.

Unsorted Cardbox sufferers should send an SAE to the PD Software Library, Winscombe House, Beacon Road, Crowborough, Sussex TN6 1UL for information.

S Ludlam, University of Sheffield

Feeding continuous paper

The back plate of the PCW printer is normally positioned with the ribs facing forwards and the back plate at a steep angle. This is the normal position for single sheet stationery, but can cause problems when using continuous stationery.

Just remove the back plate and turn it around, so that the ribs face backwards, and replace it on the printer. It will then lie at a flatter angle which is much better for separating incoming from outgoing continuous stationery.

One potential problem can occur with the tractor feed mechanism. Usually the paper bulges in the middle as it goes over the tractors, and this bulge then causes problems when the paper has to bend upwards to go up the back plate..

The cure for this is the latest 'High-Tech' accessory for Joyce - a 10 inch length of 1" dowel! This is placed as a roller across the paper in the dip behind the tractors, where it flattens the offending bulge and allows it to bend freely. This permits carefree printing of long documents unattended without fear of a paper jam.

(Richard Cook, London SW5)

Marking italic text

Have you ever written a long letter and printed it out only to find you have to revise and reprint the whole thing because you forgot to cancel an Italic or Bold command?

Unlike underlining, LocoScript does not show you on the screen when text is in italic or bold, and even if you have switched on the option to Show the codes you can easily forget to turn the effect off on long passages.

All very irritating. But thanks to the 'phrase' memory's capacity to store codes as well as characters, you can devise the perfect Heath Robinson solution. Create a phrase which contains the codes (+Italic) followed by (+Reverse Video); pasting the phrase will call up both

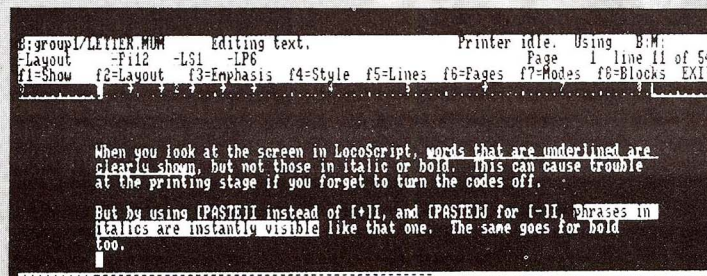
italic, and a visual reference (the reverse video) which you cannot miss but which doesn't affect the printing.

Another phrase containing (-Italic)(-Reverse Video) can be created and pasted in to cancel both effects. A similar process can be used to show Bold or Double

strike passages on the screen.

Since phrases are stored by an alphabet character, you can now turn italics on by [PASTE] I, hardly any different from the [I] you normally use. And it saves you time and trouble in proof reading and correcting.

Steve Gough, London W10



▲ A typical LocoScript screen - italic and bold sections are now highlighted to be seen easily

Working with Basic

There are a couple of shortcuts you can use while typing in Mallard Basic programs, which require some dedicated ferreting to unearth from the depths of the manual.

Probably the most used command in Basic programs is PRINT. In most it has to be typed many times.

However, you can type a question mark instead, with just the same effect. For instance, while in Basic type ?2+2, and you will get 4. Now incorporate it into a program line, eg.

```
30?2+2
```

and then LIST it, you will find it now reads

```
30 PRINT 2+2
```

You do not need a space either between the line number and the ?, or between the ? and what follows, so

there is a saving of several keystrokes.

Another useful shorthand is to replace REM by a single quotation mark:

```
10' This is a comment
```

This has the same effect as REM, but unlike the PRINT? case, it does not actually print out as REM when LISTed. It's often useful to create some breathing space in the middle of a long program, for example:

```
130 (End of one section)
```

```
140 '
```

```
150 (Start of next section)
```

(Rodney M Bennett, London W10)

Saving Basic programs

If you save Basic programs that are not finished or still need work done on them at a later date, then you may end up with a file full of odds and ends. It's then you find you can't

remember what MYFILE.BAS is.

If the program isn't in final form, a good idea is to save it using the form specifier A, ie. SAVE

"MYFILE.BAS,A". This saves the program as an ASCII file, so you can use the CP/M command TYPE. This way you can check your Basic files easily without having to load each

one just to see what it does.

Note that when you want to load a file later (with the LOAD command), it doesn't matter whether the file has been saved using "A" or not. Also, if you do save a file with "A", you can always change your mind and save it normally at a later date.

(Philip Last, Louth)

Cracker of a tip

Here's advice that will be of interest to users of the spreadsheet program "The Cracker".

Making up an autostart disc

As always, first find a blank formatted single density disc, suitable for use in drive A, and using PIP copy onto it J14CPM3.EMS, PIP.COM and SUBMIT.COM. Also copy from the Cracker disc the files CRACKER.OVR, CRACKER.COM and, if you need the help messages while running Cracker, CRACKER.HLP.

In order to create a file to instruct CP/M what is expected of it, put your system disc back in A and type BASIC RPED. Follow the instructions on the screen to create a file called PROFILE.SUB, and type into it the lines PIP M: =A: *.OVR CRACKER

Make sure you save this file on your autostart disc. Now whenever you reset the machine and insert this disc in drive A, The Cracker will start automatically.

Printer instructions

The following instructions will allow you a much greater flexibility of form design within The Cracker. Select a text mode as normal and then use whichever of the following are best suited to your needs:-

- [ALT]+O [RETURN] - shrink print (132 columns possible)
- [ALT]+R [RETURN] - return to normal number of columns
- [ALT]+N [RETURN] - enlarge print
- [ALT]+T [RETURN] - return to normal print size
- [ALT]+P [RETURN] - snapshot of screen
- [ALT]+Q [RETURN] - emphasised print

Please note you cannot edit text in these modes, but in practice this is not restrictive.

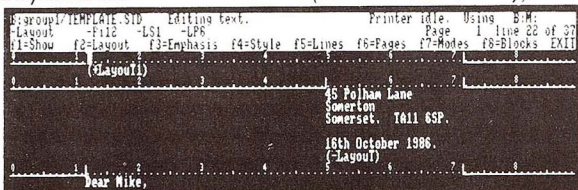
(J S Smith, Dumfries)

Economical addresses

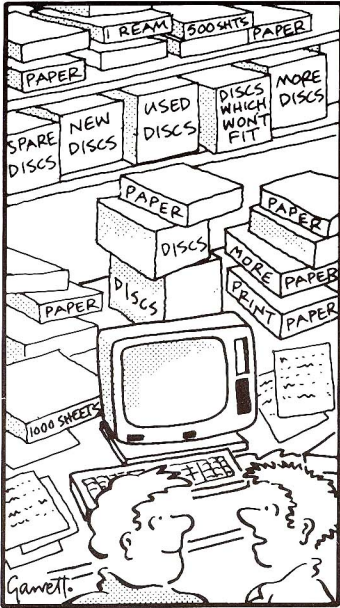
As was pointed out in issue 1 of 8000 Plus, using the TAB key really takes up a lot of disc space. When writing letters that require your address to be in the top right corner you should set up a layout in the

TEMPLATE.STD document so that the left margin is as far across the page as you need.

Then just revert to the base layout after you've typed the address. (Stewart McCall, Corby)



Using different layouts to make address typing easier



"AT LEAST NOW WE'VE GOT THE PCW WE WON'T BE SNOWED UNDER WITH ALL THAT PAPERWORK..."

Editing ASCII files in LocoScript

Producing ASCII files in LocoScript is easy, simply involving use of [F7] (Make ASCII file) at the Main Menu stage. However, it is not immediately evident how you load an ASCII file in order to word process it. Trying to EDIT one directly produces the response "Not a LocoScript document".

Fortunately it is possible to do this. Simply CREATE an empty document and then press [F7] while editing it and select INSERT TEXT. Choose the ASCII file you wish to edit as the text to be inserted, and then use LocoScript to process it normally.

Note that when you save the document again it will be in normal LocoScript form, not ASCII. If you want to, you can use Make ASCII file to reconvert it. (Ross Maynard, Hythe.)

Function keys in CP/M

Although the keys [F1] to [F8] are primarily designed for use with LocoScript, they also have specific meanings within CP/M. They are either useful while running programs like WordStar, or just at the ordinary "A>" prompt.

For example, if you read last month's tip-offs, you will know that [ALT] + P makes the PCW printer echo everything you type from then on until you type it again. The key marked F7/F8 will do this at a single keystroke.

The full definitions are as follows:
 F1/F2 is [ALT] + Z
 F3/F4 is [ALT] + Q
 F5/F6 is [ALT] + S
 F7/F8 is [ALT] + P

(Martin Evans, Cambridge)

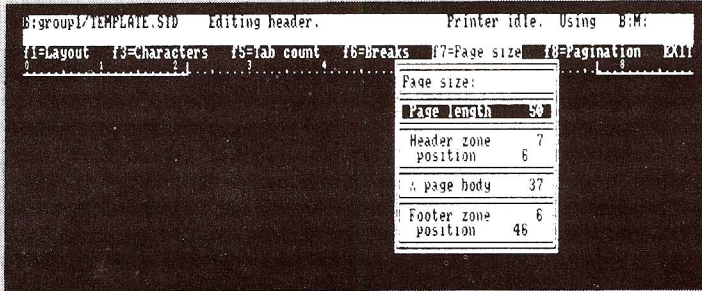
Using A5 paper

If you want to print using A5 paper then you will have to alter the margins and page length to suit the new size. Start editing the TEMPLATE.STD that you will use, and press 'F7 Modes', select "Edit Header", and finally 'F7 Options' to get to the "Editing Header" screen.

For a page done in Pitch 12 and Line Pitch 6, assuming 1 line headers and footers with a single line space between them and the document, the suggested page settings are as

follows: To alter the margins, select 'F1 Layout', and set the left and right margins to 22 and 77 respectively. Then press [EXIT] to get back to the previous screen, and press 'F7 Page Size'. Set the options as shown below.

Finally, before printing an A5 document out, go into the Printer Control State by pressing [PTR], and select 'F1 Options'. Move the cursor down to Form Length, and change it to 50. (Philip Last, Louth)



The page layout options set up for A5 paper

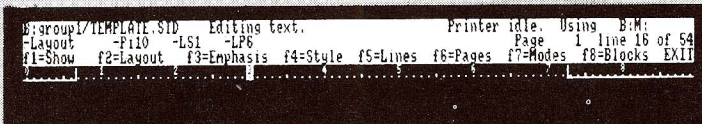
Margin settings

If you have ever tried setting up your own templates then you will no doubt have discovered by now that changing the pitch isn't enough - you need to change the margin settings as well.

This table shows suggested margin

Pitch	Left Margin	Right Margin
10	07	73
12	09	89
PS	09	89
15	11	110
17	13	127

positions for the five pitch sizes on A4 paper. The settings give, as near as possible, a 20mm margin to left and right assuming Right Justification is on. If not, then the right hand margin can usually be a little wider. (Philip Last, Louth)



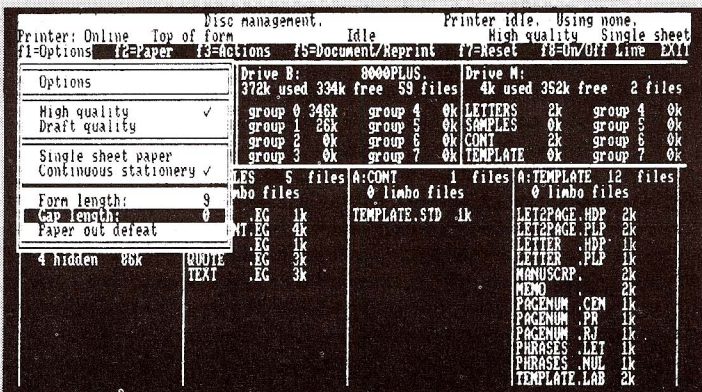
The base layout ruler for Pitch 10 text.

Label printing

When using LocoScript for printing labels you'll probably find that a lot of experimentation is required before you find correct settings for the printer. With the labels that are sold by most companies you'll find that the sizes are different to those that

are recommended by Amstrad.

For labels 3 1/2 by 1 7/16 inches on continuous stationery, you will need to set the following options before you print, by pressing [PTR] and using the 'F1 Options' menu:



Deleting LocoScript phrases

When preparing text for printing in 15 or 17 pitch, it is often a nuisance that lines of more than 90 characters run off the right hand edge of the screen.

If the margins for the base layout are set at say 0 and 80, and Layout 1 at the full width settings, then the text can be prepared in the base layout so that it is all visible. Then as the last thing, just insert Layout 1 at the top of the document, press [SHIFT] + [PAGE] to run to the end of it and everything will be re-laid in the correct width for printing. (Margaret Rugg, Exeter)

Long lines in LocoScript

Since there is a limit to the overall number of characters you can have stored as phrases, if you want to define some very long phrases you might have to delete some others to make room.

Unfortunately there is no "delete phrase" function directly available. What you have to do is store a new phrase with nothing in it in place of the old phrase. To do this, press [COPY] [COPY] and then the letter of the phrase you wish to delete.

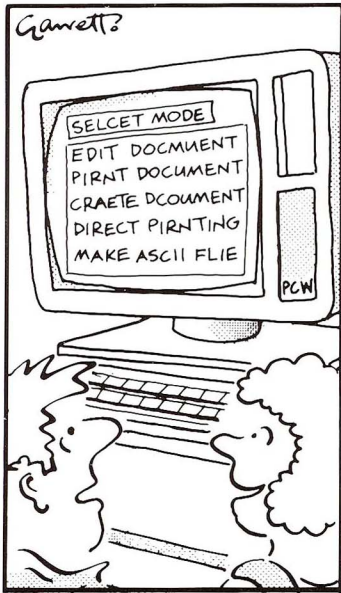
Don't forget that amending phrases is normally only effective for the current work session and you have to save the changes (or deletions) if you want to use them again. When you 'F8 Save All Phrases', LocoScript creates a new PHRASES.STD file in Group 0 of Drive M. This will have to be 'F3 Copy'd across to your Load-Up disc. (Richard Cook, London SW5)

Overcoming LocoScript's sloth

LocoScript is often said to be very slow in handling long documents. One way round this is to split a long document into several short files, at most 10 or 12k long each. Editing each one is now relatively painless, since they are quite short.

Now when you've finished all the editing and want to print out the whole document, edit the first file and move to the end of it. Make sure it ends in [RETURN] and the special features like italic and bold are turned off. Then use [F7] and "Insert text" to insert the second file, which will run through ending conveniently at the place to insert text again. Now use [F7] to insert the third file, and so on.

This way LocoScript will easily handle a file of up to half the disc space in length - 360k on an 8512. Even Tolstoy would have been able to manage with that! (Margaret Rugg, Exeter)



" TRUST ME TO GET THE DYSPLEXIC VERSION ... "

Getting a bigger disc drive

There is a simple underhand way of upping the size of your disc drive by using CP/M's SETDEF utility. You can tell CP/M that if it doesn't find a file on a certain drive, to look on another one for it. Since it will then do this automatically, the second drive can be thought of as an expansion of the first. Here's how it works:

If you have an 8256 and your program is too big to fit in its M: drive, then copy over all the files that will fit anyway. Make up a work disc which contains only the files that wouldn't fit on M:, which hopefully leaves you with a generous space left to work with on the disc.

Now, with A as your default drive, type
SETDEF M: , *

Whenever you try to read a file while using A as the default drive, it will now look first on the M drive

and then on the A drive without you having to specify which, just as though you had a single drive big enough to hold all the files at once. If you create a new file, it will go to the default drive, A, rather than M, so new files are safely stored on a real floppy rather than the more dangerous RAM disc.

(Robert Broome, Computer Training College)

Ribbon re-inking

Some PCWs used in businesses get very heavy use, doing non-stop printing every day. The printer ribbons only last for a few hundred pages or so, and if you have buy a new ribbon twice a week it can get pretty expensive.

Don't fork out £5.95 for a new ribbon without first rejuvenating the old one by spraying it with WD-40 – best take the ribbon out of the printer first though!

(Glenn Myers, Beaconsfield)

TipOffs editor adds: another solution to this is to reuse the ribbon by re-inking it. One company who will do this is Aladdink of 4 Hurkur Crescent, Byemouth, Berwickshire TD14 5AP. They charge around £1.80 per ribbon.

Scrolls of screens

One of the frustrations of the PCW is its inability to scroll the screen by a pageful at a time. Many people have been reading a long document on screen and pressed [PAGE], only to discover the top few lines of text disappearing off the top of the screen.

However, if you press [SHIFT] with the up or down cursor keys, you will see that this moves the cursor by approximately 2/3 of a screenful at one go.

Also, [SHIFT] plus the left or right cursor keys moves the cursor to either the end of the line or (better still) back to its starting point.

(Glenn Myers, Beaconsfield)

Basic hacking

Here's a challenge for all CP/M and Basic experts out there. The short listing shown below will allow you to look at any part of the PCW's memory, although since you are running Basic the TPA will necessarily always show Mallard Basic.

A couple of fun areas to look at are around 4500 (Basic error messages) and around 20000 (Basic keywords). But here's an oddity – the characters at and following 22466 say "Acorn computers". Can anyone explain why?

(Raymond Holroyde, Manchester)

```
10 INPUT "Start";st
20 INPUT "End";ed
30 FOR a=st TO ed
40 mem$=CHR$(PEEK(a))
50 IF ASC(mem$)>127 THEN mem$=CHR$(ASC(mem$)-128)+CHR$(9)
60 IF ASC(mem$)<32 THEN mem$=CHR$(9)
70 PRINT mem$;
80 NEXT:END
```

To coin a phrase

LocoScript's phrases provide a convenient way of typing often used words into documents. Owing to covert pressure from the Estate Agents Mafia, the standard set of phrases you get includes such useful gems as "convenient for the M62" and "on frequent bus route".

However, there is a second set of phrases which are useful for letter writing hidden in the group named TEMPLATES of the LocoScript master disc, in the file PHRASES.LET. These are:

- A: For the attention of
- B: By return of post
- C: With Compliments
- D: Dear Sir/Madam
- E: Please find enclosed
- F: Yours faithfully
- L: Thank you for your letter of
- M: Please mark the envelope for the attention of
- O: Our Ref:
- P: Private and Confidential
- R: We look forward to your reply
- T: Thank you
- V: R.S.V.P
- W: With reference to your
- Y: Your Ref:

To be able to use these, take your LocoScript startup disc and rename (f5) PHRASES.STD as PHRASES.OLD. Then move (f4) PHRASES.LET to the first group renaming it PHRASES.STD. The new phrases then come into effect the next time the computer is switched on. There is room for you to add a number of other phrases such as address and name, "Yours sincerely" etc.

(Jacquie Errington, Woodbridge)

Using PIP to avoid LocoScript..

The LocoScript 'f3 Copy' and 'f4 Move' commands only handle one file at a time, so it gets a little tedious when, for example, you want to move the contents of an entire group to a new disc. However, you can use CP/M's command PIP to sort out LocoScript discs.

The key point is that LocoScript and CP/M both use the same file and directory structure. CP/M has 16 "user areas" which are intended to keep different people's files separate, but on single user machines like the PCW this is redundant and you normally only use area 0.

Put your LocoScript disc in and type DIR – only the documents in group 0 are listed. If you type DIR [USER=ALL], you see the documents in all the groups being listed. LocoScript groups 0 - 7 (as they are called before you give them names) occupy user areas 0 - 7, whilst Limbo files are in areas 8-15 – the Limbo files for group X are in area 8 + X.

So using your CP/M expertise you can transfer whole groups of LocoScript documents to a new disc using PIP. Suppose you want to move group 4 of one disc to group 1 of a new disc, because the old disc is too full. Note how the group numbers 1 and 4 are used in steps 2 and 3, so you can change them to the numbers you actually want to use.

1. Put your CP/M disc in and type PIP [RETURN]
2. Put your old LocoScript disc in drive B and the new disc in drive A and type
A:[G1]=B:*. *[G4]

On single disc machines, do this in two parts via the M drive to avoid disc changes for every file. Type [RETURN] to leave PIP.

3. If you want to delete the files from the old disc, type
USER 4
ERA *.*

4. If you want to keep the limbo files too, copy across group 12 on the old disc to group 9 on the new disc in the same way.

(David G Smith, Paddock Wood)

...and using LocoScript to avoid PIP

On the other hand, you may be a CP/M user who finds PIP just too horrible to contemplate. If you cannot figure out the CP/M commands to PIP files around, use LocoScript as a tool.

Do this by starting up LocoScript, then insert the disc which you wish to copy a file FROM, and press the f1 key to get the directory. Copy (f3) the documents into the M drive, then remove the disc and insert the disc to copy the files TO. Press f1 again, and then use Move (f4) to complete the procedure by transporting the copies from the M drive to the A drive again.

It might sound long winded but really it is very simple, and it has the advantage that you can see exactly what is going on and the size of the files being moved.

(Matthew Tod, Stoke)

Improved underlining

When it comes to underlining a heading, LocoScript leaves something to be desired. The line is squashed up against the word and looks rather untidy. All is not lost however – it is possible to obtain far more stylish underlined headings by altering the line spacing and using other characters for the line. Here is how it is done:

1. At the start of a new line set the line spacing (with [F5]) to 1/2 and the line pitch to 8. You can't use proportional spacing, so if you are, change to Pitch 12 for a similar effect.

2. Type in the required heading and press [RETURN]
3. Reset the line pitch and spacing to what you were using before
4. Turn on full underlining (F3).
5. Turn off the codes display (F1) so that you can line up the columns accurately on the screen for the next stage
6. Type spaces until the underline line is as long as the heading. (If you type underline characters, although the screen looks OK they will print out as a broken line instead of continuous).

7. When you are happy, turn off underlining, press [RETURN] and carry on with your text. Remember to change back to proportional spacing if you want to.

To speed this up, you could store the basic control commands in a phrase on your startup disc. Other underline effects can be obtained with the use of superscript and characters such as = or *.
(Simon Barrett, Cardiff
Peter Wall, Letchworth)

```
E:group1/HEADINGS.      Editing text.      Printer idle. Using B:
-Layou1 -P112 -LS1 -LP6      Page 1 line 1 of 37
f1=Show f2=Layou1 f3=Emphasis f4=Style f5=Lines f6=Pages f7=Modes f8=Blocks EXIT
```

```
(#UL)This is the normal underlined heading(-UL)
(#LSpace2)(#LPitch8)This is the improved underlined heading no. 1
(#LSpace1)(#UL)_____(-UL)(-LPitch)
(#LSpace2)This is heading no. 2
(#LSpace1)(#SUB)===== (-SUB)
(#LSpace2)This is heading no. 3
(#LSpace1)(#SUB)***** (-SUB)
```

This is the normal underlined heading

This is the improved underlined heading no. 1

This is heading no. 2

This is heading no. 3

Some examples of different style headers

Mathematicians' delight

If you use your PCW to type mathematical formulae, you may just have been frustrated by the lack of symbols for square and cube. Here's how to get them:

```
(#Pitch17)(#Super)2(-Super)(-Pitch)
(#Pitch17)(#Super)3(-Super)(-Pitch)
```

They are a little cumbersome but can be made simple to use by saving them as LocoScript phrases, under S and C respectively. Then they are always available as "[PASTE] S" or "[PASTE] C".

Because superscripted letters are half the height but with the same width as full size letters, they should be printed in pitch 17 to prevent them looking stretched.

$$a^2 \times a^3 = a^5$$

An example of squares, cubes and other powers

If you like decimal points that look like decimal points and not full stops, then use SuperScript full stops – (+Super) . (-Super) . Again, if you use it a lot, save it as a phrase.

Although the PCW can print most of the simple fractions (use the number keys on the top line, together with the [ALT] key), more complex fractions can also be got at for a little more effort. Put the numerator in superscript, and the denominator in subscript, and again use Pitch 17 to give the characters the correct look:

```
(#Pitch17)(#Super)x(-Super)/
(#Sub)y(-Sub)(-Pitch)
```

(Philip Last, Louth)

Screen contrast

Often it is easier to see the screen, particularly on a sunny day, if it is in reverse video. There is no way of doing this for LocoScript, but you can do it for CP/M programs.

When you are running CP/M and faced with the familiar "A>" prompt, put in your CP/M master disc with the file PALETTE.COM on it and type

```
PALETTE 1 0 [RETURN]
Similarly, typing PALETTE 0 1 returns you to normal green-on-black text screen.
```

If you know how to customise discs with a PROFILE.SUB file, you might like to make up special discs for yourself. Leave one side of the disc normal, and on the other side have exactly the same files but include a PROFILE.SUB file with the reverse video command.

Mark the side of the disc "Normal" and "Reverse Video" (or "Cloudy and "Sunny"), then depending on your mood or the strength of the sun your programs can be run in either screen mode.
(Justin Dennison, Guildford)

Jazzing up your screen

You've probably seen how LocoScript manages to underline characters on the screen, and print text in reverse video, and wondered how it is done. With a bit of trickery, you can add some spice to your boring screenfuls of text in just this way.

To underline some text, before you print it out, print the characters CHR\$(27) "r", and afterwards CHR\$(27) "u" to turn underlining off.

Similarly, for reverse video, print CHR\$(27) "p" beforehand and CHR\$(27) "q" afterwards.

As a final frill, the command PRINT CHR\$(7) causes the PCW to beep at you. One

good idea is to put a beep command just before an INPUT statement, to

remind users that they are expected to type something.

```
list
10 PRINT CHR$(27)"r"
20 PRINT "This is some underlined text"
30 PRINT CHR$(27)"u"
40 PRINT CHR$(27)"p"
50 PRINT "And this is some reverse video"
60 PRINT CHR$(27)"q"
Ok

run
This is some underlined text

And this is some reverse video

Ok
```

Underlined and reverse video text

BASIC BONANZA

The Basic manual that Amstrad provides with the PCW machines is notable mainly for its total opacity. Some useful facts just aren't in there, and your chances of finding the ones that are printed are slightly less than the odds on Ronald Reagan joining the Communist Party.

GEOFFREY CHILDS of Winchcombe has written in with a hoard of invaluable tips to help programmers get the most from Mallard Basic. You will need to know the bare bones of Basic to be able to use these hints effectively ...

Direct command editing

If you are writing a line in Mallard for direct execution and you make a typing mistake, then you probably curse and retype the whole thing. This gets pretty annoying if it's a long line, like a complex calculation.

However, you don't have to go back and start again. Type [ALT]+A and you have your command line back for editing with the arrow and delete keys.

Looking at your program

If you are interested to find out how Mallard Basic works behind the scenes, you can see how it stores the text of your program. Using the PRINT and PEEK commands, take a look at the area of memory around 31382, and you will see part of your program stored there. The special Basic words like PRINT are not stored letter by letter, but you'll certainly see any characters you've typed in between string quotes appearing.

If you know how to use it, you can modify your own program by the back door, using POKE to alter the lines directly. But be warned, ill-advised use of this can cause the PCW to crash.

```
list
10 a$="Zaphod Beeblebrox"
20 PRINT a$
Ok
for n=31382 to 31420: ?chr$(peek(n));:next
éi"Zaphod Beeblebrox"
Ok
for n=31382 to 31420: ?peek(n),:next
29 0 10 0 3 0
0 225 236 34 90 97
112 104 111 100 32 66
101 101 98 108 101 98
114 111 120 34 0 10
0 20 0 179 3 0
0 225 0
Ok
```

Two simple lines of Basic text, and a look at the way they are stored.

Cursory commands

If you have written a Basic program which moves the cursor around the screen a lot, it can be very annoying to see the wretched cursor bobbing up all over the place and wrecking your beautiful tidy screen. You can turn the cursor off by the command PRINT CHR\$(27)"r".

When the program ends, you will need to turn the cursor back on again (have you ever tried editing without a cursor?), which you do by PRINT CHR\$(27)"e".

Altering the TAB command

When you use the TAB command in Basic, the cursor moves to the column you specify and pads the previous columns with blanks. It would be nice to be able to use some character other than blank as the filler.

There is a way to do this in Mallard. Try typing POKE 24348,42

PRINT TAB(50)

and you will see the effect. For the more technically minded, 42 is the ASCII code for "*".

Probably the most useful filler character is the full stop, for which the ASCII code is 46, so type that instead of 42 in the example. You can use any character you like, and the codes are listed on pages 113 to 118 of the PCW CP/M manual.

```
print "price: ";tab(30)"£5"
price:                               £5
Ok
poke 24348,46
Ok
print "price: ";tab(30)"£5"
price:.....£5
Ok
```

The effect of altering the TAB command

Sending text to the printer

Have you ever written a program that normally prints to the screen, but that might need hard copy on occasions? The simplest way to achieve this is by changing PRINT to LPRINT. This could mean some hard work editing if you have a lot of lines. Instead you can do it by typing POKE 18527,90

and from then on all PRINTs will appear on the printer, not the screen. You can reset this to normal by POKE 18527,100

Telling the time

The PCW has a clock inside, although it counts the seconds in a rather eccentric way. The simplest way to explain how it can be used is to show a simple Basic program which reads the clock.

As you will see, the essential information is stored in three bytes, numbered 64502, 64503 and 64504. Not only can you read these, but you can also set them by POKE-ing them with 0 so you can "zero" your clock.

```
10 DEF FNT(X)=X-INT(X/16)*6
20 HRS=FNT(PEEK(64502!))
30 MINS=FNT(PEEK(64503!))
40 SECS=FNT(PEEK(64504!))
50 PRINT"Time is" HRS"hours" MINS"
mins" SECS"secs"
```

The program to read the time from the PCW memory

```
run
Time is 0 hrs 32 mins 56 secs
Ok
poke 64502,0:poke 64503,0:poke 64504,0
Ok
run
Time is 0 hrs 0 mins 5 secs
Ok
```

An example showing clock setting and reading

SuperCalc to LocoScript

Spreadsheets can be very powerful tools for manipulating and presenting charts of data, and SuperCalc is widely used by PCW owners. One snag is that if you are writing formal reports, you need to be able to work the spreadsheet results into your text, but SuperCalc printouts are usually pretty bald tables.

Having spent many hours constructing the spreadsheet and entering the data, you may have been resorting to re-typing the results into LocoScript, making extensive use of the decimal tab facility. In fact, it is possible to transport your final results from SuperCalc2 into a LocoScript document, keeping the same layout as the spreadsheet used.

All that is necessary is as follows:

1. Finalise your layout in SuperCalc2 and turn off the borders display if this is not

required on the final printout

2. Save the area you want to put in your LocoScript file, by 'Output'ing it to the disc, not by 'Save'ing it. To do this, use the commands / O u t p u t D i s p l a y D i s k. This will create a file on the disc with a name you choose, for example 'yourname.prn'.
3. Start LocoScript up. The file created by SuperCalc must be copied onto the M drive: put the SuperCalc work disc in the drive, and use [f3] to copy it to a group on M.
4. Change back to your LocoScript work disc, and create the report document up to the point where you want the SuperCalc results to appear. Make sure you are not using proportional spacing.
5. If your spreadsheet data is more than 80 columns wide, you must set your margins wide enough to accommodate the total width, and your pitch to one which will allow the table to fit across the page (eg. 15 or 17 pitch). You can get up to 140 columns or so across an A4 sheet in 17 pitch.
6. Once you are all set with the correct layout, press [f7] and select the 'Insert Text' option. You will be faced with the Disc Management screen - highlight the SuperCalc2 file which you copied to the M drive, and press [ENTER] twice to do the text insertion.
7. Sit back with a smile of satisfaction as the SuperCalc2 display you saved is placed perfectly into your document.

CA Stanley, Mansfield

Bypassing passwords

Remember that first week, reading all about CP/M and discovering you could set passwords on files? Now a month or two later, faced with a disc of files with read protection, you can't remember any passwords! There doesn't seem to be any way to circumvent the password system from inside CP/M without being a hacker, but there is a way to cope.

Start up LocoScript, and then put your CP/M disc with the protected files on it in the disc drive (and press [f1] to see its contents on the Disc Management screen). Now "Create" an empty file, select "Insert Text" from the [f7] menu, and choose the CP/M file that you need to unprotect.

Finally, when you've finished the edit select "Make ASCII file" from the Disc Management screen to create a file that CP/M can make sense of. Voila. This method, of course, only works with text files. You can't use it on a .COM file, for instance.

Chris Lilley, Ashfield

Using [ALT] with keys

When you need to use an indent tab, for example, the manual tells you to press [ALT] and [TAB] together. Sometimes it seems to work, and the magic symbol appears on the screen. But more often than not, it doesn't work and you are faced with the ordinary tab symbol on the screen.

However, there is a secret knack to doing this right every time. The trick is to press the [ALT] key first of all, and then while still holding it down press the [TAB] key too. It works like a charm every time.

RH Gladden, Crewe

This technique is correct for all [ALT] sequences. You use [ALT] in exactly the same way as you use the [SHIFT] key to get capital letters. ▶

That's Life

It has become traditional to hide a game called "Life" in computer operating systems - Unix has one, as do many other large computer operating systems. The PCW has one hidden amongst its LocoScript files on side 1 of the systems discs.

First, an explanation of how to play Life. It was developed as a diversion by a group of mathematicians some years ago, and follows a few simple rules. Think of a colony of bacteria which breed and change as generations go by.

The rules for breeding are simple: any bacterium with two or three neighbours stays alive in the next generation. If a bacterium has less than two neighbours, it dies of loneliness, and if more than three it

dies of overcrowding. If any empty space on the screen has exactly three neighbours then a new bacterium is created in that space for the next generation. If you aren't sure how the number of neighbours is worked out, imagine a Noughts and Crosses game grid - the neighbours of a cell are the eight cells around the central one.

To "play" life, you set up an initial arrangement of cells (the bacteria), then sit back and watch the generations pass. The object is to find starting patterns which survive (some patterns become extinct after a few generations), and also which look pretty on the screen!

So how does the PCW do it? Life is cunningly hidden in the communications program called MAIL232.COM, which is on the

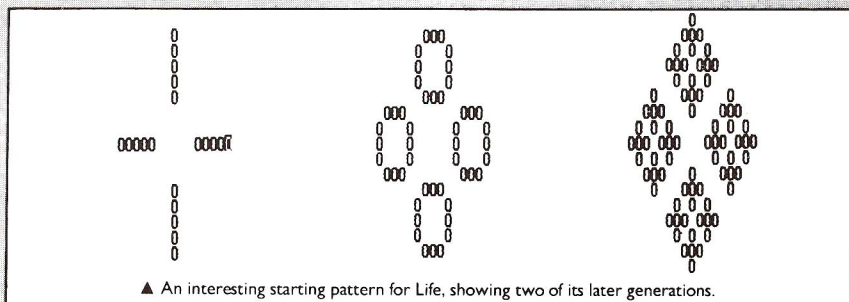
LocoScript system disc. To run it, boot up CP/M, then put your LocoScript disc in the drive. Type MAIL232 [RETURN], wait for the screen to clear, and then press the [f3] key. Move the cursor bar to the last option on the menu that appears, "Transfer as ASCII" then hold down the [EXTRA] key and press P.

Now the screen goes totally blank apart from a cursor. Here's where you set up the initial bacterium pattern. Use the arrow keys to move the cursor, and press the [RETURN] key whenever you want to place a bacterium at the cursor position. Pressing [RETURN] again will delete the bacterium if you make a mistake. Once you're ready, press the space bar and the generations will roll by before your eyes. To stop the process, press space again.

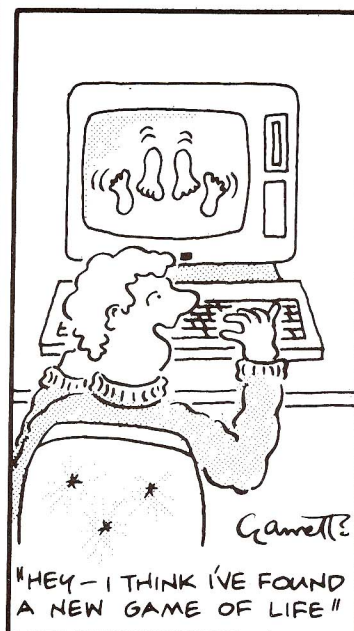
Chris Lilley, Ashfield



▲ The main screen of MAIL232, at the point where you press [EXTRA]+P to set Life going.



▲ An interesting starting pattern for Life, showing two of its later generations.



"HEY - I THINK I'VE FOUND A NEW GAME OF LIFE"

Non-standard printers from BASIC

If you have a special printer connected up to the Centronics parallel interface of the PCW, but you still want to use the standard printer for quick draft printing, you might be getting annoyed at the trouble it takes to switch between them.

If you are using LPRINT statements from BASIC, here's a way to quickly switch between the two different printers. Type in this listing to go on the end of your program (make sure the line numbers don't clash), and then the command GOSUB 40000 will direct all output to the standard printer, and GOSUB 40100 switches it to the Centronics interface printer.

```
40000 POKE 8796, &HEA
40010 POKE 8797, &HFD
40020 RETURN
40100 POKE 8796, &HD0
40110 POKE 8797, &HFE
40120 RETURN
```

N.L. Bowen

STOP right there!

Many people don't realise that the [STOP] key in LocoScript can avert disaster, or at least mitigate its consequences.

If you accidentally pressed [DOC] and can see yourself in for a three hour wait as your life scrolls by your eyes, pressing [STOP] twice will cancel the operation. It is also useful if you press [CUT] at the wrong time and see your last day's work being excised - [STOP] will curb the damage.

The only thing [STOP] will not halt is the progress of exiting from a file, and certain other disc management operations like file copying.

Richard Hopper, Buckfastleigh
JT Brien, Ballinamallard

Addressing labels without tears

A neat way to address labels and envelopes involves using the PASTE feature of LocoScript.

When you type a letter, of course you have the addressee's details somewhere in it. Before you finish editing it, position the cursor at the start of the name, and press [COPY]. Move to the end of the address, and make it into a Phrase by pressing [COPY] and a letter, like A. Now you can close the document and return to the Disc Management screen.

Now for the clever part. To print the address label, just go into direct printing (press D) and then do [PASTE] A (or whatever letter you chose instead of 'A'). This prints out the stored address very neatly.
Garett Hayes, Co. Dublin; Mary C Duncan, Aberdeen; Alex Bruce, Harrow

Hyphenation in LocoScript

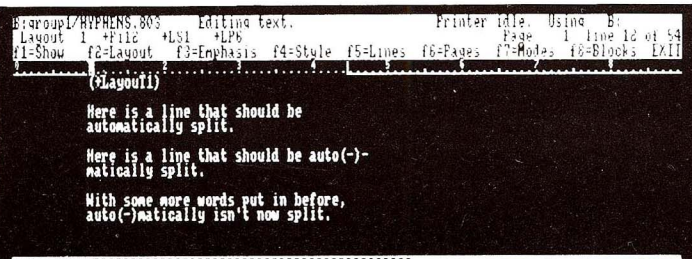
You can add a bit of class to your documents by understanding how hyphens work in LocoScript.

There are two kinds of hyphen - "soft" and "hard". A hard hyphen will print out as a normal hyphen on the screen, but as LocoScript knows not to break the word over two lines. This is useful for phrases like "toss-up", if you want to ensure that you don't get "toss-" on one line and "up" on the next.

Soft hyphens are a bit more subtle. If you have a long word in a letter, it might not quite fit at the end of one line and so LocoScript will put it at the start of the next, leaving the

previous line looking unnaturally short. If you insert a soft hyphen in the word, then LocoScript will break the word over two lines. If you subsequently edit the document so that the long word is not over a line break any more, then LocoScript will not print out the hyphen as part of the word.

Hard and soft hyphens can be got at from the 'f5' menu while editing. Alternatively, the [+] menu can be used to set a hard hyphen, and the [-] menu for a soft one. Just typing a hyphen normally in a word is a sort of mixed hard and soft hyphen. The hyphen is always printed in the word (like a hard hyphen), but if the line length demands it LocoScript will break the word at the hyphen, like a soft hyphen.



Here is a line that should be automatically split.

▲ A line with no hyphens

Here is a line that should be auto-matically split.

▲ Inserting a soft hyphen allows a word break

With some more words put in before, automatically isn't now split.

▲ But the hyphen won't show if it isn't needed

Centring large sections of text

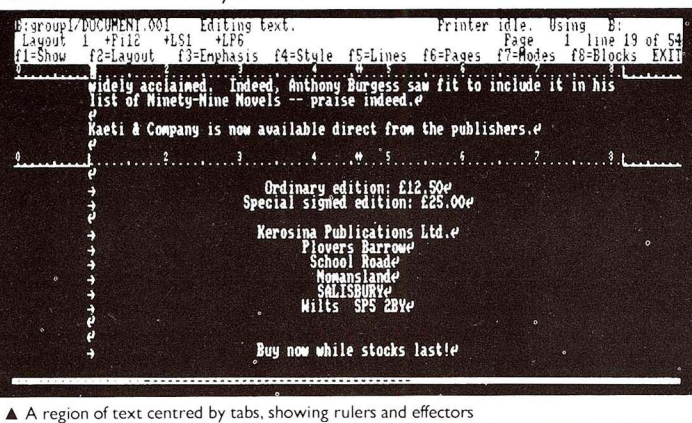
The 'f5 lines' selection on the Editing Text screen to produce centred lines is most useful in the production of small advertising leaflets and so forth.

The drawback is that the centring process stops operating at the end of each line, leaving you back at square one with the need to enter the command once again. A better solution is to have a "Centre Tab" defined in the middle of your line, ie.

at the "46" mark on the ruler for each Layout that you use.

To do this, whenever you define a layout (using 'f2 Layout' and 'brand New layout', position the cursor in the middle of the ruler (usually at mark '46') and press the 'f5 Centre Tab' key. Now when you use this ruler, just press [TAB] at the start of each line you need to centre, which is much more convenient for long stretches of centred text.

Frank Armstrong, Stockport



▲ A region of text centred by tabs, showing rulers and effectors

Printing multiple copies of a document

Many people are disappointed to discover that there are no facilities for LocoScript to print out several copies of the same letter or document. For instance, in organising church services you might want to circulate a number of people in the congregation, or provide the month's hymn lists for the choir. Here is one way to overcome the problem:

When you have edited your text, make sure you put an "end page here" marker using the 'f6 Pages' menu. Then press [COPY] and make the entire document into a block in the normal way - make sure you include the end of page marker in the block. Suppose you called this block number 1, then to get another copy you would just press [PASTE] and then 1.

To get 10 copies, repeat this 9 times in quick succession, and then go away and do something else for a few minutes while LocoScript does its stuff. When you come back, you will find 10 identical pages in your document, and you can then print the whole document in one go using the 'Print all pages' option of the print command, and if you are using continuous stationery you can go away and have another cup of tea while it prints.

Do make sure your stationery is feeding properly before you depart, though, and that it has somewhere to go other than to roll itself around the printer roller! One final tip: if you are making up long documents with repeat pages in the way just described, it is best to store the document in the M drive. This way you avoid cluttering up your valuable floppy disc space with 20 copies of the same page.

Owen Vigeon, Lancaster

Preventing word breaks

People who aren't very used to word processors may not understand the idea and use of "hard spaces". Since LocoScript automatically decides where the ends of lines are to occur, you may find that things like "Mr. Smith" get unavoidably split, with the "Mr." at the end of one line and the "Smith" at the beginning of the next.

The solution to this is to replace the gap between "Mr." and "Smith" by a hard space. This means that although it prints out as a space, LocoScript knows not to split the two words up, but to run the whole phrase on the next line.

To type the name so that it won't be split up by LocoScript, type the first word, then don't press the space bar but input a hard space instead. Do this either with the f5 key, or by pressing the [+] key and then the space bar. Finally, type the remainder of the phrase.

Programs that change the key layout

Programs like NewWord and WordStar alter the PCW keyboard to suit their own needs, which is very nice. Unfortunately, when you finish using these programs the keys are still set up for them which can make running something else afterwards impossible without resetting the machine entirely.

The way around this is to use the SETKEYS program to reset the keys to normal CP/M use. For example, following on from the NewWord boot disc program published in issue 1's TipOffs, create a file called "KEYS.CPM" containing the lines

```
14 NS "↑" E #91 "↑_"
06 NS "↑F"
E #94 "↑F"
79 NS "↑↑"
E #96 "↑↑"
15 NS "↑A"
E #93 "↑A"
```

Then alter the PROFILE.SUB file so that it automatically uses SETKEYS to reset the keyboard when you exit from NewWord, like this:

```
PIP
<M: =NW.COM
<M: =NW*.OVR
<
SETKEYS KEYS.WP
M:
NW
<LB
A:
```

```
SETKEYS KEYS.CPM
Now on leaving NewWord, other programs which expect the cursor keys to behave 'normally' will work correctly.
A Hooley, Bury
```

LocoScript alarm calls

As everyone knows, some of LocoScript's operations like scrolling through a file or exchanging every occurrence of a word for another one throughout a document can take a long time. Tea time, you say, and rush off as ever for the kettle.

LocoScript can be made to give you an alarm call when it has finished its labours. The trick is this: once you've finished giving LocoScript its instructions and it is going about its slow way, press a meaningless key like the # grid at the centre of the four cursor keys several times. When LocoScript finishes its work, it will see this meaningless instruction awaiting it and will beep long and loud in protest, once for every time you pressed the key.

For example, if saving a long file, press [EXIT], [ENTER] and then # several times: now you can go away, and the PCW will beep when it is ready for the next command.

Ray Gladden Crewe
Mike Ninnim, Burnley

Touch typing tip

Bona fide ten fingered touch typists might find the PCW keyboard rather flat. So to increase its slope, you can

glue a couple of rubber feet to its own usual pads. This raises the back of the keyboard by about half an inch, and makes touch typing much easier and faster.

Justin Dennison, Guildford

Safe from prying eyes

How often have you written something to disc intending it to be printed out once and then discarded, and then realised that you would really like to get rid of it altogether? Perhaps for reasons of confidentiality you simply don't want anybody else to see it.

So what do you do? Simple, you ERASE it using f6 from the Disc Management screen. But of course you won't really have got rid of it, but you will have moved it to the "Limbo" section from where it can be easily recovered, perhaps to your embarrassment or worse.

Well, there is an answer. Again from the Disc Management screen select 'f8' and press the [+] key over the "Limbo" option. Now the directory will show the Limbo files. If you had deleted a file called TOPSECR.LET it will still be shown as TOPSECR.LET on the directory, but will say "limbo" at its right. Place the cursor over that and 'f6 ERASE' it again, and it will have completely eliminated that file.

But be warned, use this facility with care and only erase files in this way if you are quite sure that you never want to see them again, because you won't.

Wolf Ruskin, London NW1

Hey, Robin!

Now for a Batman tip: collect the Bag and Boots first, then the Thruster which is hiding behind a box. The Belt isn't then far away but you need to go back and find a bit of "speed" to make a quick dash. One more hint, if you haven't yet realised it; the game takes place on seven or eight levels and you are not on the top one when you start!

Max Batten, Bromley

High Quality print in SuperCalc

One of the annoying features of SuperCalc is that no matter how you set up the printer beforehand, the text always appears in draft quality.

The snag is that just before it starts printing, SuperCalc chooses draft quality to output in. A way around this is to give the print command from SuperCalc, and then just as it starts going press the [PTR] key and change to high quality print. Although the first few characters will be in draft, the remainder of the output will be in high quality print.

Because of this, it is handy to make sure that the first line of your spreadsheet is always blank to give you time to get to the [PTR] key.

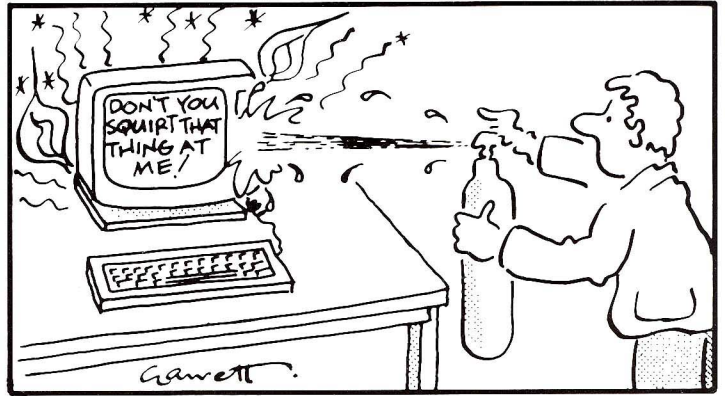
JR Hobart, Wickford

TipOffs editor adds: This is a nice and easy solution, but there is a way to

ensure that the print is in high quality from the word go, although it takes a bit more effort.

Before it starts printing SuperCalc resets the printer, which normally takes it back to draft mode. What you have to do is alter the printer's default setting to high quality, so that after a reset it goes back to NLQ mode.

If you have read the CP/M article this month on setting up the printer using "escape codes", you will see how this is done: before you run SuperCalc, set the print to High Quality with the [PTR] key, and then use BASIC to send ESC d to the printer. Now SuperCalc will always print in High Quality.



Be bold – be fast

The two print options that the PCW provides – draft or high quality – seem to be two extremes. Draft is very mucky, but high quality is very slow.

To produce documents of reasonable quality at a decent speed try using the 'Bold' command in the text, and then use the 'Draft Quality' mode to print the document out.

Put the 'Bold' command at the

very start of the document, and remember that if you insert new layouts later on you may have to re-enter the bold command. To get draft quality for printing, press the [PTR] key, then from the 'f1 Options' menu select the 'Draft Quality' option. This will give a kind of intermediate quality that is good enough for many purposes.

V. William Taylor, Wantage

The rain in Spain falls mainly on my Brain

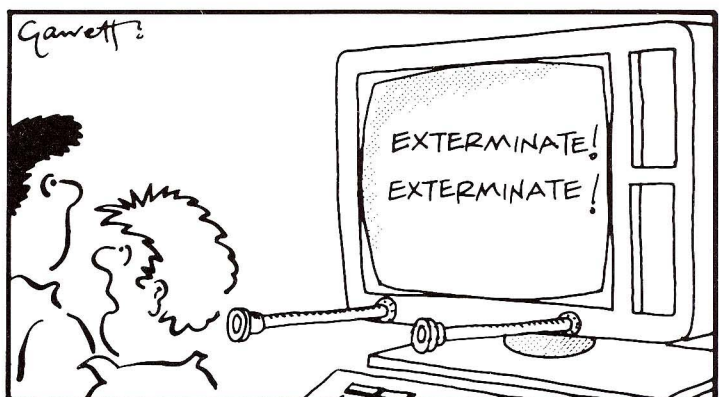
▲ Normal draft quality

The Rain in Spain falls mainly on my Brain

▲ And the intermediate "draft + bold" print

The rain in Spain falls mainly on my Brain

▲ "High Quality"



"NOT EXACTLY USER-FRIENDLY, IS IT?"

Cheap Mailmerge

LocoMail, the official mailmerger package for LocoScript, will set you back £39.95, but you can duplicate its 'Fill' command for free.

'Filling' is the process of going through a form letter inserting special lines such as names and addresses, although you do it by hand. Proper mailmerging takes the data from a file stored on disc. Filling is suitable for short mailshot runs, where it would be to much trouble to set up a data file for a full mailmerge.

So, with ordinary LocoScript, type the basic letter into a document. Place a 'Unit' marker by typing [UNIT] where you will later want to insert names and so on, and save the document when all set up.

To do the fill process, edit the file and press the [UNIT] key (which is [SHIFT] + [PARA]). The cursor now moves to the next 'unit' mark it finds, and you just type the relevant text to be inserted. Press [UNIT] again to get to the next mark, and so on until complete. Save the document, print and repeat.

A small catch is that when you put a 'unit' marker into your letter, it will end the line automatically as if you had typed the [RETURN] key. As a result, this method is best suited to marking the slots for addresses, dates and titles, where you want them to start on a new line anyway.

N Headley, Hampton

INPUTs without question marks

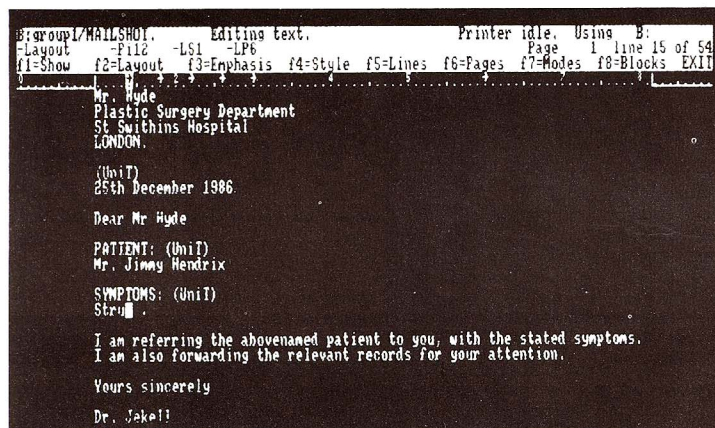
The INPUT statement in BASIC is pretty useful as a way of asking questions and getting replies from the user as a program is running. For instance, to get someone's name you might use the program line `10 INPUT "Name"; name$`. This then prints 'Name?' (note the question mark is added on automatically) on the screen, waits for the user to type their name and press [RETURN], and stores the result in the variable 'name\$'.

However, there are some useful variations to this method.

1) To avoid having a question mark automatically stuck on the end of the question, use a comma instead of a semicolon after the prompt text. So `INPUT "Press [RETURN]"; zzz` puts up on the screen an instruction to the user 'Press [RETURN]', and waits until the user does so. As a side effect, the variable zzz gets set to zero, but you can ignore that.

2) To avoid the business of pressing [RETURN] at all, you can resort to a different technique. If you wanted to tell the user to 'press any key to continue', you can use these lines:

```
zzz$ = ""
PRINT "Press any key to continue"
WHILE
zzz$ = " " : zzz$ = INKEY$ : WEND
```



▲ How a simple letter might look during filling

Spool and NewWord

Another useful program on the 8000 Plus offer disc is Spool, which allows you to print files while you are running other CP/M programs.

Normally, you have to wait for a file to finish printing before you can do anything else.

Although Spool works fine with the vast majority of programs,

including word processors like WordStar, as it stands it will not work with NewWord, the popular word processing package from NewStar. This is a 'feature' of NewWord, which is easily remedied:

► Before going any further, make very sure that you have copied your NewWord master disc onto a backup as described in the manual. You are about to alter the

NewWord program files, and if things go wrong you will need to revert to your backup copy.

► Take your NewWord working disc with the file NW.COM on it, and copy that file to the M drive (with the command `P I P M := NW.COM [RETURN]`)

► Take your NewWord disc which has the file NWINSTALL.COM on it and type `NWINSTALL [RETURN]`.

► You are now being asked for the name of the file to install: type `M : NW [RETURN]`

► Press [RETURN] again, which means that the modified version will be stored straight back to NW.COM rather than any other file

► Now you are at the main menu screen. Press H

► This takes you to 'Patch Menu #1'. Press 3

► At Patch Menu 3, press U. Then press X twice to finish the modifications.

► You are faced with a final question asking you whether you want to change the current options – answer N to this, and press [RETURN]. Now the amended version of NW.COM is saved, and you are ready to go.

All changes have been made.

Newword is now installed for...

AMSTRAD PC16256
Epson MX80/100

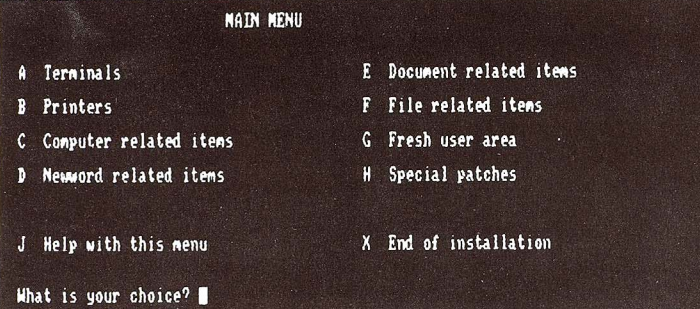
Do you want to change this? Y/N N

N>

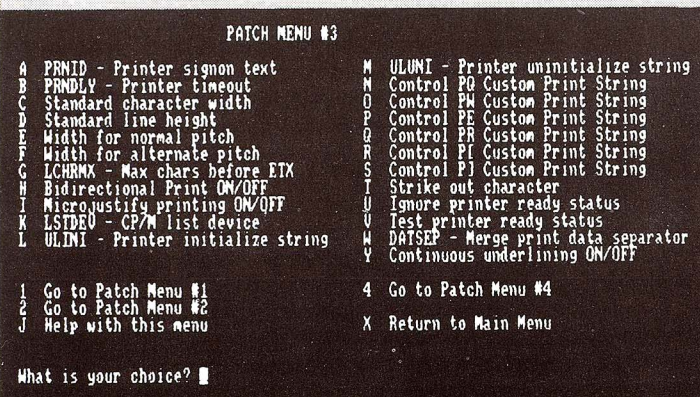
▲ And the final exit message once you've finished

Copy this version of NW.COM back onto your NewWord program disc. You should find that NewWord now lives in perfect harmony with Spooler. Of course, once you've gone through the process just described the changes are permanent, and you don't have to do it again.

If things haven't gone to plan, you might have made a mistake in the installation. Go back to your old copy of NW.COM, copy it and try the changes again. Only be sure to keep a backup of the original NW.COM at all times.



▲ NWINSTALL's main menu screen



▲ Patch menu #3, where you choose option U

Multicolumn printout from LocoScript

If you produce newsletters or other circulars, you probably want to print out text in two columns as is customary. LocoScript doesn't (at the moment) have a command to do this, but with a bit of guile you can do it in a roundabout way.

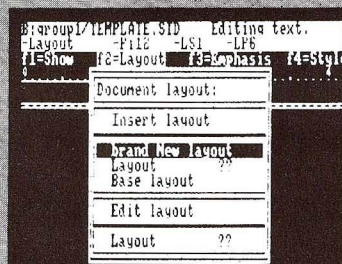
Suppose you are using standard A4 paper with proportionally spaced text. If you were doing a normal letter, with text in one column across the paper, you would use margins of roughly 10 and 89, giving 80-wide text with 2cm margins free either side.

Since 80-wide text takes up the page normally, to fit two columns and a gap between them into the same space the columns ought to be about 38 wide, leaving a gap of 4 spaces between them. So, the first thing you have to do is set up two LocoScript 'Layout's: one must have a left margin at 10 and a right margin at 47, and the other is to have a left margin at 52 and right margin at 89. Columns 48 to 51 are therefore the gap in the middle.

It is best to reserve a special Group on your LocoScript disc for documents to be in multicolumn format. Go to that group and start editing the TEMPLATE.STD document. To set up the Layouts, press the 'f2=Layout' key and then [ENTER], which picks the 'Brand New Layout' option. You are now dropped into the 'Editing Layout' screen - look at the top to see which layout you are editing; probably it will be number 1.

Press the cursor down key to get

to the ruler line, then the cursor left and right keys to put the cursor at 10. Set the left margin by pressing [f1]. Similarly, set the right margin at 47 by moving the cursor there and pressing [f2]. Now press [EXIT] to store this layout.



▲ About to set up one of the Layouts

From the editing screen, repeat the whole process to get a second layout with a left margin at 52 and right at 89. Again, make a note of the number which LocoScript assigns the new layout - probably '2'.

Now you are set to go. At the start of the document to be multicolumned, make sure you are using your first Layout (with the margins at 10 and 37). Type in the whole document normally. Then, once you've finished all the fine editing and are about to print it out, go to the head of the second page and insert the second Layout (with the margins at 52 and 89). At the head of page three, revert to layout 1, and so on ... each odd page having the first layout and each even page having the second layout at its top.

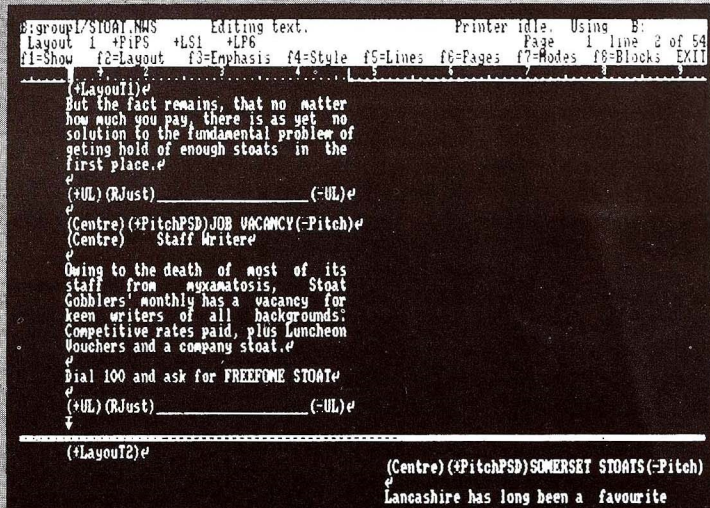


▲ The first layout you need just prior to saving it

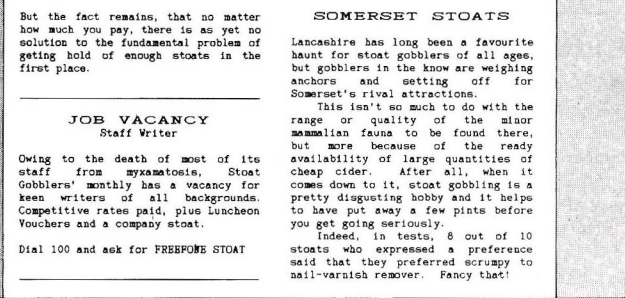
Now you are set to print. Save the document normally and print it. It is best to use single sheet paper for this; put the first sheet of paper in the printer, and when it is finished and ready to print page 2, put the same piece of paper in again. The second page, because of its special margins, is printed in a separate column to the right of the first one. Now do pages 3 and 4 together on a fresh sheet, and so on.

This is a very workable method. The only snag is that you have to be accurate about positioning sheets of paper in the printer exactly the same each time, or you will find the column gap varies a lot, or that the text lines on the two columns are not properly aligned. Practice makes perfect.

Charles Williams, Stanmore



▲ LocoScript's editing with a document for two-column printing
▼ And the results of printing it out



Using SHELL

One of the programs on an 8000 Plus subscriber offer disc is Shell, a utility which allows you to run CP/M programs fairly painlessly from LocoScript-style menus.

To start Shell going, you type Shell and a menu of all your files appears on the screen. When you run a program, the PCW leaves Shell, runs the program chosen, and when it has finished runs Shell again from scratch. This has one potential drawback, which is that you must have the file SHELL.COM on every disc you use, otherwise Shell cannot restart on a program's completion.

► There is a way around this though, which is to use the M drive. Assuming you are at the A> prompt, here is what to do:

► Find your CP/M working disc with PIP.COM on it, put it in the drive and type PIP [RETURN]. After some whirring, you see an asterisk prompt appear.

► Put your disc in the top disc drive,

the side containing SHELL.COM facing the screen. Type

M: =SHELL.COM [RETURN].

► Press [RETURN], which gets you back to the A> prompt.

Now to run Shell, type M: SHELL (and, as ever, press [RETURN]). Even if you swap discs in drive A to one which does not have SHELL.COM on it, the program will carry on running and re-running happily.

For experts, you can go further and make Shell part of your startup routine. Copy the file SHELL.COM to your CP/M start of day disc. If you haven't already, also copy to this disc the file PROFILE.ENG, and then edit it using RPED. As you will see, it contains a SETDEF command and a series of PIP commands which copy commonly used utilities into the M drive.

Insert a line in the PIP section (the block of lines beginning "<") which instructs it to copy SHELL.COM too. This command is <M: =SHELL.COM[0] Finally, add a final line to the end of

the file which is M: SHELL, save the file and be sure to rename it as PROFILE.SUB (use the REN command).

Now every time you start CP/M with this disc is will automatically go into Shell's user friendly menu. Liam Greenslade, Manchester

Automatic WRDCOUNT

It is possible to prepare a disc which, when you start CP/M up, automatically runs a BASIC program of your choice. If you have typed in the WRDCOUNT program listing published in issue 1, this can make swapping between LocoScript and BASIC much easier.

Take a blank disc and copy onto it from your master CP/M disc J14CPM3.EMS, BASIC.COM and SUBMIT.COM. In case you are still frightened of CP/M, just start CP/M up and have your copy of the master

disc in drive A. Then type: PIP [RETURN] (you see an asterisk appear ...) M: =J14CPM3.EMS M: =SUBMIT.COM M: =BASIC.COM (Put your disc with WRDCOUNT.BAS into drive A ...) M: =WRDCOUNT.BAS Change discs so that your work-disc-to-be is in drive A instead. Type: A: =M: * * [RETURN]

Now you are back at the 'A>' prompt. Next, using the RPED text editor (see this month's pages 25-6) prepare a file called PROFILE.SUB containing the following line: BASIC WRDCOUNT If you are going to run a program other than WRDCOUNT, just substitute its name in place of WRDCOUNT in PROFILE.SUB. Now if you reset the machine by pressing [SHIFT] + [EXTRA] + [EXIT] and insert this disc, your Basic program runs automatically. James Willis, Alton

LocoScript's Character Set

The LocoScript manual has a set of keyboard diagrams at the start showing weird and wonderful characters which can be got at with various combinations of [ALT] and [EXTRA], but it can be hard to find the exact symbol you want at any time. Here is a more organised list showing the characters available to LocoScript users:

Maths Symbols

<	less than or equal	[ALT] <
≈	approximately equal	[ALT] =
≡	equivalent	[SHIFT]+[ALT] =
≠	not equal	[EXTRA] =
>	greater than or equal	[ALT] >
±	plus or minus	[ALT] -
	modulus bar	[EXTRA] fullstop
÷	division	[ALT] /
×	multiplication	[SHIFT]+[ALT] /
\	backslash	[EXTRA] %
↑	'to the power of'	[ALT] U
∴	therefore	[ALT] ;
⇒	is implied from	[SHIFT]+[ALT] <
⇒	implies	[SHIFT]+[ALT] >
∞	double implies	[EXTRA] 9
∞	infinity	[ALT] %
°	degrees	[EXTRA] 5
½	1/8 fraction	[ALT] 1
¼	1/4 fraction	[ALT] 2
⅜	3/8 fraction	[ALT] 3
½	1/2 fraction	(key by itself)
⅝	5/8 fraction	[ALT] 5
¾	3/4 fraction	[ALT] 6
⅞	7/8 fraction	[ALT] 7

French Characters

ç	o-cedilla	[ALT] comma
ç	capital o-cedilla	[SHIFT]+[ALT] comma
«	open quotes	[EXTRA] <
»	close quotes	[EXTRA] >

Greek Characters

α	alpha	[ALT] A
β	beta	[ALT] B
γ	gamma	[ALT] G
Γ	capital gamma	[SHIFT]+[ALT] G
δ	delta	[ALT] D
Δ	capital delta	[SHIFT]+[ALT] D
ε	epsilon	[ALT] E
θ	theta	[ALT] Q
λ	lambda	[ALT] L
μ	mu	[ALT] M
π	pi	[ALT] P
Π	capital pi	[SHIFT]+[ALT] P
ρ	rho	[ALT] R
σ	sigma	[ALT] S
Σ	capital sigma	[SHIFT]+[ALT] S
τ	tau	[ALT] T
φ	phi	[ALT] F
χ	chi	[ALT] X
ψ	psi	[ALT] Y
ω	omega	[ALT] O
Ω	capital omega	[SHIFT]+[ALT] O

German characters

ß	double S	[EXTRA] S
---	----------	-----------

Accents

(type the accent *first*, then the character to go under it. So to get e- acute, press [EXTRA]+6 and then type e)

´	acute accent	[EXTRA] 6
ˆ	circumflex accent	[EXTRA] 7
˘	grave accent	[EXTRA] 8
¨	umlaut accent	[EXTRA] 2
˜	tilde accent	[EXTRA] -

Spanish Characters

¡	open exclamation	[EXTRA] !
¿	open interrogative	[EXTRA] ?

Scandinavian Characters

Å	a-ring	[ALT] 8
Å	capital a-ring	[SHIFT]+[ALT] 8
æ	ae dipthong	[ALT] 9
Æ	capital ae dipthong	[SHIFT]+[ALT] 9
ø	o-slash	[ALT] zero
Ø	capital o-slash	[SHIFT]+[ALT] zero

Punctuation Symbols

¶	paragraph	[EXTRA] P
†	dagger	[EXTRA] D
•	bullet	[SHIFT]+[ALT] fullstop

Monetary Symbols

R	peseta	[EXTRA] #
¢	cents	[EXTRA] \$
¥	yen	[EXTRA] Y
f	florins	[EXTRA] F

Other Symbols

↑	up arrow	[ALT] U
↓	down arrow	[ALT] N
←	left arrow	[ALT] H
→	right arrow	[ALT] K
↔	double ended arrow	[ALT] J
⊗	arrow into paper	[ALT] I
⊙	arrow out of paper	[SHIFT]+[ALT] I
©	copyright symbol	[EXTRA] C
®	registered symbol	[EXTRA] R
™	trade mark symbol	[EXTRA] T

Remember that these special key combinations work by holding down [ALT], [EXTRA] or [SHIFT] while the second key is pressed, in much the same way as you use [SHIFT] to get upper case normal characters.

Richard A Cook, Poole

Cutting a dash

In David Langford's 'Dash it!' insert (November issue) he recommends two or three consecutive hard hyphens saved as a phrase, as an approximation to a dash. A different way of doing it is to save a single 10 pitch double width hyphen (+Pitch100)-(-Pitch) as a phrase. This way, you get a single, long, continuous dash.
Jonathan Baylis, London SE13

True random numbers

The snag with Basic's RND function to generate random numbers is that the resulting numbers are not truly random. However, Geoffrey Childs' tip in issue 2 on how to set and read the PCW clock can be adapted to generate more or less true random numbers from BASIC. Since the clock can be made to count in seconds, it can be used to generate a 3600 different numbers which only depend on the time that you read the number.

Every time you need a *really* random number, type RANDOMIZE PEEK(64504!) in the program. After this, the function RND will return a decent random number. For example, X=RND sets the variable X up to contain a random number.

Terry Dwyer, Loughborough

Problems with pitches

One of the nice things about LocoScript is the way you can change the character pitch (the size of the printed characters) anywhere in a document. In a tip in issue two, some margin settings were suggested to get 2 cm margins on A4 paper for all the different pitch sizes you can use.

These settings do indeed work, BUT only if they are set up in the 'Base Layout'. If you mix the margin settings within one document, they do not work. For example, create a document and set the Base Layout to 12 pitch, with left and right margins at 09 and 89 as suggested. Then type in several lines. Next, insert a new Layout and set the pitch to 15, with the new left and right margins at 11 and 110. Now type in some text and the text will scroll right off the screen and continue beyond the right

margin.

A similar thing happens for pitch 17, with its margins of 13 and 127 - when you print it out, only part of the text for 15 and 17 pitch appears.

What has happened is that the line lengths are being based on the margins defined in the base layout. The original right margin was 89, therefore the new margin of 110 will be extended 21 characters to the right of this, ie. off the right hand edge of the page. For 17 pitch, the right margin of 127 will be 38 characters too long, nearly half a page extra!

To sum up; beware of changing the margins and altering the pitch size at the same time. Unless you are using the same character pitch as defined in the Base Layout, you could get some funny effects.

Charles Hassell, London SW12
If you want to change pitches in mid

document, the best way is to leave the Layout settings well alone, and just use the 'F4=Style' menu to alter the pitch. This way, the right hand margin will be properly aligned on the printed page, although it may not seem so on the screen.

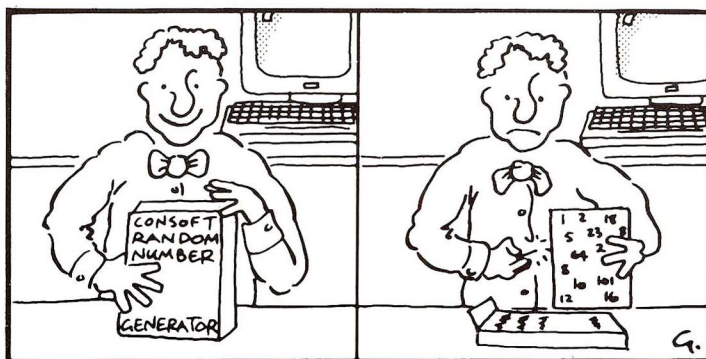
64,000 file question

For those who haven't found out yet, one side of a 3" disc in the PCW's top drive can store a total of only 64 files at any time. This total is regardless of how much space is free on the disc, and which groups are used for the files. After this point, a message 'directory is full' appears, and you will have to delete some documents.

Safe copying

Occasionally, when you are copying a particularly long or important file from one disc to another, it would be nice to reassure yourself that it has indeed copied properly.

If you are using CP/M, you can use PIP's little known 'Verify' option to do just this. All you do is add [V] to the end of the PIP command, with no spaces before it. For example, PIP B:=A:BOOK.DOC[V] copies your new novel from drive A to drive B, and then verifies that it has gone OK



LocoScript short codes

There are a lot of short cuts to 'doing without menus' in LocoScript. Many of these are shown in the mystical shorthand of the capital letter codes in the Set and Clear menus – see page 133 of the dreaded manual.

It's simple really! All you do is to type, for example, **SB** to get SubScript, and **SB** to cancel that particular code. Type fast enough, and the 'Set' menu won't appear at all, so this puts the code into your text very much more rapidly than using either the ordinary menus or even the Set and Clear menus. If you use the 'f1 Show' menu and tick the 'Codes' option you will see the codes appear as you type them in the document as normal.

If you press **SB** and forget what initials you need to type to get a code, press 2 on the keypad (or just wait a second) and the Set menu appears. Now you can see the right letters capitalised in the relevant choice – so **I** for italic, **RV** for reverse and so on.

Finally, to get a hard space you press **SB** (space bar). However, you can't get hard hyphens with **SB-**, which seems to be a 'bug' or a 'feature'!

Ray Gladden, Crewe

Running CP/M software

When you buy a new piece of software to run under CP/M, no matter how simple it is there always seem to be tortuous instructions in an obscure manual designed to prevent you using it.

The first thing you always ought to do is use DISCKIT to make a backup of the master disc as described in the Amstrad manual (although some programs like Prospell or certain games cannot be copied).

Secondly, put your newly copied disc in the drive, type **DIR** [RETURN] and you will see a directory listing of all the files on the disc. The files have names of up to eight letters, a dot, and three more letters. The final three letters tell CP/M what *kind* of file it is – .BAS means a BASIC program, and .COM means a CP/M command file.

So, if you have just bought a (fictional!) database called 'TurKey', on the directory listing you should see a file called **TURKEY.COM** or **TURKEY.BAS**. If .COM, the program will be run by typing **TURKEY** [RETURN], and if .BAS then by loading BASIC and typing **RUN "TURKEY"** [RETURN].

Unfortunately, you'll still have to read the manual to get any further.

Running BASIC files from CP/M

When you are faced with the awesome 'A>' prompt in CP/M and you type a name like **BURBLE** [RETURN], CP/M looks on the current disc for a file called **BURBLE.COM** and runs it.

However, there is a sneaky way of hacking CP/M so that instead CP/M will look for **BURBLE.BAS** and run that as a BASIC program – handy if you do a lot of BASIC programming.

You will need:

- ▶ One blank, formatted disc
- ▶ PIP.COM
- ▶ J14CPM3.EMS
- ▶ BASIC.COM
- ▶ SID.COM (side 3 of the master discs)

Using PIP, transfer the above files onto the new disc, and then, type **SID J14CPM3.EMS** [RETURN] (Despite the name, nothing to do with British Gas shares). Now for the tricky bit: very carefully type the following instructions in (pressing [RETURN] at the end of each line). If you get any of them wrong, type [ALT]C to abandon SID and get back to CP/M, then re-enter the SID command and keep trying until you get everything right:

```
S5d47
"BAS
.
S5ca2
"BASIC COM (3 spaces between
BASIC and COM)
.
S59dc
"EXEC.BAS
20
.
W J14CPM3.EMS
[ALT]C
```

What this has done is to hack your CP/M startup file (the .EMS file) so that whenever it would

normally run a .SUB file it looks for a .BAS file instead. Normally, when CP/M starts it looks for a file called **PROFILE.SUB** and, if it is there, automatically runs the command **SUBMIT PROFILE**. Your new version of CP/M instead looks for a BASIC program file called **EXEC.BAS**, loads BASIC up and runs it.

For example, take an existing BASIC program – maybe a simple one-liner like

```
10 PRINT "hello" –
copy it onto your modified disc and rename it to be EXEC.BAS. Now reset the PCW with [SHIFT]+[EXTRA]+[EXIT], and you will see EXEC.BAS automatically run in front of your eyes.

```

Now for the final frill. Make sure you have the file **SETDEF.COM** on your current disc and type **SETDEF [ORDER=(SUB,COM)]**

As long as you used your modified CP/M disc to start the PCW with, when you type at the 'A>' prompt something like **TEST**

[RETURN], CP/M runs the BASIC program **TEST.BAS**. Ordinary

One word of warning: with your modified CP/M system you cannot use **SUBMIT.COM** or the **PROFILE.SUB** facility – this is why you must keep the unmodified CP/M work disc around so that you can go back to it if necessary.

S McCall, Corby

```
A>h:sid j14cpm3.ems
CP/M 3 SID - Version 3.0
NEXT MSIZE PC END
A100 A100 0100 DAFF
#55d47
5D47 53 "BAS
5D4A 20 .
#55ca2
5CA2 53 "BASIC COM
5CAD 1A .
#559dc
59DC 50 "EXEC.BAS
59D4 53 20
59F5 00
#W j14cpm3.ems
0140h record(s) written.
#TC
#TC
A>|
```

▲ Using SID to modify CP/M

```
CP/M Plus Amstrad Consumer Electronics plc
v 1.4, 61K TPA, 2 disc drives, SIO/Centronics add-on, 368K drive M:
A)SETDEF [ORDER=(SUB,COM)]
Search Order - SUB, COM
A)test
Mailard-80 BASIC with Jetsam Version 1.29
(c) Copyright 1984 Locomotive Software Ltd
All rights reserved
31597 free bytes
hello
ok
test
10 PRINT "hello"
ok
|
```

▲ And a TEST.BAS file running straight from CP/M

Countdown

If you are writing long documents like books, you will probably be storing them in separate chunks since LocoScript doesn't handle screens of text very fast. This means that you have to jiggle the page numbering so that the second chunk begins one page after the first chunk ends, and so on.

A quick way of finding out how many pages a section has without actually editing it is to 'print' it. Press **P** for print, and select the 'Print some pages' option. The menu now tells you what the last page in the document is numbered as. Press [CAN] to cancel the print command. Now you can set the first page number for the next section correctly.

Angela Roger, Stirling

More SuperCalc print styles

People seem to be having a lot of trouble getting SuperCalc to produce high quality ('NLQ') print. After getting it going with **SC2** [RETURN], you can change between draft and NLQ at will with the [PTR] key. To change, press [PTR], use the cursor right key to highlight "high quality", press **SB** and [EXIT].

If you want to change the style of print, to get italics or whatever, give the /O command for 'Output' and as normal choose **D** or **C** for 'Display' or 'Contents'. But before choosing **P** for 'Printer' choose **S** for 'Setup' – you then enter **S** again for setup codes and the following can be used to change the style of

the spreadsheet you print out:

Condensed text	[ALT] 0	(on)
	[ALT] R	(off)
Bold	[EXIT] G	(on)
	[EXIT] H	(off)
Enlarged text	[ALT] N	(on)
	[ALT] T	(off)
Italic	[EXIT] 4	(on)
	[EXIT] 5	(off)
Underline	[EXIT] -1	(on)
	[EXIT] -0	(off)
Pica text	[EXIT] P	
Elite text	[EXIT] M	
Superscript	[EXIT] S0	(on)
Subscript	[EXIT] S1	(on)
Both of these	[EXIT] T	(off)

Will Parfitt, London SE22

Bits of WordStar

The listing published in the December 8000 Plus to make WordStar files readable works fine, but is a slight case of overkill. The objective is to copy all the characters of a WordStar/NewWord document file to a new file, without their top bits set. This converts the file from an illegible document to a plain ASCII file suitable for TYPEing or printing.

PIP can do the job very easily with its [Z] option. To view a WordStar document called filename on the screen, type PIP CON:=filename[Z] and to send the file to the printer PIP LST:=filename[Z] GR Yorke, Newcastle, Staffs

Tricks of Headers and Footers

The rules for placing headers or footers on solely the first or last pages of a document with LocoScript can be very confusing.

Using the 'Pagination menu' you can set 'First page differs' for headers and footers, and then you go to set up the actual header and footer text. In this 'Editing Pagination' screen, there is a section marked "End of footer 1: used only for the first page", but if you put a footer in here and then print out a document of only one page, the footer will not appear.

If you read the Addendum to the LocoScript manual very carefully, you will find a paragraph which reads "LocoScript arranges that the Header text on a single page document will be the one for a first page, and its Footer text the one for a last page."

The moral is then, for single page documents, either put the footer for the page in the section marked 'End of footer 2: used for all pages except the first', or to use the 'All pages same' pagination instruction.

Mrs. H. Hammond, Bingley

Logo 'FILL' command

DR Logo as featured on the PCW has a hidden command which is not listed in the Amstrad manual - a command to fill an enclosed area with shading.

To use the command, just put the turtle inside the blank enclosed area that you want to fill. Make sure the pen is down (with pd), type fill and press [RETURN]. You can even erase an area of solid white by having the pen in erase mode (type pe) and then filling, but it does totally erase the entire shape.

Dean Rossiter, Wellingborough

Margin notes

If you have to type up the script for a speech or a brief, you need to leave a wide right hand margin into which you can place notes, references or instructions for switching visual aids on and off.

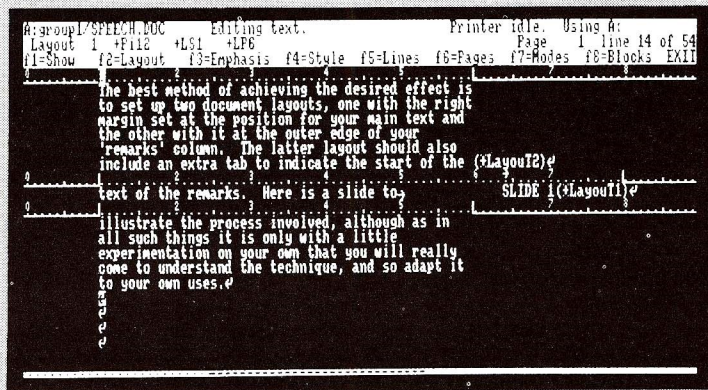
There is no simple way of writing text in multicolumn format, but you can do occasional margin notes without too much trouble. Suppose that you normally write text between a left margin of 10 and a right margin of 80, but you want to leave a space at the right for notes. You need to have one format for writing text between columns 10 and 60, and one for notes between columns 64 and 80 (leaving a small column gap).

While editing your document, use [f2] to create a Brand New Layout with the margins at 10 and 60 as desired. Then create another

new Layout with the margins at 10 and 80, and a Tab marker in column 64.

Type in all your text using the first (narrower) layout, and then go through finding the points which you need to have margin notes for. On each line to be annotated, type a [RETURN] at the end of the line above, insert the code for the wider layout, type a TAB at the end of the next line to take you into the note column, type the note, enter the code for the narrower layout again and press [RETURN].

Since putting these [RETURN]s at the end of lines stops them being justified as they would if they were part of a paragraph, your document will look best if you have justification turned off throughout. Alexander Deuchar, Bushey



▲ This ...

The best method of achieving the desired effect is to set up two document layouts, one with the right margin set at the position for your main text and the other with it at the outer edge of your 'remarks' column. The latter layout should also include an extra tab to indicate the start of the text of the remarks. Here is a slide to illustrate the process involved, although as in all such things it is only with a little experimentation on your own that you will really come to understand the technique, and so adapt it to your own uses.

SLIDE 1

▲ ... prints out as this

Full disc trouble in LocoScript

If you are writing a document which takes up to 16k bytes onto a disc which only has 15k bytes or less to spare, you are in trouble. On trying to exit the document you will get a menu offering either 'Disc Management' or cancelling the operation. Cancelling returns you to the Disc Manager screen and loses the document for ever.

It is possible, if you select disc management, to move some other document into Drive M (with the

'f4=Move' key) to make room for the new document to be saved. Pick a document larger than the one you are currently editing, move it to the M drive and press [EXIT] to return to your beloved text, whereupon you can finish saving it in the normal manner. But you must remember to copy the document that you moved to the M drive back onto a proper floppy disc before you turn the PCW off. Ted Nield, Southampton

Coping with disc errors

Sometimes for inexplicable reasons the PCW resolutely refuses to believe that a disc you are asking it to read from is usable. Some typically friendly message like 'TRACK #1 - SECTOR #03 MISSING ADDRESS MARK' appears on the screen, and you break into a cold sweat as you have visions of three weeks' typing going up in smoke.

Often such messages do indicate serious trouble, and the data on the disc is gone for ever. But occasionally you are just the victim of a glitch. From CP/M, try what buffs call a 'warm boot', which means pressing [ALT]+C. This may rectify some disc errors. From LocoScript, press the 'f1=Disc Change' key from the Disc Management Screen.

If all else fails, turn the PCW off, restart it and try again. If the disc still won't register, start panicking. You could try contacting some of the PCW user groups to see if any of their resident hackers can repair spoiled discs.

David Smith, Paddock Wood

Thumbing a lift

When running HitchHiker's Guide to the Galaxy, the reference card provided is sketchy, to say the least, and to make matters worse the screen seems to overwrite itself all the time.

In fact, all the files are on the disc for it to auto-load and then to work correctly, but it won't, unless you do the following:

Format a fresh disc, and using PIP (not DISCKIT) copy all the files to the new disc. On a PCW8256, this means putting your CP/M disc in the drive and typing PIP [RETURN]. You get an asterisk prompt, so now put the HitchHiker master disc in, and type B : = A : * . * .

The drive will whirr, and you will be periodically prompted to 'Insert disc for drive B' and 'Insert disc for drive A'. Whenever you are asked to insert disc B, put your newly formatted disc in the drive, and for disc A put your HitchHiker master disc in. Finally, when the asterisk prompt returns, press [RETURN].

For 8512 owners, start PIP but this time put the Guide master disc in the lower drive, your new disc in the upper drive, and then type A : = B : * . * . No disc swapping is needed.

Now you have an auto-booting Guide disc. Reset the machine and insert the disc in drive A, and the game will start correctly without any screen overwriting.

Chas Nicholson, Woking

Moving and copying program lines

Although BASIC provides no function to move or copy program lines to another area of the program, it is possible to do this with the EDIT command. The same technique can also be used just to change the line number if mistyped at first.

Suppose you have just typed in line number 100, and you want to copy it to line 110. Type EDIT 100 [RETURN], and line 100 comes up on the screen. Now use the cursor and delete keys in the normal way to edit the line number to read 110 instead. Press [RETURN], and if you list the program you will see that line 100 has been copied to line 110. This is very handy if you are entering a long series of similar program lines, perhaps mathematical calculations where only a minus sign varies from line to line.

If you wanted to *move* line 100 to line 110 instead of copying it, you can now type 100 [RETURN] which deletes the original line 100 for you. Obviously, this process of moving a line is just the same as changing the number on a mistyped line without having to retype the whole thing.

David Gray, Wisbech, and others

A free typeface

For academics and others who need to produce miniscule footnotes in documents, there is a sneaky way of getting a small typeface for them.

The secret is to print footnotes in 17 pitch text and subscript mode. For LocoScript users this means inserting the codes (Ⓜ Pitch17)(Ⓜ SuB) before the text in question, and (Ⓜ Pitch)(Ⓜ SuB) afterwards.

In CP/M, you can get the same effect by sending the dreaded 'escape codes' to the printer. Type [ALT]P to start echoing text to the printer, then press [EXIT] followed by [ALT]+[SHIFT]+0

DUNCAN MINITYPE

I think that I have unwittingly unearthed another type size which was never acknowledged to be there. It would be very good liberally sprinkle footnotes about their pages.

In an attempt to find alternative typestyles and sizes when CP/M, without the nausea of going into Basic and typing horribl "LPRINT CHR\$(27);CHR\$(15);", I discovered how to manipulate esc CP/M and these are the results:-

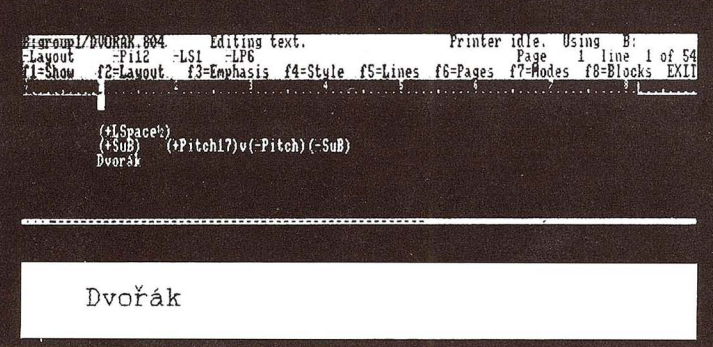
(the letter O, not the numeral 0). This gets you into condensed (17 pitch) text, so then type [EXIT] followed by S and 1 (digit, not letter this time) to get subscript

text too. When you've finished printing in small type, do a printer reset to get back to normal. JS Duncan, Peebles

Czechmate

Following the recent spate of correspondence in the letters pages over getting foreign accents out of LocoScript, you might be interested to see a way of printing out the Czech composer Dvorak's name correctly.

The accent on the 'r' is a subscript, 17 pitch lower case 'v' on the line above the 'r', with the line spacing set to 1/2. It might get a little tedious if you use it regularly!



Enveloping the truth

The PCW printer is not very good at printing on thick paper - 100 grams seems to be about the limit - and envelopes are particularly troublesome. However, with traditional, cheap 9x4 envelopes the real Hi-Tech answer is: 'let your fingers do the helping!'

Place the envelope in the feeding position, and as you operate the bail-bar gently push down on the envelope to help it round the roller. Reset the lever, straighten the envelope as necessary, and then start to print. While printing the address, gently pull up on the envelope to help the roller feed each line as it comes out. It works every time. JT Fynan, Hedge End

Copying LocoScript discs

If you have recently upgraded your PCW8256 to an 8512, you may well want to copy all the files currently on a CF-2 format disc to the larger CF-2DD format for the new double density disc drive. Four sides of drive A to one DD disc - great! But what a drag doing it with LocoScript's [f3] copy. DISCKIT can't copy a single to a double density disc, and using PIP is difficult and laborious on LocoScript discs.

You might like to assemble the following short program which auto-loads and will transfer or update all groups of LocoScript documents onto the same groups in drive B. Finally it displays a complete alphabetical list of the B disc directory.

First, using DISCKIT, format two discs, one in drive A and one in B. Then reset the computer ([SHIFT]+[EXTRA]+[EXIT]) and boot up LocoScript. Turn the disc over (to the CP/M side), press [f1], and copy the following files to the

M disc:

J14CPM3.EMS
 PIP.COM
 SUBMIT.COM
 DIR.COM

Insert the newly formatted single density disc back into the A drive and copy the four files back to group 0.

Now you must create two new files, also in group 0. The first is called BU.SRC ('Back-Up Source Code'). Erase the template ([CUT], [PAGE], [CUT]), and type the following with a [RETURN] at the end of each line:

```
PIP
<B:[G0]=A:*. *[G0]
<B:[G1]=A:*. *[G1]
<B:[G2]=A:*. *[G2]
<B:[G3]=A:*. *[G3]
<B:[G4]=A:*. *[G4]
<B:[G5]=A:*. *[G5]
<B:[G6]=A:*. *[G6]
<B:[G7]=A:*. *[G7]
<
DIR B: [USER=ALL]
Then [EXIT] and select Finish
```

Editing. On the Disc Management Screen, select [f7] (Modes) and Make ASCII file (choose the 'Simple text file' option). When asked for a group and name for the result, put it in group 0 with the name BU.SUB

The second file to create is called BOOT.SRC ('Boot-Up Source code'). Again remove the template part and type in

```
PIP
<M:=*.EMS
<M:=PIP.COM
<M:=DIR.COM
<M:=SUBMIT.COM
<M:=BU.SUB
<
M:
```

This time after [EXIT] etc. make a new ASCII file from BOOT.SRC in group 0 with the name PROFILE.SUB

Now to use it. Make sure the new single density disc is in drive A and reset the PCW with [SHIFT]+[EXTRA]+[EXIT]. It goes into CP/M, goes through its

doings and ends up with the prompt 'M>'. Change the discs for your LocoScript single-density disc in drive A, and the newly formatted double density disc in drive B. Type SUBMIT BU [RETURN] and watch it all happen. At the end, a directory of all the files comes up with how much space you have used.. If you want to add the contents of another A drive disc onto the same B disc, put the next disc in drive A and type SUBMIT BU again.

If things don't work, you have made a typing error in one of the two files. Go back into LocoScript, erase the existing .SUB files, amend the .SRC files and redo the ASCII file making.

Note that this process does not copy Limbo files. To copy them too, add another 8 lines to BU.SRC with G8 to G15 in the square brackets.

Bj Paterson, Leigh-on-Sea

PART II

CPM PLUS

A series of tutorial features taken from
the first five issues of 8000 Plus.

Use your operating system to keep your system operating!

How to avoid a slipped disc

Worried about doing your back up? Try this prescription for PCW health care from our very own Dr Ben Taylor MD FRCS MBE.

If you are anything like me, the first thing you did on unpacking your shiny new PCW was to totally ignore all the dire warnings about making working copies of your discs and to plough straight into LocoScript using the Amsoft master disc. Of course Amstrad don't actually provide any blank discs with the machine to enable you to follow their advice, but if you're still regularly using master discs without backup then you're taking a big risk. This month we're going to explain some of the often overlooked basics of working with floppy discs, and also throw in a few techniques to make running programs from CP/M a little easier.

Different types of disc

The PCW 8256 uses discs called CF-2 discs. These can be put into the disc drive label side outermost, but either way up. You can read and write to either side of the disc, and each side acts quite independently of the other, which means that you will have to format both sides of the disc separately before you can use a new CF-2 disc.

The PCW 8512 also uses CF-2 discs in its top drive (called "A"), but takes identical-looking CF-2DD discs in its lower drive ("B"). These discs can go in the B drive either way, but once you have decided which way then you can't use the other side – you might as well think of them as being single sided discs, and cross out the other side's label to remind you. The B drive can usually read CF-2 discs, but cannot write to them. The A drive can never read or write CF-2DD discs, so try and keep the types of disc separated in your drawer.

When you're faced with a blank disc, before you can use it you have to "format" it – this is like saying you have to rule the lines on a piece of paper before you can write onto it.

Getting your back-up

The warnings you see with software about keeping copies of all discs you use are not just put about by a cynical cartel of disc manufacturers out to boost sales. The golden rule is to ask yourself whenever you finish using your PCW, "If I couldn't ever use again the discs I am putting away now, do I care?" If you don't care, then fine, but otherwise you ought to make "back-up" copies of all your discs.

Losing your disc is far more common than you think; no matter how reliable any computer is, your pet Labrador might still run amok with your new bestseller novel disc. Or your elbow might have an argument with a cup of coffee. Or, a disc may just decide to develop a bad sector.

You're particularly at risk if you are working with discs created on someone else's PCW – a slight difference in

alignment in the drive heads on the two machines can cause write-fail errors. And don't be fooled into thinking that discs are safe from chewing by the PCW just because you have set the write protect tabs.

For discs that you've used to store LocoScript documents on, there is only one way to back up a disc, and that's using the DISCKIT program that comes on side 2 of the master discs. Start the PCW up (or press [SHIFT] + [EXTRA] + [EXIT] to restart it) and insert the DISCKIT disc in the top drive, with side 2 facing the screen. Now CP/M starts up, and you will see the distinctive "A>" prompt. Type DISCKIT [RETURN] (note the meaning – Disc kit), and what follows depends on whether you are using an 8256 or an 8512. DISCKIT gives you instructions as it goes along, so just follow those keeping a very careful track of which disc is which if you are copying from one to another. DISCKIT allows you to copy onto unformatted new discs, since it automatically formats them as it goes. Remember that if you format a disc you will lose all the data that was stored on it, and if you copy a disc then any data previously on the disc you are copying onto will be lost too.

If you are copying particularly valuable data, then after you've copied a disc you might feel happier if you verify it; select DISCKIT's VERIFY option, and it will check that the disc you have copied indeed contains the same data as the original.

Living in peace with your discs

All that has gone before is equally applicable to LocoScript users and CP/M users. Now we're going to look in some more detail at how to get the best performance from your discs using CP/M.

The PCW machines have, in addition to the floppy disc drives built into the monitor unit, a "RAM disc". This is a section of computer memory that is reserved inside the PCW to act like a floppy disc. Files can be stored and retrieved from it, and it is referred to by CP/M as "M:", as compared to A: or B: for the floppy discs. This means that on a PCW8256, you really have two disc drives, and three on an 8512.

Reading and writing on the M: drive is a lot faster than to the A: and B: drives, because there are no mechanical motors to run up to speed. By transferring your programs to M: and running them from there, you can speed up programs that make a lot of use of discs, and also cut out all that annoying disc swapping on a PCW8256.

Many commercial programs already make use of this facility automatically. If you've got a big package that still runs from ordinary floppy discs, then here's how to give it go-faster stripes:

Tech Note

The B drive has two disc heads, and stores data on both sides of a CF-2DD disc at once with only one disc directory for both sides. Formatting with DISCKIT processes both sides at once, so if you try to format the "other" side of the disc, you will erase the first side too, even if that side has the write protect tab set!

- Load in CP/M as normal, so that you see the prompt A> on the screen
- Clear out your M: drive by typing ERA M:.* [RETURN], and replying Y to the question "ERASE M:.* (Y/N)?" (Make very very sure you type M:.*, NOT A:.*, or you might lose your floppy disc contents – you can guard against this by setting the write-protect tab.)
- Find your CP/M work disc that has the file PIP.COM on it (the original is on side 2 of the discs that come with the PCW)
- Put it in the floppy disc drive, and type PIP [RETURN]. You will see an asterisk prompt.
- Remove the CP/M disc from the drive, and replace it with the floppy disc which contains the program you want to run.
- Type M: =*. The disc will whirr for a minute or two and the screen will tell you as it copies files across to M:
- When you see the asterisk prompt again, type [RETURN]. Now you will see the CP/M "A>" prompt again.
- Type M: [RETURN]. This tells CP/M that all your files are now in the RAM disc, and you will see the CP/M prompt change to "M>"
- Now you can remove the program's floppy disc from the drive. You should be able to use the program exactly as if it was running from the A drive. The floppy disc drive is now free for you to use purely as data storage, without being cluttered up with program files.

e.g. "A:MYFILE.DAT". Alternatively, many programs allow you to change your default disc drive; make sure your default is A: or B:.

WARNING!

Remember that when you turn the PCW off, all the files stored in M: will pass on to the great floppy disc in the sky.

Automating commands

The business of transferring all your program files from floppy disc to M: is well worthwhile, but can be tedious if you have to do it for a lot of individual files. Luckily there is a way in CP/M of doing it automatically using the SUBMIT utility.

Here's an example: suppose you have a database program called "DB", which you like to run from the M: disc. And suppose this database needs three files to run, DB.COM, DB.OVR and DB.ERR. Every time you run the program, you want to copy these files from A: to M: and then run the program. To do this, set up a file called, say, DB.SUB (it can be called anything you like as long as it ends ".SUB") which contains these lines:

```
PIP
<M: = DB.COM
<M: = DB.OVR
<M: = DB.ERR
<
M:
DB
```

(If you don't know how to create a new text file on your disc, see the PCW CP/M manual section 2.4). You will need to have the CP/M files PIP.COM and SUBMIT.COM on the same disc as the database files. Whenever you run your DB database, just type SUBMIT DB [RETURN], and you will see the commands automatically happening before your very eyes.

There are a couple of points to watch out for:

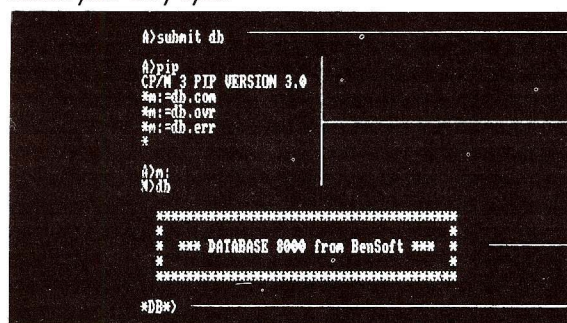
1. The PCW8512 has a large enough RAM drive to store an entire floppy disc from the A: drive, but the 8256 does not. On an 8256, if after you type "M: =*.*" in PIP it transfers some files and then says "ERROR: DISK WRITE NO DATA BLOCK", it means the RAM disc is full.

Your only option then is to try to cut out the non-essential program files from the floppy disc – the program manual should tell you which files are needed to actually run the program and which are, for example, simply for demonstration or installation. To transfer an individual file called FILE.DOC you would type (in response to the asterisk prompt in the PIP utility)

```
M: = FILE.DOC [RETURN].
```

If there are several files involved you should refer to the tip below titled 'Automating Commands'!

2. It is important to save any work you do on a real floppy disc, A: or B: rather than M: (otherwise you will lose your files when you switch off). Whenever your program asks you for a file name, prefix your file name by "A:" or "B:",



This is all you actually type in – and if you follow the advice in "Making an auto-start disc" you won't even have to do that.

All these commands are being printed on screen and executed automatically from the file DB.SUB

The program DB.COM comes up on screen.

Now you begin using the database normally.

Making an Auto-start disc

The SUBMIT utility has many hidden features which deserve a closer look. One that we'll take up for now though is the ability to have customised discs to start up a program at the beginning of a session without your having to do any typing at all.

When CP/M starts up from scratch, before it gives you its first "A>" prompt it has a look on the floppy disc to see if there is a file called PROFILE.SUB. If there is, then it automatically runs the SUBMIT utility with this file. So considering the example with running DB from a submit file, if we rename DB.SUB as PROFILE.SUB, then all the commands in it will be automatically run when you start CP/M with that disc in.

The only problem comes in how exactly to adapt your program disc so that CP/M can start up from it – this is called "making a boot disc". CP/M is contained in a file on your current boot disc called something like J14CPM3.EMS (the digits might be different depending on what version of CP/M you have), and you have to copy

this file to your working disc, by using PIP. This file is quite large, about 40k, so you might have trouble fitting it onto your work disc, particularly with PIP.COM, SUBMIT.COM and all your work files – if you can't cut down the number of work files to fit everything in then you are basically stuck, and you won't be able to make a boot disc very easily.

To sum up then, take your submit file, which we've called DB.SUB so far, and rename it by typing REN PROFILE.SUB DB.SUB and then make it into a boot disc by copying the .EMS file across from your normal start-up disc; you'll have to use the M: drive as temporary storage:

```
PIP M: = J14CPM3.EMS
      (put your work disc in drive A)
PIP A: = M: J14CPM3.EMS
```

Now, if you insert this disc when you first power up the machine, or reset it using [SHIFT] + [EXTRA] + [EXIT], your DB program will start up straight away without you having to type anything.



FRIENDLY KEYBOARDS

BEN TAYLOR explains how to customise individual keys to do your bidding.

Users of LocoScript are positively pampered by the PCW – all the keys on the keyboard are nicely labelled and set up to make life specially easy.

However, if you run programs using CP/M you know that things get a lot harder. In general, the f-keys and the arrow keys cannot be used, and at the same time many CP/M programs (like *WordStar*) use horrific choices of keys for commands, like [ALT] + KQ. Wouldn't it be nice to be able to use those spare keys to replace the complex multiple keystrokes?

Fortunately there is a way to do this on the PCW, which is to use the SETKEYS program that comes free on the CP/M master discs delivered with the machine. You can make all the unused keys on the keyboard do useful things, including automatically typing entire CP/M command lines.

The keys numbered

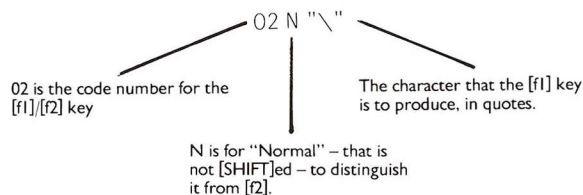
The principles behind altering the keyboard layout are quite simple really. Every key on the keyboard has a unique number associated with it, as shown in the diagram – so the space bar is key number 47, and so on. But the alphabet letter that the PCW links with each number can be altered, thereby changing the effect of pressing that key.

In fact, the most useful part of SETKEYS is that you can define a single keystroke to "expand" to more than one letter – it could represent an entire string of words or characters.

[EXTRA] + ½. Let's fix the keys so that the unused key [f1] produces the backslash character.

So, checking on the number diagram, [f1] is key number 02. Note that the number, 02, identifies the actual physical key labelled [f1]/[f2], and so the key number for [f2] would also be 02. To distinguish [f1] from [f2] we have to tell SETKEYS a bit more, namely whether the [SHIFT] key is being pressed or not.

Without further ado, here's the command that sets up [f1] as we want:

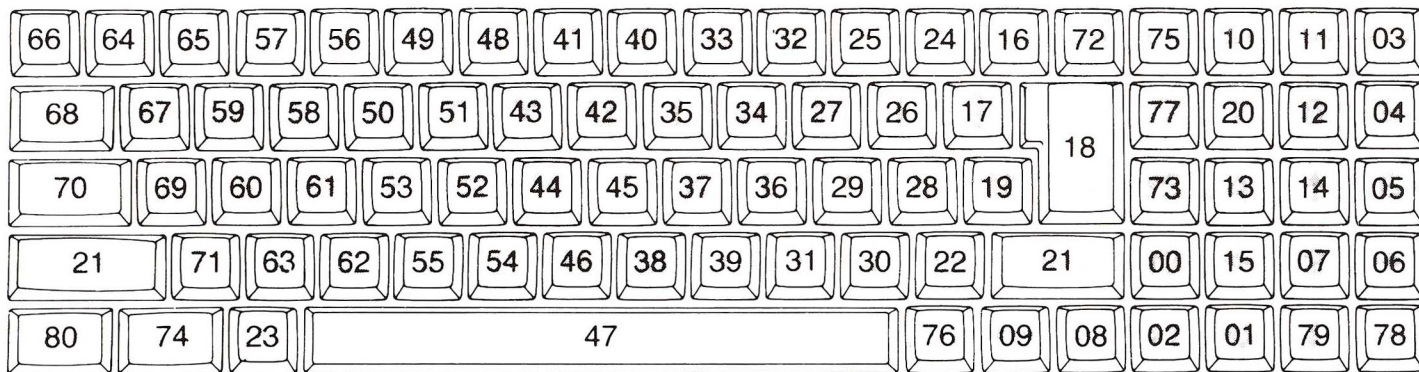


The next step is to get the SETKEYS program to read and act on this cryptic command. To do this, the command first has to be put into a CP/M text file (see box at end of feature), which for argument's sake will be called BSLASH.KEY.

Now for the exciting part. Find a disc with the file SETKEYS.COM on it (side 2 of the original master discs), and copy it onto the same disc as BSLASH.KEY is on.

Type
SETKEYS BSLASH.KEY

Now if you press [f1], a backslash appears on the



▲ The PCW keyboard, showing each key with the code that SETKEYS knows it by.

The key facts

The simplest way to explain is to go through a worked example. Suppose that a program you use often requires you to type a backslash, "\", as part of its commands. On the PCW, backslash is awkward to get, you have to type

screen. [f1] will produce the backslash character from now on until you reset the machine (eg. to run LocoScript), and it can be used inside any program. Characters produced in this way can be deleted and edited like any normal keystrokes.

You can of course set up many keys at the same time – you don't have to have a separate command file for each. Just put each command on a new line one after the other in the file that you give to SETKEYS.

Some final frills

Remember how you identify a key to SETKEYS by its number and its shift state – “02 N” means key code 02 (the [f1] key) used Normally (ie. unshifted). The “N” in there can be replaced by several things: “S” means [SHIFT], “A” means [ALT] and “E” means [EXTRA].

Here are a couple of example SETKEY commands, and by now hopefully you will be able to see where their translations come from:

- 1) 02 S "a" means: set the [f2] key (which is [SHIFT]+[f1]) to “a”
- 2) 78 A "b" means: set [ALT]+[ENTER] to be “b”

With a bit of effort you can convert your PCW keyboard to one that can enter all the commands you regularly use for your favourite programs at a single keystroke. You can of course put the SETKEYS command in your PROFILE.SUB command file to set the keys automatically when you start CP/M up.

It's a good idea to put sticky labels on the tops of your redefined keys, in case someone else unused to your conventions uses the machine.

Memory jogger

In fact, you can put any text you like on the end of the command line, and it is ignored by SETKEYS. This means you could type a comment to remind you what the command does: 02 N “\” (Set [f1] to \)

Boffin note 1

The full story is this: [RETURN] is ASCII code 13 (decimal), and M is the 13th letter of the alphabet, so we use !M to represent [RETURN]. All the ASCII code numbers 1 to 26 can be specified to SETKEYS in this way, offering the full range of control functions.

SPECIAL KEYS

Some characters, such as the “per thousandth” sign, cannot be typed directly from the keyboard in CP/M, but nevertheless can be displayed on the screen. To get at them, you have to know their ASCII codes. These are listed on pages 113-118 of the PCW CP/M manual – for example, the per thousandth sign is code 181.

Using ASCII codes in SETKEYS is broadly similar to ordinary keys. When defining [f1] to be \, the command was: 02 N “\”
To define it to be a per thousandth sign, the command is just 02 N “↑!181”

Note how the code, 181, is between

single quotes and preceded by ↑, but otherwise placed exactly where the backslash was in the first example. So putting this new command in a file and running SETKEYS with it will now define [f1] to print out a per thousandth sign on the screen.

Another possible problem crops up if you want to specify to SETKEYS that a key is to represent [RETURN] (a useful thing to do as you'll see). How you would go about this? – to set the [f1] key to be [RETURN], you want to have a command something like 02 N “ [RETURN] ”

But you simply can't type this! If you press [RETURN] after the open-quote,

the line will end and when you run SETKEYS it won't understand what 02 N “

means. So there is a special way to specify the [RETURN] key: you type ↑M instead (“↑” is [EXTRA]+[;] on the keyboard). To set [f1] to be [RETURN] then, you would type this line into your SETKEYS file: 02 N “↑!M”

This mightn't sound too useful, since no-one would want to change the [RETURN] key's placing, but it does have an extremely powerful function in helping to create whole command lines (see separate box).

COMMAND LINES

Getting keys to produce special single characters is useful, but it is also possible to produce entire lines of text at a single keystroke, although this requires a little more effort.

Suppose that a command you use regularly is ERA *.BAK, to delete your backup files and clear some disc space. You can do this using SETKEYS in a two stage process.

First, pick a number between 80 and 99. This will be the special number that will identify the text to be produced by the appropriate keystroke. So, picking 80 as our chosen number, the first command to give SETKEYS is E #80 "ERA *.BAK↑M"

This says to “expand” (hence the “E”)

the special code 80 into the letters ERA *.BAK followed by a [RETURN] character – as explained in the “special keystrokes” box, [RETURN] is represented by ↑M. Now all we have to do is tell SETKEYS which key will generate the special code 80, and this is done similarly to the backslash example. Suppose we want to use the keypad key “2” to do the file erasing for us, then put in the command file the line:

07 N “↑!#80”

07 is the code for the “2” key (at the centre of the cursor keys)

N is for Normal use – no [SHIFT], [ALT] or [EXTRA]

So, bringing it all together, create a file (we'll call it ERABAK.KEY) containing the two lines: E #80 "ERA *.BAK↑M"
07 N "↑!#80"

Now, back at CP/M type SETKEYS ERABAK.KEY. From now on, pressing the keypad key “2” will automatically type on the screen ERA *.BAK and, because of the final ↑M, execute the command as if you had typed it normally, ending with [RETURN].

80 is the special code to be generated. The quotes and arrows all have to be there – don't ask why!

Boffin note 2

In fact, the # sign introduces a hex code, so rather than talking of codes between 80 and 99, really the codes are from hex 80 to hex 9E. 9F means “ignore this code”.

CREATING A FILE

In this article we've talked a lot about creating text files of a few lines. If you're using LocoScript and its “Make ASCII file” option, this is very time-consuming and frustrating.

If you don't have a special CP/M text editor like WordStar, there is a quick way to create short files using the CP/M command PIP, which everyone has on their master disc. Put your PIP.COM disc in the drive, and type PIP filename=CON: [RETURN]

After the [RETURN], PIP waits for you to type the text to go into the file. Type each line, press [RETURN], and then hold down the [ALT] key and

press J to get to the next line. After the last line of the file, type [ALT]+Z, and you are returned to the familiar A> prompt.

Creating the file ERABAK.KEY, as used elsewhere in the article, and checking it by TYPEing it afterwards. ▼

```
B>pip erabak.key=con:
E #80 "ERA *.BAK↑M"
07 N "↑!#80"
B>type erabak.key
E #80 "ERA *.BAK↑M"
07 N "↑!#80"
```

If you make a mistake typing the file, you can't edit it with PIP, so you'll have to retype the whole thing. This is why it's only useful for small files.

And using it – after the SETKEYS, pressing the single key '2' on the cursor key pad produces the rest of the text. ▼

```
B>
B>
B>setkeys erabak.key
B>ERA *.BAK
ERASE *.BAK (Y/N)? y
B>
```

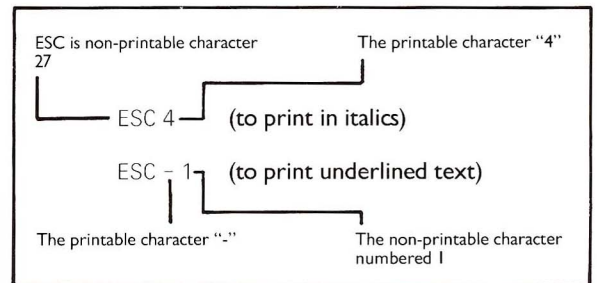
PRINTING WITH CONFIDENCE

The standard PCW printer is capable of outputting a wide variety of different text styles. LocoScript users are accustomed to being able to freely mix italic, bold, superscript and underlined text in sizes ranging from 17 to 10 characters per inch.

However, when using CP/M to print files, the options seem to shrink. Pressing the [PTR] key means you can control whether you use draft quality or high quality print, but other than that you seem to be forced to use 10 characters per inch standard text all the time.

But it isn't that bad really; with a little trickery, and knowing the black art of how to read the Amstrad manual, the printer can be made to use its full range of styles even from CP/M.

ordinary printable character. However, if there is a second following character, usually 1 or 0, then that refers



WHAT CHARACTERS ARE

Boffins out there will all know that characters are really the dreaded "bytes" that computer magazines always go on about. All this means is that computers only understand numbers, so each printable character is assigned a number between 0 and 255.

Not all the numbers between 0 and 255 correspond to actual printable symbols – after all, there are only 10 digits, 52 alphabet letters and a few punctuation marks. These are assigned numbers ranging upwards from 32. The numbers below 32 are assigned to non-printable or "control" characters. In particular, the character that everyone refers to as ESC is character number 27

to the PCW.

Since we're using BASIC as the way to handle characters in this article, it's as well to know how it copes with characters when printing. To send a printable character like "M" to the printer, the BASIC command is LPRINT "M"

However, ESC as we have seen is not printable but is known as character number 27, and the command to send that is LPRINT CHR\$(27)

This means, "send character 27 to the printer" – CHR\$ is BASICese for "character number".

The great escape

Consider for a moment what happens when you ask the printer to print a document. The PCW sends the characters down the connecting cable in a long stream, and the printer faithfully puts them on the paper. Now and again, the PCW sends a special character which doesn't produce any print but for example tells the printer to move to a new line.

It's these special characters that we are interested in. One character, known as "escape" or "ESC" for short, is very commonly used. When ESC is sent to the printer it means, "don't print the next few characters because they are an instruction to change print style". Exactly what the change is depends on the characters that immediately follow the ESC. For example, they might mean "print all text from now on in italics." In order to make the printer do what you want, you first have to find out what the correct command is that you have to send, and then you have to work out how to send it. The second part of this

is easiest done from BASIC, which has a special command called "LPRINT" to send characters to the printer.

Stealing the manual's secrets

Appendix II of the PCW CP/M manual gives all the information that you need to control the printer, although doesn't tell you what to do with it. Pages 126 to 137 list the functions that the printer does, and with each one is the code needed to activate that function.

For example, look at page 132: the top heading is "Italic text". Under that it says that to print italic characters you send ESC 4 to the printer, and to revert to normal you send ESC 5. This means that after the ESC 4, all text printed appears in italics until the ESC 5 is printed, after which it's back to the usual typeface.

Most of the special codes are in the form ESC followed by 1 or 2 letters or digits. These following characters can mean one of two things depending where they are, and unfortunately the PCW manual doesn't often distinguish between them.

The character immediately following the "ESC" is an to a non-printable character whose number is 1 or 0 respectively. For example:

See the box on "What characters are" for an explanation of printable/non-printable characters, and all will become clearer with some examples.

Doing the hard work

Armed with the information from the manual on what codes control which printer functions, you now have to get them from your head to the printer. The easiest way to do this is from BASIC. Run BASIC from your CP/M master disc, and it will give you its "Ok" prompt.

LPRINT is the command to send characters to the printer. Try typing

```
LPRINT "Hello from 8000 Plus" [RETURN]
```

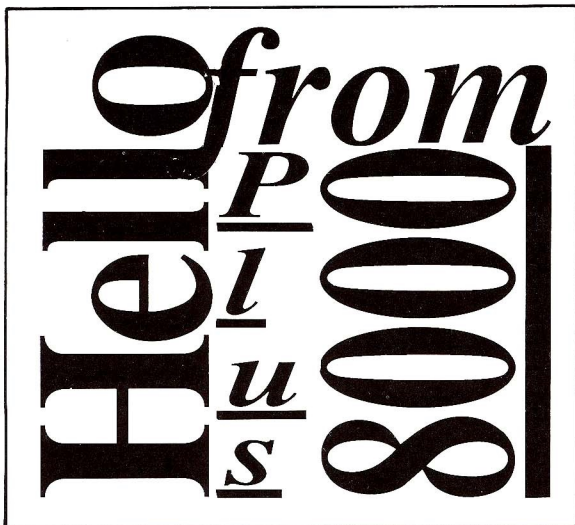
and the message appears in draft quality normal print on the printer. Now let's make it come out underlined.

Looking at page 133 of the manual, underlining is done by ESC - 1. The hyphen is a simple printable character, and the 1 is a non-printable code. ESC is non-printable code 27, so type

```
LPRINT CHR$(27)+"-"+CHR$(1) [RETURN]
```

Now try the "Hello ..." message again, and it comes out underlined.

Effects can be combined, so you can print in italics at



the same time as underlining. Again looking at the manual page 132 the required code for italic text is ESC 4. Type LPRINT CHR\$(27)+"4" [RETURN]

and then try the hello message again to see what happens.

You can reset the printer to normal with the [PTR] key (see the "In an emergency" margin note), so now try a few other effects from the manual like proportional spaced text, or superscripts.

The settings that you do this way using BASIC will stay in effect until you reset the printer, even if you leave BASIC. For example, if you set up italic text and then enter your CP/M word processor or spreadsheet, files printed from there will appear in italic text.

Unfortunately, many CP/M programs set the printer up the way they want for printing. For example, no matter what text size you start off in, NewWord will normally print in 10 pitch. Also, SuperCalc resets the printer before it starts, so you always get draft quality text out of it.

Saving your changes

As experienced users of the PCW printer well know, the paper is prone to jam as it goes through the rollers, or some other more human error means you want to reset the printer and start all over again.

Unfortunately, when you do reset the printer (from the [PTR] command line in CP/M) the settings you have made will all be lost, and it will revert to the standard draft quality, 10 characters per inch text that you normally get. It's very awkward to go into BASIC and redo things.

The solution to this is to send the printer the command ESC d, which freezes the current state of the printer as the one to be used after a reset is done. So, after you have made all the changes you want to the normal print settings, just before you leave BASIC type LPRINT CHR\$(27)+"d"

Make sure you type a lower case "d", or the command will be ignored. Now you can reset with impunity, without losing your hard work.

The exception to the rule

Most of the printer codes in the manual are in the form "ESC p l", and we've covered use of these in some detail. But some, like that on page 131 for Condensed text (17 pitch) just says "SI". Just to the right of SI is the number 14 in brackets - this is Amstrad's way of telling you that SI is a non-printable character whose code number is 14.

So to change to 17 pitch text, just send the single special character SI to the printer. Do this from BASIC by typing LPRINT CHR\$(14) [RETURN]. There's no need to send ESC or any other characters.

```

CP/M Plus Amstrad Consumer Electronics plc
v 1.4, 61K TPA, 2 disc drives, SID/Centronics add-on, 360K drive M:
)basic
Mallard-80 BASIC with Jetsan Version 1.29
(c) Copyright 1984 Locomotive Software Ltd
all rights reserved

31597 free bytes

Ok
LPRINT "Hello from 8000 Plus"
Ok
LPRINT CHR$(27)+"~"+CHR$(1)
Ok
LPRINT "Hello from 8000 Plus"
Ok
LPRINT CHR$(27)+"4"
Ok
LPRINT "Hello from 8000 Plus"
Ok
system
)

Hello from 8000 Plus

Hello from 8000 Plus

Hello from 8000 Plus
    
```

The printer commands being used and the effects they produce

To sum up then, the rule for reading the printer commands from the manual is this: single characters like "p" or "l" are to be sent to the printer as they stand, using LPRINT from BASIC as described earlier. Words like "SO" and "DC4" stand for special non-printable characters whose number is printed alongside them when they are first referred to in the manual. Don't try to send the characters "S" and "O" to the printer, but send SO's number, 14, instead, using CHR\$(14) in BASIC's LPRINT command.

In an emergency

All this changing print styles can make you end up with a confused printer that can't just print normally. To get back to the usual setup, press the [PTR] key and you will see the printer status line at the bottom of the screen. Use the cursor-right key to highlight the RESET option, and press This resets the printer, and [EXIT] gets you back to CP/M.

AUTOMATING THE PRINTER CHANGES

Typing all these commands from BASIC works fine, but is very laborious. There is a CP/M utility called SETLST which does it faster, but is harder to set up.

It works in a similar way to the SETKEYS program covered in the last issue of 8000 Plus. Set up a simple text file containing the characters to be sent to the printer, and SETLST then sends this file to the printer, as will be revealed.

The only problem is how to type non-printable characters into this text file. The up-arrow key ([EXTRA] + ; on the keyboard) introduces a non-printable code, which is typed between single quotes, so non-printable character number 1 is typed as ↑'1'. Since ESC is used so often, this can be typed by its name, again between single quotes after and up-arrow: ↑'ESC'.

For example, suppose you always want to work in high quality, proportional spaced text from CP/M. Set up a text file called, say, HQPS.PTR, containing the lines

```

↑'ESC'p↑'1'
↑'ESC'm↑'1'
↑'ESC'd
    
```

The first line sends ESC p l, to set proportional spaced print, and the second line sends ESC m l for high quality text.

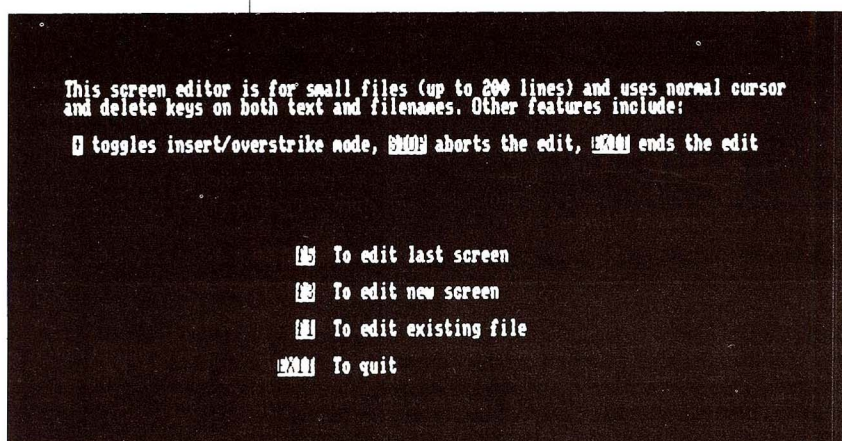
Now put a disc in the current drive with the file SETLST.COM on it and type SETLST HQPS.PTR [RETURN]. From now on, text going to the printer from CP/M (eg. when you print a file by saying PIP LST:=filename) will appear in high quality proportional spaced print.

The virtue of the last line of the HQPS file, sending ESC d, is that it saves the high quality, proportional spaced setting as the default. If you have to do a printer reset subsequently, after paper jam for example, then the printer will revert to this state rather than its usual draft mode 10 pitch text.

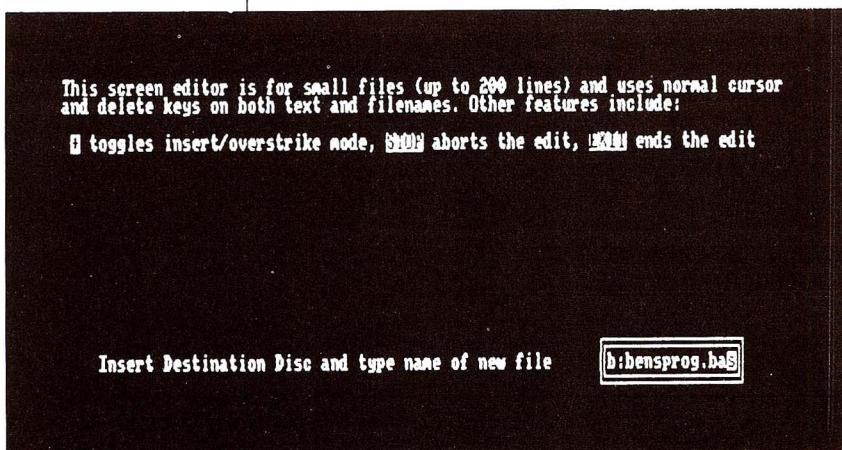
If you save HQPS.PTR on your startup disc, with SETLST.COM, you can use the PROFILE.SUB autostart method to set your PCW printer up automatically whenever you start CP/M.

LEARN TO BE AN EDITOR

The PCW's text editor RPED laid bare under the scalpel of skilled surgeon Ben Taylor



▲ The main menu screen of RPED



▲ Entering the name of the file to edit

If you use CP/M and programs that run from CP/M, you will have discovered a regular need to produce what are called 'ASCII files'. Don't be put off by the name – 'ASCII file' is just a confusing way of saying a plain text file. One of the most common occasions when you need to create an ASCII file is if you are trying to program. Since this issue of 8000 Plus is covering programming languages in some detail, it seems appropriate to discover the easy way to create the kind of files they need to work with.

Unfortunately LocoScript is singularly bad with ASCII files. It wasn't until a few months after its first release that it could even produce the things, and even now the time spent swapping between LocoScript and CP/M is so tedious that it is impractical to edit files that way. The answer lies in one of the free programs hidden on the PCW master discs – the RPED text editor.

Starting from scratch

To run RPED you will need to have a disc with the files BASIC.COM and RPED.BAS on it. You will find these on Side 2 of the original PCW master discs, but you should of course have copied them onto a work disc by now. Starting it going is simple: you just put that disc in the current drive and type BASIC RPED [RETURN]

BASIC starts up with its introductory burble, and after a short pause you are faced with RPED's master menu screen. Since we are creating a new file, press the [F3] key, for 'To edit new screen'. You will see a new prompt appear on the screen asking for the name of the file which will be used to hold your new text.

As it stands, the space for the filename is preceded by 'a:', meaning your file must go onto the A drive. If you want to store the file on B or M rather than A, you can change this; use the cursor left key (not the delete key) to get the cursor over the 'a', and just type 'b', 'm' or whatever you want. Now use the cursor right key to get back to the spaces for the name of the file and type it. Press [RETURN] or [ENTER] to confirm the filename, and the screen clears, leaving only a few cryptic prompts on the top line.

This blank screen is for editing, and the prompts at the top tell you how to use it (well, sort of). It's really very easy; just use the cursor keys to move the cursor to wherever you want to put some text, and start typing. Press [RETURN] when you have filled a line, and, as you'd expect, you're moved to the left hand end of the line below as you would expect. The two delete keys work as normal, and if you put the cursor in the middle of some text and start typing, the new text is inserted, shifting the remainder of the line to the right.

RPED is only a simple editor, and isn't designed for massive files. You can't create files longer than 200 lines –

USEFUL KEYS IN RPED

[I] switches you between insert and overstrike mode. As explained above, if you type new text in the middle of old text, the new stuff is inserted and the old shifted along. If you press [I], new text overwrites the old characters that used to be in those positions. [I] again changes back to insert mode. [ALT] ↓ inserts a blank line above the current line you are on, so you can type in new text.

[CUT] deletes the current line you are on and closes up the gap. [EOL] takes you to the end of the current line, and [LINE] (press [SHIFT]+[EOL] for that) takes you to the start of the current line. [RETURN] takes you to the start of the next line. [EXIT] saves the current file and returns you to the main menu screen. [STOP] aborts the edit altogether, and returns you to the main menu screen.

FIENDISH DEVICES

Ben Taylor finds out how the PCW controls serial and parallel interfaces for printers and other hardware

Since this issue of 8000 Plus deals with connecting up non-standard printers to the PCW, it seems appropriate to look at the facilities that CP/M provides for talking to external pieces of hardware.

For all its higgledy-piggledy design in many areas, CP/M does present a consistent and quite rational interface to the outside world, through what it calls 'devices'. Most software which needs to use the external hardware (such as a printer or telephone modem) will do all the device handling automatically, but it is useful to know what CP/M does, in case things go wrong.

Through the door

At the heart of the PCW is a mass of silicon chips, which by themselves are of little use – you couldn't send them instructions without a keyboard, and you couldn't see what was going on without a display screen. The channels by which any computer communicates with its user are very important.

First, an analogy. Imagine an office where there is a central open plan area, with doors leading off it. One door labelled 'Fred Bloggs' goes to your boss's office and one labelled 'Edith Snooks' to the managing director's office. When you began work, you were taught which door leads where.

However, one night the MD is caught embezzling funds and is sacked. Your boss is promoted to MD but doesn't want to change offices, so now you have to get used to the MD's office being labelled 'Fred' rather than 'Edith' and being through a different doorway. All memos, questions, decisions and so on are routed through this new doorway, but as far as the workers are concerned nothing has really changed; there is still an MD taking decisions through a doorway.

The insides of the PCW are much like this central open plan office. It has doorways to and from the world outside through which it sends its data, and these doorways can be changed around depending how you want your PCW to work.

The DEVICE command

Getting back to CP/M, these 'doorways' are the 'devices' mentioned earlier. There are two kinds of device – 'logical' and 'physical'. Back to the office analogy, think of the post of managing director as a logical device, but the person who is the MD at any time as a physical device. It doesn't matter *who* is MD physically, as long as someone does the job.

The PCW has several input and output tasks that need doing. It needs to receive its commands from the user, and display its replies somewhere, and it needs to have somewhere to send printer output. As a refinement there are also two auxiliary channels, one for input and one for output, which can be used if you need to read from and write to two things at once. These tasks form the 'logical devices'.

Now think of what hardware you have – the physical things you can touch, or 'physical devices'. There is a keyboard, a monitor screen and a printer. If you have bought an interface unit for a new printer or a modem, you will also have a Centronics (parallel) and a Serial interface at your disposal. It only remains to work out which physical device corresponds to which logical device.

Every device, logical and physical has a name. The logical devices are:

- CONIN: CONsole INput (where command lines are read from)
- CONOUT: CONsole OUTput (where responses to commands are displayed)
- AUXIN: AUXiliary INput (for receiving input from other computers)
- AUXOUT: AUXiliary OUTput (for sending output to other computers)
- LST: LiST device (where everything for printing is sent)

The physical devices are:

- CRT Literally, the Cathode Ray Tube. Means the screen if an output device is expected, or the keyboard if an input device is expected
- LPT The standard PCW printer (stands for Line PrinTer)
- SIO Serial Input/Output port if you have one
- CEN CENTronics output port if you have one
- NUL The 'null' device, ie. nothing at all

Within reason, any physical device can be associated with any logical device. The only restriction is that the physical device in question must be of the right type – you can't assign the printer to the console input, because printers can only deal with output. The command which tells the PCW which device goes where is, not unreasonably, DEVICE.

Find side 3 of your PCW master discs (titled 'Programming Utilities') and copy DEVICE.COM onto your CP/M work disc. Now try typing DEVICE [RETURN], and you will see a screen of information come up. This tells you which physical devices are currently assigned to which logical ones, as shown in the example, and gives you a chance to change them. For now, press [RETURN] to return to the A> prompt.

If you have a serial/parallel interface unit attached, you will see that the PCW automatically assigns SIO to both AUXIN and AUXOUT for communications. Otherwise they are assigned to NUL (the null device), as in the example.

```

A>device
Physical Devices:
I=Input,O=Output,S=Serial,X=Xon-Xoff
CRT  NONE  IO      LPT  NONE  O
Current Assignments:
CONIN:  = CRT
CONOUT: = CRT
AUXIN:  = Null Device
AUXOUT: = Null Device
LST:    = LPT
Enter new assignment or hit RETURN
    
```

Tells you what hardware you have available

The console input and output are both to the CRT (ie. keyboard and screen respectively)

The auxiliary ports are unconnected (no interface unit attached)

Listings will be sent to the ordinary printer

The printer can only do Output

The CRT can do both Input and Output

▲ The results of running DEVICE, with the normal startup assignments in force

Changing things around

Now we can use DEVICE to have some fun (if CP/M can ever be fun). As you now know, the standard PCW printer is a physical device known as 'LPT', and can only do output jobs. The PCW expects to be able to use an output device for its console output, and although this is normally the screen, it can be changed. Any device capable of output can do the job. Type

```
DEVICE CONOUT:=LPT [RETURN]
```

This directs all console output to the printer instead of the screen, so the screen appears to go dead and your CP/M session carries on as normal but on the printer instead. Note the direction of the command: it is always DEVICE logical device name=physical device name, and all logical device names have colons after them.

Type DEVICE CONOUT:=CRT [RETURN] to get



▲ Before and after a SETSIO operation

the screen back again. You can actually combine these commands by typing DEVICE CONOUT:=CRT,LPT [RETURN] which sends the console output to both the screen and the printer, providing you with a hard copy of your work session for later reference.

A particularly useless example is DEVICE CONOUT:=NUL, which just kills the screen dead. All commands you type will work, but the screen will not echo them – this is useful to programmers who want to prevent passwords being echoed on the screen as they are typed.

Sending files to printers

The most important use of DEVICE is to allow the PCW to send text for printing to printers other than the standard one. When the PCW prints a file it sends it to the device LST: (a logical device, remember). This is usually attached to the physical device LPT, ie. the standard printer.

To hook up a different printer, you will need to buy a serial/parallel interface for your PCW, which slots on to the back of the machine. This is because the standard printer is specially built, and other printers cannot be plugged directly into its socket on the PCW.

Slot the interface on while the PCW is turned off, and then start CP/M normally – as it starts up, CP/M looks around and registers the presence of the new interface on its back. Suppose your printer uses a Centronics interface, as most do, then to send printer listings to it just type

```
DEVICE LST:=CEN [RETURN]
```

You will see that in DEVICE's output the new physical devices SIO and CEN are listed, which weren't there when the interface box wasn't connected. From now on, all printed output goes down the Centronics line to the new printer, until you switch back by typing DEVICE LST:=LPT. If you are using a serial interface instead of the Centronics, use SIO instead of CEN throughout. So to print a file called TEST.BEN, type

```
PIP LST:=TEST.BEN
```

SETSIO and serial interfaces

While on the subject of connecting printers up through the serial/parallel interface unit, some more details on the serial interface are in order.

If you have a choice of using Serial or Parallel operation for a device (eg. a printer), choose parallel as it is much simpler to use. Communications lines need to use serial interfaces, but the commands for configuring the interface will be in the software itself – this includes Amstrad's own MAIL232 program on side 1 of the master discs.

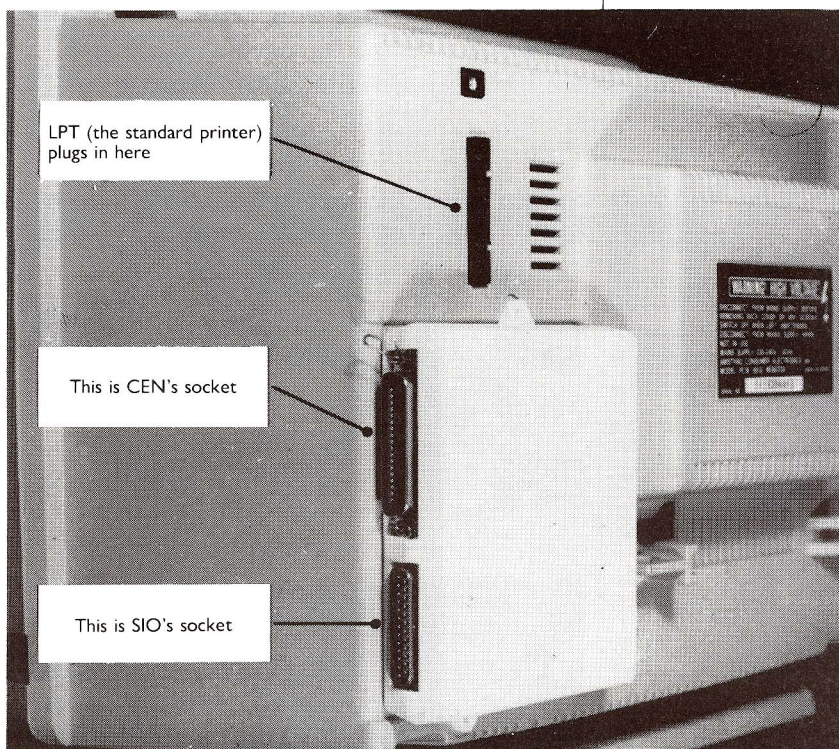
However, if you are forced to use a serial printer you will need to get to grips with SETSIO. Like DEVICE.COM, SETSIO.COM is to be found on side 3 of the master discs – copy it onto your work disc. Now inspect your printer manual to determine what the required characteristics of the line are to be. Printers often have a row of tiny switches ('DIP' switches) so that the you can choose your own characteristics. Suppose you, or someone else, have set the printer to

work at 2400 baud, 7 bits per character with 1 stop bit, no parity, and XON/XOFF protocol.

If you didn't understand that last sentence, you will see why you are well advised to avoid serial interfaces. This is not the place to explain such terms, but just make sure the words correspond to the printer manual's. Then type SETSIO 2400, BITS 7, STOP 1, XON ON, PARITY NONE [RETURN]

This sets up your PCW serial interface to match your printer's, so the two can communicate. If you get something wrong, you can change a single entry (like speed) by, for example, SETSIO 2400, which alters the baud rate to 2400 while leaving everything else unchanged. SETSIO [RETURN] by itself lists out the current settings of the serial interface.

Redirect the listing output to the serial interface with the command DEVICE LST:=SIO, and away you go.



▲ The back of a PCW, with an interface box, showing where the physical devices connect

A final catch is that if your printer cannot print, for example if it runs out of paper, and you try to send a file to it then CP/M will give an error like 'CEN not ready - Retry, Ignore or Cancel?'. Get your printer ready and press R for Retry – if you press C for Cancel, you will find that CP/M has assumed the Centronics line is broken and disconnected it. Typing DEVICE will reveal that LST has become attached to the null device, and you will need to do another DEVICE LST:=CEN before you can print again.

Simpleton note

Don't use a sledgehammer to crack a cliché – if you want to echo all screen output to the printer to record a session, use [ALT]+P from CP/M. DEVICE CONOUT:=CRT,LPT is a much harder way of doing much the same thing.

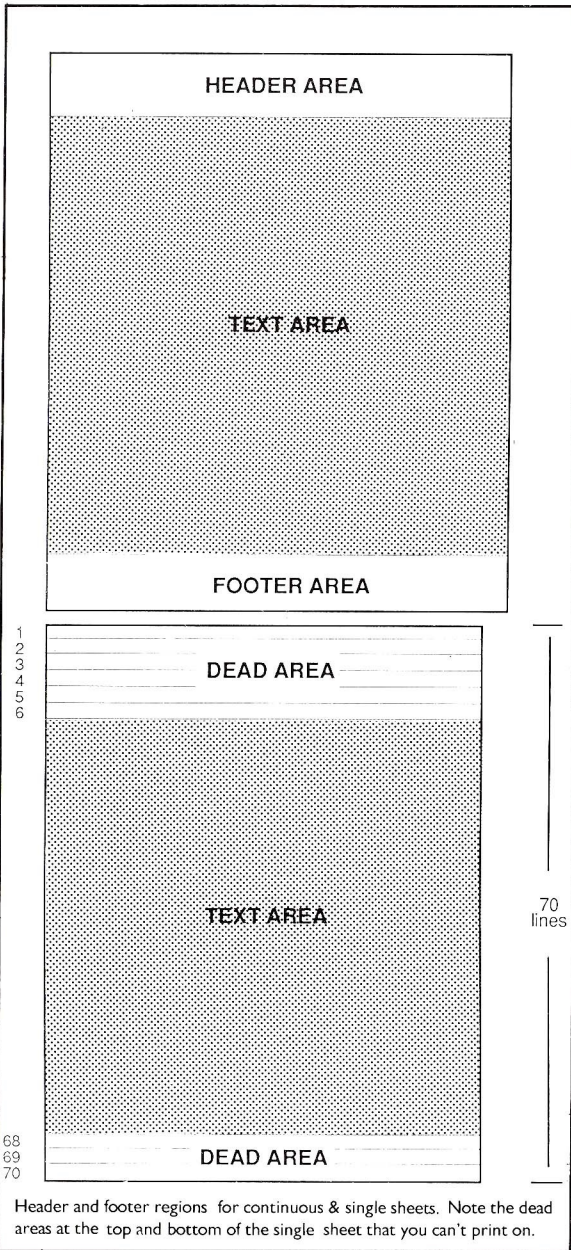
PART III

LOCOSCRIPT

A series of tutorial features taken from
the first five issues of 8000 Plus.

Anatomy of a page

The key to using headers and footers is to understand how LocoScript splits up the page when printing. As shown in box 1, there are three areas: the header (top of the page), the text area called the page body, and the footer (at, you guessed, the foot of the page). You can set the number of lines allocated to each of these areas to be anything you want, as long as the total number matches



the overall page length.

Single sheet A4 paper has 70 lines per page. The perforated continuous stationery that is used with the tractor feed mechanism on the printer can print 66 lines per page – this size is known as American Quarto (although it is occasionally possible to buy continuous sheet A4 paper).

The PCW printer can put text on any line on continuous sheet paper, but can't print on the top 6 lines or bottom 3 lines of single sheet paper. This is simply because, as with an ordinary typewriter, the paper feed rollers have to have enough paper to be able to grip the sheet properly. For this reason, to use headers and footers it is best to try and buy some continuous sheet paper, or else you won't be able to print them at the very tops and bottoms of pages. ▶

HEADERS AND

Nothing to do with football, just the production of nicely labelled documents. Ben Taylor negotiates one of LocoScript's trickier features.

FOOTERS

Most of the instructions in manuals and tutorials about LocoScript are aimed at producing simple letters of one page or so. However, LocoScript is capable of producing much more complex results, from esoteric professional documents to complete novels.

Once you progress beyond about two pages of text, it becomes important to make sure the page layout gives the document a consistent and attractive look – in particular, automatically numbering the pages is useful. This kind of thing is done using LocoScript's Headers, Footers and the Base Layout, and that's the area that we're looking at this month.

It really is well worth while taking the trouble to use headers and footers, although they have a justifiable reputation as one of LocoScript's more impenetrable areas. Early versions had bugs in, with the result that you couldn't actually do page numbering, but these have now mostly been ironed out.

LOCOSCRIPT

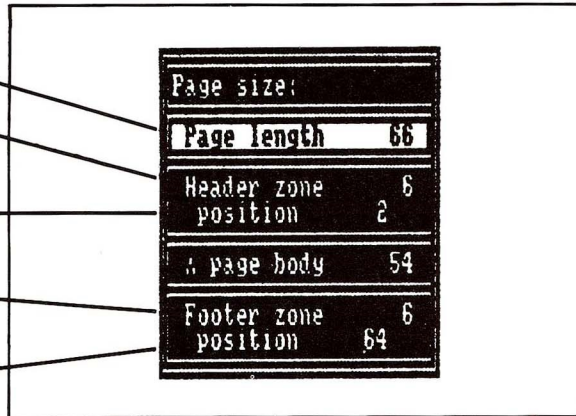
The page is 66 lines long (standard continuous paper)

Line numbers 1-6 (the top 6 lines) are reserved for header text

The header text actually starts on line 2 (so line 1 of the header text has been left blank)

Line numbers 61-66 (the bottom 6 lines) are reserved for footer text

The footer text actually starts on line 64 (so lines 61-63 are blank)



Defining the page layout

You can instruct LocoScript to format the page for any size of paper, but you have to tell it how many lines can fit on the sheet, even for standard sizes. This is done in the *Page size* menu of the "Editing header" screen:

Using this menu is just a matter of simple arithmetic. You can alter the page length and the header and footer zones and positions, but the "page body" is calculated by LocoScript and is not directly alterable.

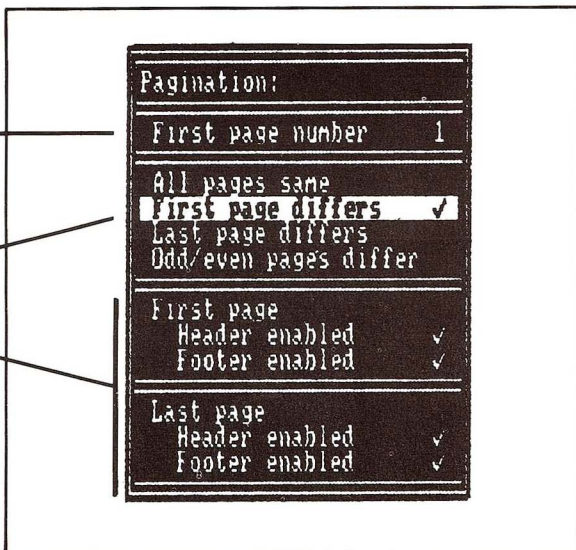
The page body is the text area proper, which is the page length less the number of lines in the header and footer zones. In the example menu shown, lines 7 to 60 inclusive are the lines that the text you type into LocoScript will actually appear on. To alter this, you must juggle the page length and the header and footer zones appropriately.

Defining header and footer text

The header and footer texts themselves are just typed into the relevant slots in the "editing pagination" screen in just the same way as you do in the ordinary LocoScript screen.

You can use any LocoScript format command in headers and footers, like pitch changes, tabs or underlined text. The style commands are treated as separate from the ones used for the main page body, so if you set a header in bold type, it isn't necessary to turn bold off for fear that the whole of the page body will end up in bold too.

You will see there appear to be two sets of header and footer areas. This is because you can make certain pages special – for instance, you might not want to have a



Changing this starts page numbering from the specified number.

This is useful if you are storing and printing documents in separate chapters.

If you want to differentiate between types of pages (eg. odd and even numbered ones), set that up here.

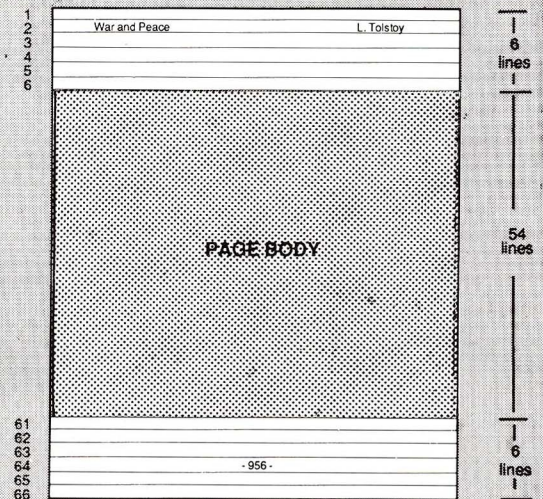
Best to ignore these choices, and set them all to *enabled*

Novel layout step by step

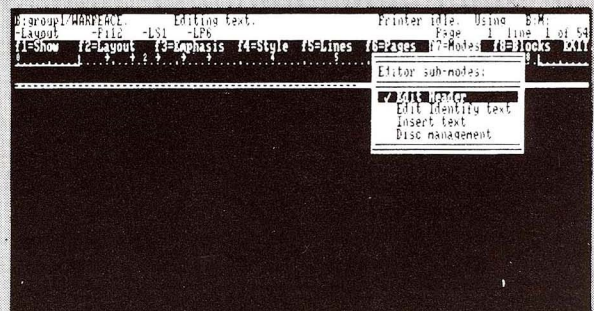
As an example, we'll use LocoScript to create headers and footers for a straightforward document of several pages, like *War and Peace*.

First, think of the layout you want. The header text will say "War and Peace" on the left, and have the author "L. Tolstoy" on the right. The footer will be the page number, centred between hyphens like "- 23 -".

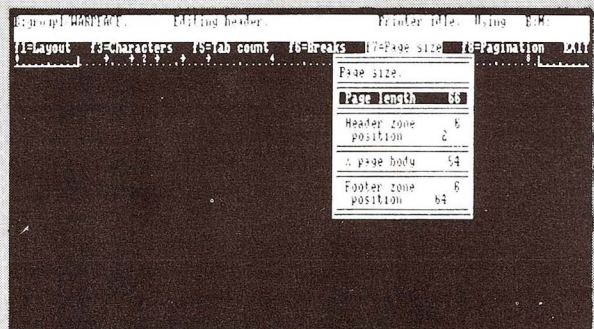
Suppose you want to print it on continuous paper (so the page is 66 lines long) with header and footer margins of 6 lines each, the actual header or footer text being on lines 2 and 64 of the page respectively. As a refinement, the header text should not appear on the very first page.



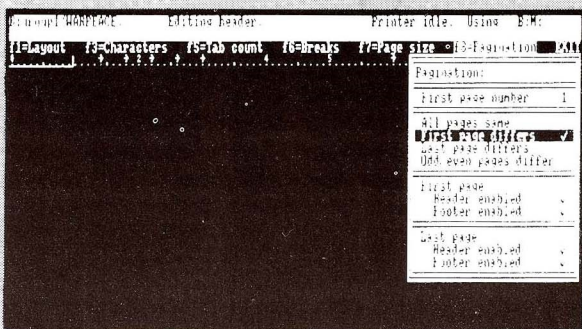
1 Produce a rough sketch showing which lines you want your headers and footers to appear on.



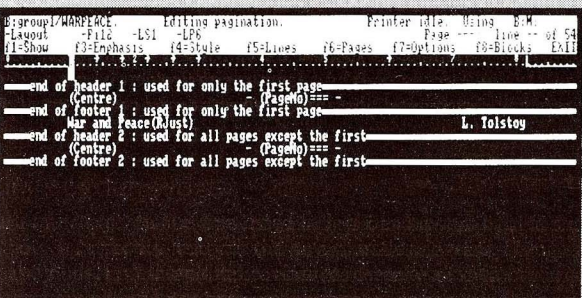
2 In LocoScript, create (or edit) the document that is going to contain your story. While editing, press [F7] to get the *Modes* menu and select the Edit Header menu option. Then pressing [F7] twice will take you to the *Page size* menu.



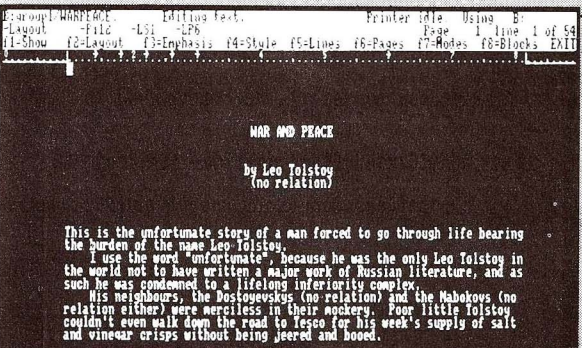
3 Now you fix which line numbers will carry the actual headers and footers: Following your page plan sketch, set the page length to 66, the header and footer zones both to 6, and the header and footer positions to 2 and 64 respectively.



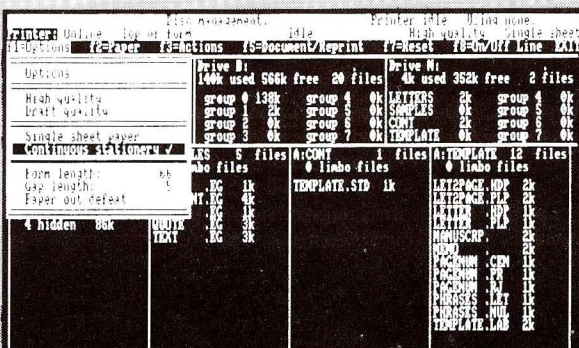
4 Press [ENTER], clear the Page size menu, and then press [F8] for the *Pagination* menu. Set the "First page differs" option (highlight it and press the [+] key). Then leave the menu, and return to the "Editing pagination" screen (use the [EXIT] key).



5 Type in the header and footer texts as needed. (Note: in the footer, - (PageNo) = = - will cause the page number to be printed centred between the two dashes.) Then press [EXIT] and select "Use this pagination".



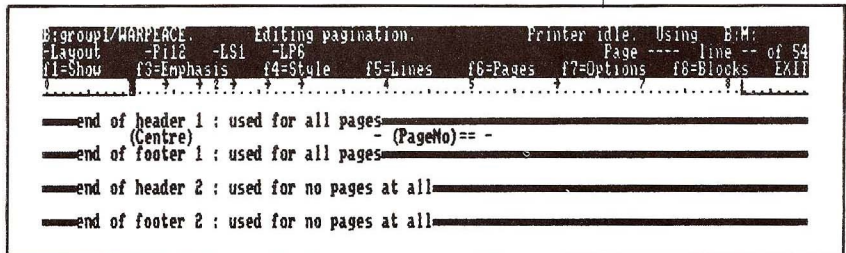
6 You're now back in the ordinary text editing screen, and can begin typing your masterpiece. The headers and footers will not be shown during the ordinary course of editing, but they will automatically appear when the document is printed.



7 When you come to print the document, press [PTR], and [F1] for the *Options* menu. Make sure the "Continuous stationery" option is ticked (use the [] key to set it if not).

header on the very first page of a document, but only for the continuation pages. Or you might want to put the page number in different places for left and right hand pages.

To use this, you first open the *Pagination* menu from the "Editing header" screen and set it up as required – look at the labelled example for details. Then, back in the editing pagination menu, just type in the header and footers you want.



Page numbering

One of the most common uses of headers and footers is to print out page numbers automatically. This is done with the LocoScript *Page Number* command.

You have to tell LocoScript how many spaces to allow for the page number, and within the space you reserve the number can be left, right or centre justified.

Suppose you know your document will not run beyond 99 pages, that means you need two spaces for the number. The way to insert this in a header is to type as part of the header text where you want the number to appear, [+] PN which tells LocoScript to print the current page number – you will have to have the "show codes" edit option in effect to see anything on the screen.

Then immediately afterwards type = =. This centres the page number within those two spaces – note that the equals signs themselves will not be printed out. Alternatively, you could have used << to left justify, or >> to right justify. Any other text in the header is just reproduced as normal.

▲ The way to print page numbers in the form "- 42 -" at the bottom of all pages



Printing out the document

There is a final trap for the unwary after you have bravely hacked your way through the jungle of LocoScript headers and footers, and this comes at the actual printing stage.

Confusingly, even though you have defined all the page lengths and stationery types in your document, the PCW printer has to be told them again separately, or else the page breaks will happen in the wrong place.

So before printing, go into the printer *Options* menu and check that the paper length specifications match the ones defined in the document itself.

LOCOSCRIPT

In which Philip Last explains how to keep tabs

When preparing any document, from a simple letter to the manuscript for a book, knowing how to lay out the page in a way that will catch the reader's eye is an essential skill. The art of making someone want to read a paragraph by its look can mean the difference between the Booker prize and a rejection slip.

LocoScript has a surprisingly large variety of commands to help you lay out your documents in different ways. This month we take a look at how to get the measure of rulers, tabs and the dreaded layouts – why, when, where and how to use them.

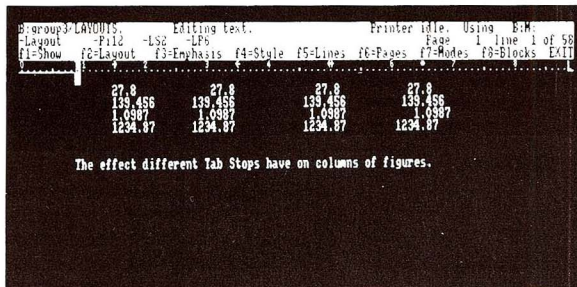
Using Layouts

The basic elements of laying out pages are tab stops, as used on simple typewriters. To set and alter these, you need to be able to create and edit LAYOUTS. Other features that layouts give you control over are the positions of the left and right margins, and whether or not the text is "justified" – each line expanded so that it is aligned to the right hand margin.

First of all, here's a whistle-stop tour of the kind of things you can achieve with different layouts, and tabs in particular.

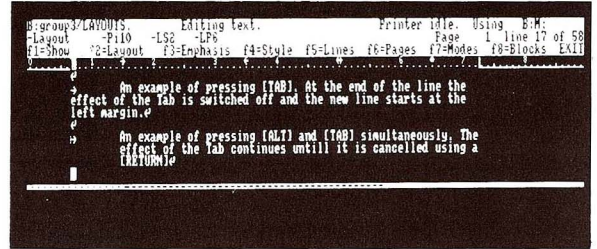
LocoScript offers 4 different types of Tab Stop: Tab, Right Tab, Centre Tab and Decimal Tab. Each has a different effect upon the text. With **Tab**s, text will be left justified to the tab position; a **Right Tab** will right justify text to the tab position; a **Centre Tab** will center text about the tab position; with **Decimal Tabs**, the first full stop in text or decimal point in a column of figures is positioned at the tab.

To position text at a Tab Stop use the [TAB] key. The cursor will move to the first Tab Stop. Any text then entered will start from that position. If you want to move to the next Tab Stop, simply press [TAB] again, just like on a typewriter.



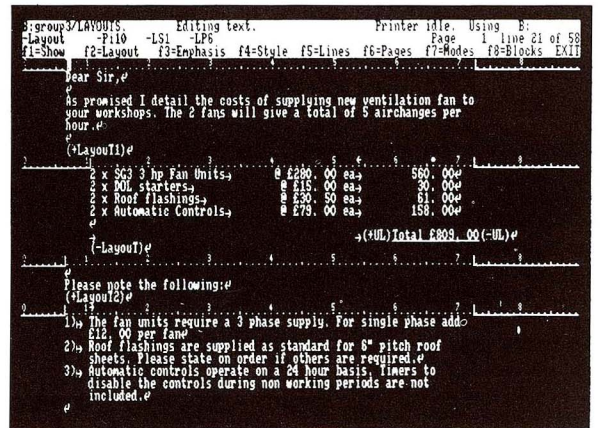
The positioning of text at a Tab Stop lasts only until the end of the line. If the text continues past the line end, the new line will start at the left margin. To allow a whole paragraph to be positioned at a Tab Stop, LocoScript has another type of Tab, the **Indent Tab**.

To get this press [ALT] and [TAB] – ie. hold down [ALT] and then press [TAB] at the same time. In effect, this moves the left margin to the Tab Stop until the next time you press the [RETURN] key, when it reverts to its old position.

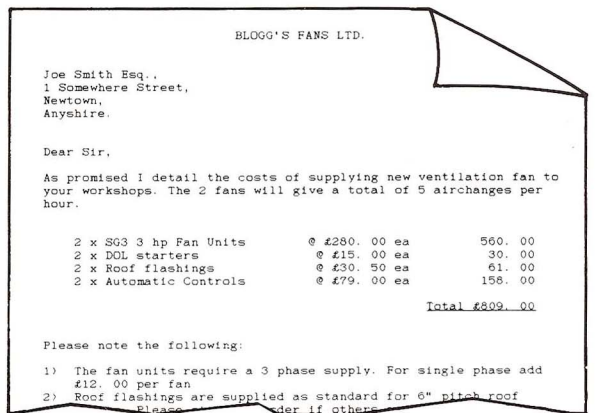


One fine point about using Indent Tabs is that you can't use the more complex Tab Stops (Right Tab, Centre Tab and Decimal Tab) while an Indent Tab is in force. If you try to, the effect will be as though you had used an ordinary Tab instead.

Using an Indent Tab is a useful way of highlighting a section of text. Another way is to alter the margin settings. Pulling in the left and right margins will centre the text and make it stand out from the rest of the document. Changing the Pitch as well as turning Italics or Justification on or off can all make text stand out.



▲ An example of using different layouts in a letter.



▲ This is how it looks printed out

More on Indent Tabs

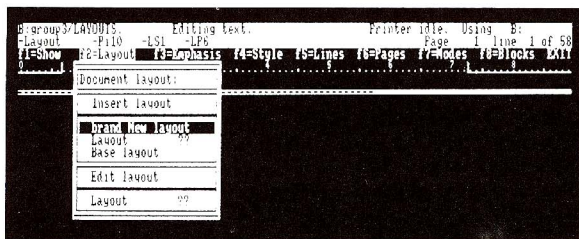
You can use the Indent Tab even if you have no Tab stops set up. If you insert an Indent Tab in the text, LocoScript will start the next line at the tabs position and continue doing so until you press [RETURN].

'S LAYOUTS

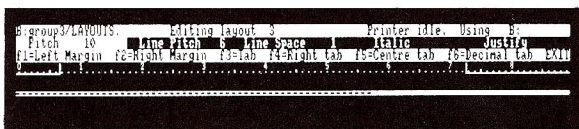
on LocoScript's various layout features.

A New Layout

Here's a step by step look at how to actually go about creating a new layout. Before starting, try to decide exactly what you want to achieve. If you want to insert Tab Stops to lay out columns of figures then count how many characters are likely to be in each column and how far apart they will be spaced. If there are a lot of columns, can you get them all in? You may need to increase the character Pitch (ie. use smaller text). Do you want to highlight areas of text by bringing the margins in or just by indenting the paragraph?



1 Step 1 is to press [f2] and up will pop the *Document layout* menu. The menu is divided into two parts, the top section, labelled Insert layout and the bottom section, labelled Edit Layout. The 'brand New layout' option will be highlighted and this is the one to select by pressing [ENTER]



2 The top three lines of the screen (the information lines) will change. The lines that concern us are the middle line and the ruler line below it. The middle line displays the current settings for Pitch, Line Space, Line Pitch, Italics and Justification. Pitch is highlighted and can be changed by using the [+] and [-] keys. When the Pitch is set, simply move along to the next option using the cursor right key.

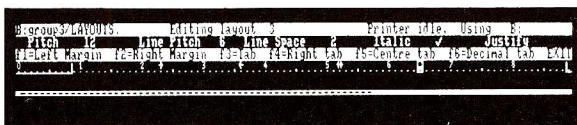
LocoScript assigns each new layout a number, which it is known by from then on. This number is shown on the top line of the *Editing layout* screen – for instance, it's '3' in the example.



3 All the options on the middle line can be changed using the [+] or [-] keys. When you have set Line Space, move to Line Pitch, Italics and Justification until all the options are set. If you want to change any of them, go back using the cursor left key. When you've finished move down to the ruler line with the cursor down key.



4 The cursor highlights the position of the left margin. If you want to change this use the cursor left and right keys until the cursor is positioned where you want it. Press [f1] and the left margin will move to the cursor position. Do the same with the right margin but press [f2] to position it. If you attempt to put the left margin the wrong side of the right margin the PCW will beep in frustration.

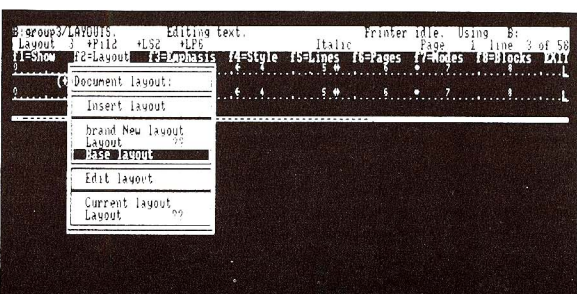


5 Tab Stops are inserted by moving the cursor to the position you want them and by pressing [f3] (Tab), [f4] (Right Tab), [f5] (Centre Tab) or [f6] (Decimal Tab) depending on what type of Tab Stop you want. If you put a Tab in the wrong position, centre the cursor over it and press the [-] key. Unless you've altered the base layout (see later), you won't be able to insert more than 10 Tab Stops.



6 When you've finished making all the changes, press [EXIT] to return to the Editing text screen. The new layout, with a carriage return, will be inserted in the text. The information lines at the top of the screen will display the layout number and Pitch, Line Space and Line Pitch settings. The ruler line will change to that set up under the new layout.

If you want another layout, repeat the procedure. You can't get more than 5 layouts without altering the base layout.

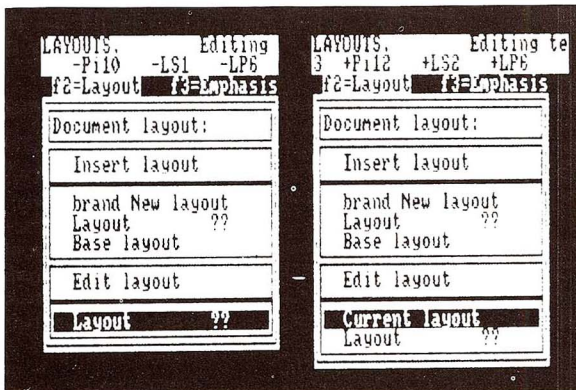


You can return at any time to the base layout by highlighting that option on the Layouts menu and pressing [ENTER]. Similarly, once you've created a layout you can reuse it later on by highlighting the 'Layout ??' option on the Layout menu, typing the number of the layout in question and pressing [ENTER].

Editing an Existing Layout

If you create a layout and then realise later on that you have perhaps put the Tabs in the wrong places, all is not lost. Another option on the 'f2 Document layout' menu is 'Edit layout' – if you're currently using the base layout then the menu will have only one option, Layout ??. If you already have a layout in place then a further option, Current layout, will be offered. Highlight the one you want. If it's the current layout, press [ENTER], if not then first enter the number of the layout you want to change before pressing [ENTER].

Once in the layout editing mode, changes can be made in exactly the same way as described in 'A New Layout'. The main difference is that the layout won't be inserted when you exit back to the Editing text screen. To do that you must select the option, Layout ??, from the top half



of the layouts menu. Be careful when editing a layout that you have already used in your text. Any changes to the layout will take place throughout the document where that layout has been used.

Base Layouts

So far we've seen how to use ordinary layouts to do simple things like defining tabs across the page. However, there is a special kind of layout called the Base Layout which has an important role in LocoScript.

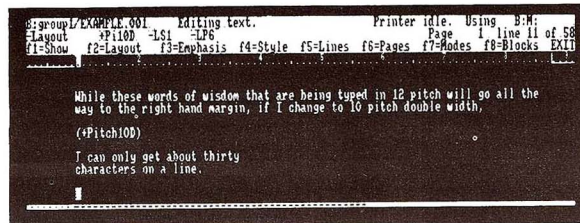
When you first create a new document (using the C key from the disc management menu) and you start typing text, a particular Layout is being used by default without you having specifically set one up. This is known as the Base Layout, and is defined in the TEMPLATE.STD file that is in the current group.

The Base Layout has certain special effects. Firstly, it determines the width of the lines on the printed page. The distance between the left and right margins gives the number of characters that will be printed on the line, using the character size (the "pitch") which is defined in the Base Layout itself.

For example, looking at the screen picture below, the left margin is set at 10 and the right margin at 88. This means that there are 78 characters per line, and the text is being printed in 12 pitch – ie. 12 characters per inch, so each character takes up 1/12". Therefore, the line length is 6".

Now this doesn't sound very surprising, but the

importance of the settings in the Base Layout becomes apparent when you change print size. LocoScript will always keep the line length to 6" from now on, with these margin settings, no matter what the pitch used is. Consider changing the pitch to double spaced 10 pitch (using the 'f4 Style' option): this allows you to use five characters per inch.



Therefore, on the 6" line you will get 30 characters. This is eminently sensible since most pieces of paper are the same width all the way down the sheet, but it is worth understanding how LocoScript judges these line lengths otherwise you will be surprised and puzzled when your line on the screen appears to end after 30 characters even though the right margin apparently allows 88.

The other important effect the header has is to establish the style that headers and footers will appear in, if you are using them. You will probably have noticed that if you are printing in bold text or some other effect and your text runs across a page break, the header at the top of the next page is not in bold. Instead, the options defined in the Base Layout are used.

Editing the Base Layout

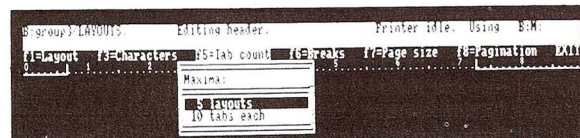
It only usually makes sense to edit the Base Layout when you are creating a new TEMPLATE.STD file. In this case, the changes you make will also be used in all files created from that template; otherwise it is a lot faster just to insert a whole new layout.

To get to the Base Layout, press [f7] 'Modes' while editing a file, and select 'Edit Header'. Press [f7] again, then [f1] on the new screen, and you are now editing the base layout. The screen you see is exactly the same as for editing any ordinary layout.

When you have finished editing the Base Layout, keep pressing [EXIT], and [ENTER] whenever a menu pops up asking you to confirm that you really mean it, until you end up back in the editing text screen.

Getting More Layouts

As we have seen, LocoScript normally restricts you to a maximum of 10 tab stops and 5 layouts. You can increase the number available to you if need be: press [f7] and select Edit Header, then pressing [f7] again will take you into Editing Header mode.



Select 'f5 Tab Count' and you will be presented with a menu containing both Tab Counts and Layouts. Highlight the option you want and type a number. Press [ENTER] to register the change. Both can be set to a maximum of 99 although doing this will add 10k to the length of the document.

To return to the Editing Text screen press [EXIT], [ENTER], [EXIT] and [ENTER] in that order.

Short Cuts

You may find some layouts to be so useful you would like to have them available all the time. If you set up the layout in Template.Std then every time you create a document in that group, the layout will be there ready for you to use.

Once set up, layouts can be inserted without using the layouts menu. Press [+] LT followed by the layout number you want. Then press [ENTER] & [RETURN].

If you want to return to the base layout, press [-] LT.

GRAPHICS FROM LOCOSCRIPT

Frank Armstrong, helped by David Hoffman, shows how to turn the tables on LocoScript

Many of the documents which you produce for business or domestic use may contain tables. Typically you'll want to have columns of varying width, provision for sub-totals and grand totals, and smart little arrows to highlight the 'Amount due now' section. Often simple blank tables are all that are needed, for filling in by hand at a later stage.

So how do you produce a table like this:

Last month in our regular LocoScript spotlight we saw how to set out figures and text in columns by using the layouts and 'tab' facilities. Now discover the delights of printing neatly ruled dividing lines, and other ways to titivate your tired eyes.

Col. 1	Col. 2	Col. 3	Col. 4	Col. 5	Col. 6	Col. 7
TOTALS					→	

Col. 5	Col. 6	Col. 7

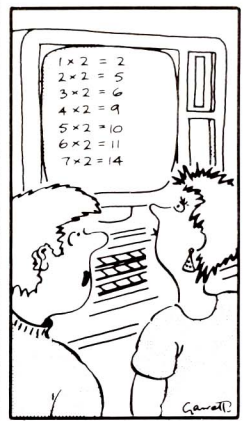
The first thing to do is to work out where you want the columns for the table to appear on the final printed page. The Editing screen in LocoScript shows a ruler at the top dividing the page into 80 or so columns.

What you have to do is set a Tab Stop at each column on the ruler where you want to print a vertical dividing line in your table. For detailed tables, you might find it useful to draw out a screen grid on paper and sketch the table by hand first of all, otherwise you will find yourself involved in a long process of trial and error to get it just right.

Create a new layout by pressing [F2] and choosing the 'Brand New Layout' option. Use the cursor down key to get at the ruler settings, and set tabs in positions where you want the column lines to appear. Tabs are set by using the cursor left and right keys to move to the column you want, and then pressing the [F3] key.

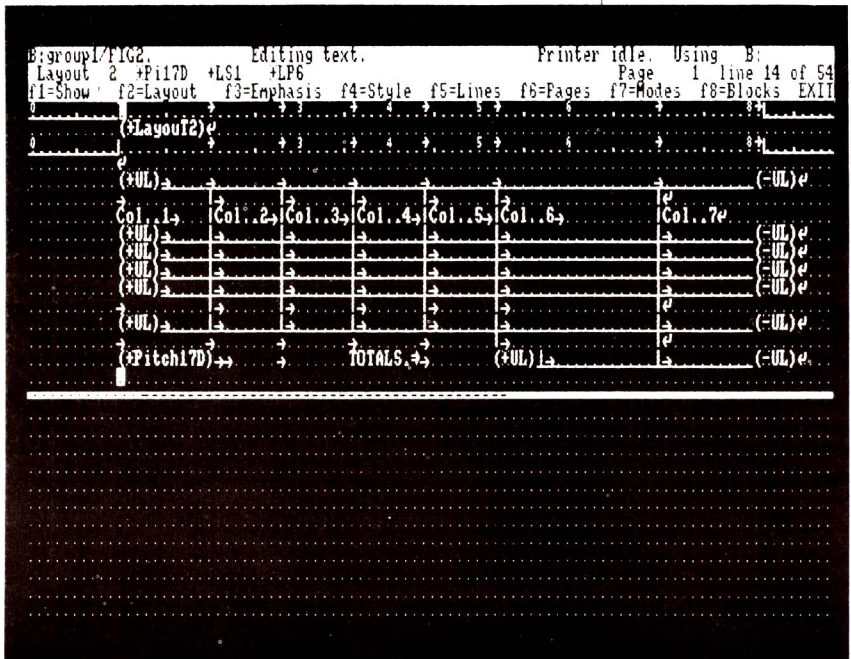
Now when you press [EXIT] to get back to the ordinary editing screen, you are ready to type the table in. For every horizontal line you want to have, go to its starting column and enter (UL (from the 'f3 emphasis' menu). Then press the [TAB] key to get to the end point of the line, and turn off underlining. It's important to use tabs not spaces even though they may appear OK on the screen, for reasons that will become apparent later on.

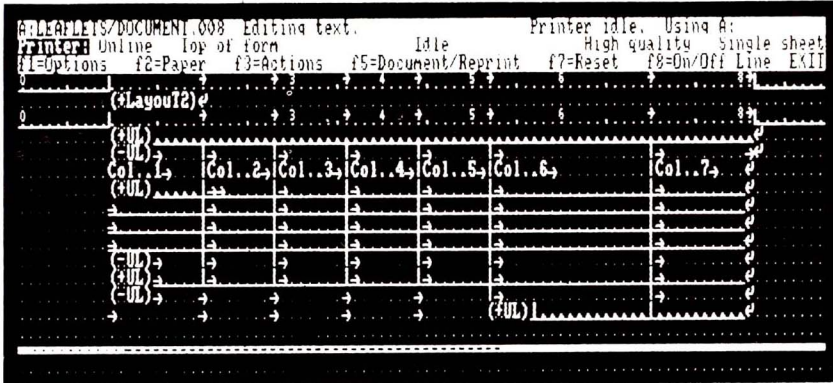
To get vertical lines, use the key combination [EXTRA]+fullstop. Use the [TAB] key to get to the correct position for the column line and place the vertical line mark there. To get the table layout produced above, you should end up with something like the screen shot on the left.



"IT FORMATS TABLES BEAUTIFULLY BUT THERE'S JUST SOMETHING I CAN'T PUT MY FINGER ON..."

▼ The correct screen for the example table





▲ An incorrect screen layout for table printing

And the printout that it produces

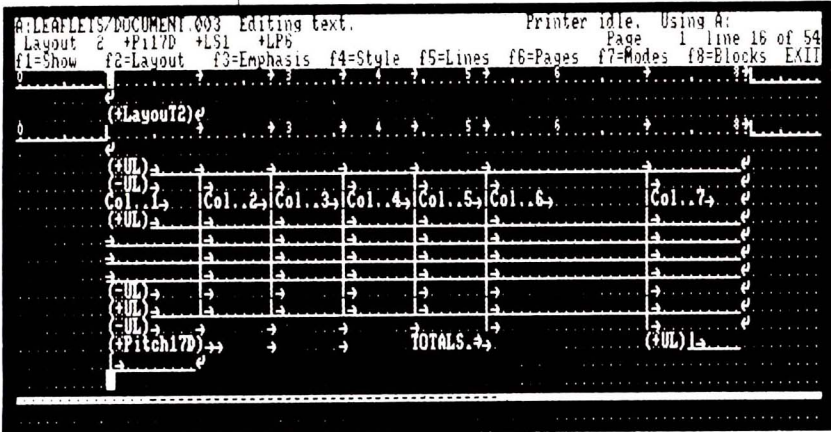
Col. 1	Col. 2	Col. 3	Col. 4	Col. 5	Col. 6	Col. 7

Getting the wrong end of the stick

However, there are pitfalls for the newcomer, and some of these are illustrated below. It is a screen printout of what appears at first sight to be a beautifully proportioned table. Smugly pressing the 'Save and Print' option after [EXIT] however produces the rather ragged table shown. Why?

When you examine the screen, you can see that on the line immediately below the (:)Layout2), the (:)UL) option has been correctly chosen, but the space bar rather than the TAB key has been used to carry the underlining to the end of the line, followed by a [RETURN]. A similar mistake has been made on the fourth line, resulting in the misplacement of the first

▼ The corrected screen layout for what was required



vertical column mark in that line.

Also, on the final line, where the first vertical bar appears to be displaced, it in fact prints correctly, since LocoScript instructions do not appear on the printed output. However, because the user has mistakenly lined up the second vertical on the line with the line above, that one does appear misplaced on the printout even though it is OK on the screen!

The 'tame' version is shown in the next screen print. Here the underlining has been carried forward on each line using the tab settings alone and, presto, it works! This produces the table we wanted in the first place. Incidentally, the smart arrow in the TOTAL line is produced with [ALT]+>.

How then can you save this table for use in future documents? LocoScript can save chunks of text as 'blocks' or 'phrases' for future use in other documents. The current example is too large to be stored as a phrase, so the best thing to do is to save the table as a LocoScript document in its own right, via the COPY key and 'f8 = Blocks' menu.

To do this, type the table into your LocoScript document, place the cursor at the first line of it and press [COPY]. Now move the cursor to the end of the table – you will see the text it covers being highlighted – and press [COPY] again. You have to specify a block number for LocoScript to remember the text by, so choose a

number from 0 to 9 and type it in. The highlighted text will revert to normal.

Now save this block as a document. Pick the 'f8 = Blocks' menu, enter the number of the block that you chose in the 'Save Block?' option, and press [ENTER]. You will now be faced with the Disc Management screen, and at the top of the screen is the ever inquisitive header prompting you to 'pick a destination group'. Use the cursor keys to get the cursor bar into the group you want, press [ENTER], and provide a name for the file as you are asked.

The block is then ready for use at any time in the future, by pressing [F7], 'Insert text' and picking the right file, providing the destination group you have chosen is not in drive M! Remember; if you do not save the block to a file in this way, it will be lost and gone for ever once you finish editing your current document.

Getting your layouts straight

Sadly, your highly-prized block has not been able to carry its layout round with it in its back pocket. Inserting it into a new document using 'Insert Text' as just described will cause it to blithely adopt whatever layout happens to be hanging around at the time. This may be quite different from the one which was used when the table was defined, with tabs in quite the wrong places.

So how do you get round it? The best way is to make sure you save a layout identifier (using 'f2 = Layout' menu and its 'Insert Layout' options) with the table text. When you define layouts, they are stored with that document for future use. If the document you define them in is TEMPLATE.STD, then those layouts will be available in all documents created with that template.

If you only use a few different layouts, you can store them all in the TEMPLATE.STD that you usually use. Then, when the LocoScript command to use a particular layout is read in, the correct layout will be used automatically. However, this only works if you are working in the same group and with the same TEMPLATE.STD file as was used when the table was created. It is a simple matter to slip your block in using f7 = Modes, Insert Text, and Bob has become your uncle once again.

Unbroken vertical lines

The process just described produces quite acceptable results, but it is possible with a little trickery to produce tables which have unbroken vertical lines and are also filled with text. To do this, you have to get to grips with altering the printer's 'line spacing'.

As before, the first step on the road to happiness is to set up a new layout which has tab stops in the correct places for aligning the column rules. Work out on a piece of paper where you want the columns to go, and then set up the layout with tabs in the right places.

The table is set up much as in the previous examples. Horizontal lines are produced with the underline code and by TABbing across to the next column, and the vertical lines are placed by [EXTRA]+fullstop.

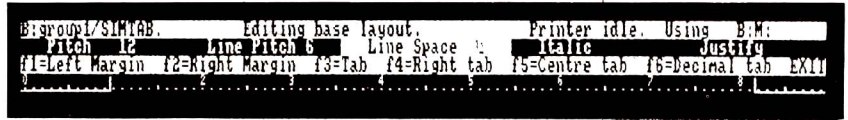
Now for the clever part. If you change the line spacing to 1/2, then when the page is printed the top of one line crashes into the bottom of the line above it. This isn't very useful for normal text, but if you force the vertical line characters to overlap then you achieve the semblance of true unbroken lines.

To change the line spacing you will need to alter the layout you are currently using. You can do this at the same time as you place the Tab stops in the layout, or afterwards by positioning the cursor somewhere in the table text and using the 'f2=Layout' menu and its 'edit current layout' option.

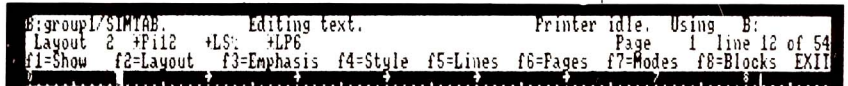
Once in the editing layout mode, use the cursor keys to highlight the 'Line Space' item, and then the $\frac{1}{2}$ key to set the spacing to 1/2. [EXIT] as normal to finish the layout editing.

Finally, for the *piece de resistance* you can try and type some text into the tables. Just type it in by eye on the screen, making sure that the column lines are still lined up. Remember you are printing out lines half-spaced, so to run text on adjacent lines on the printed page you will need to put a blank line between them on the screen, or the printed version will be garbled.

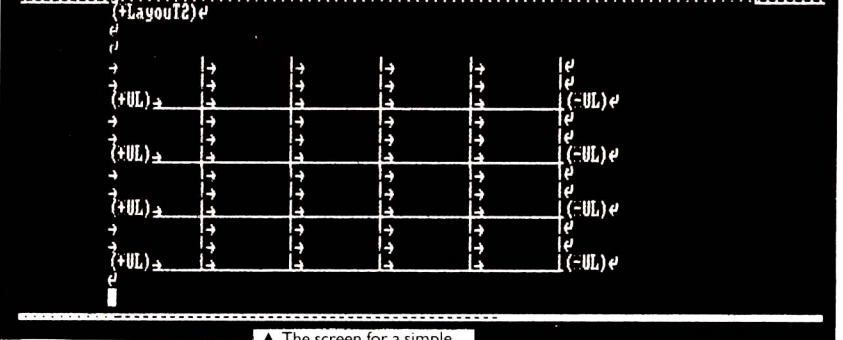
Don't forget to use tabs not spaces to line things up, or you could find mysterious things happening when you print out, especially if you use proportionally spaced text. To end with, a grand example:



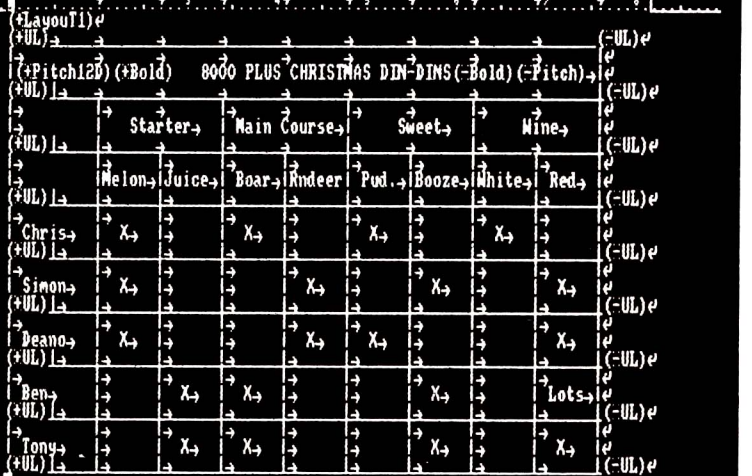
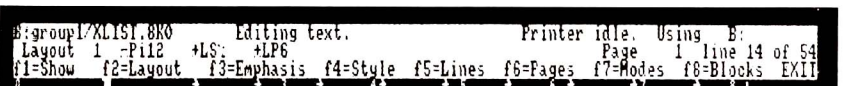
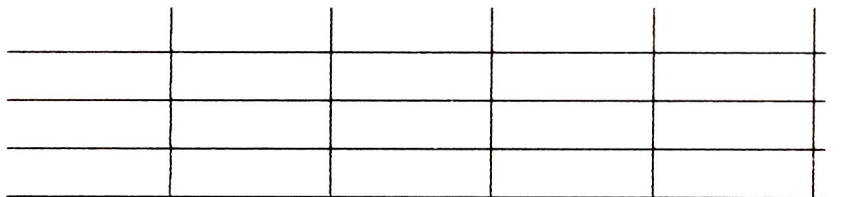
▲ Altering the layout to give half-spaced lines



▲ The screen for a simple table with unbroken vertical lines



And its results ▼



This... ►

... produces this. ▼



8000 PLUS CHRISTMAS DIN-DINS							
	Starter		Main Course		Sweet		Wine
	Melon	Juice	Boar	Rndeer	Pud.	Booze	White Red
Chris							
Simon	X		X		X		
Deano	X				X	X	
Ben		X	X	X			X
Tony		X			X		Lots X

BUSINESS MATTERS

Using LocoScript to polish off simple business letters

The advent of low-cost LocoScript word processing has encouraged many people in business to type their own letters and documents, whereas previously they might have dictated a letter for a secretary to type up. Beware, though – there are well defined conventions for laying out business letters, and ignoring these could give your business an amateurish image.

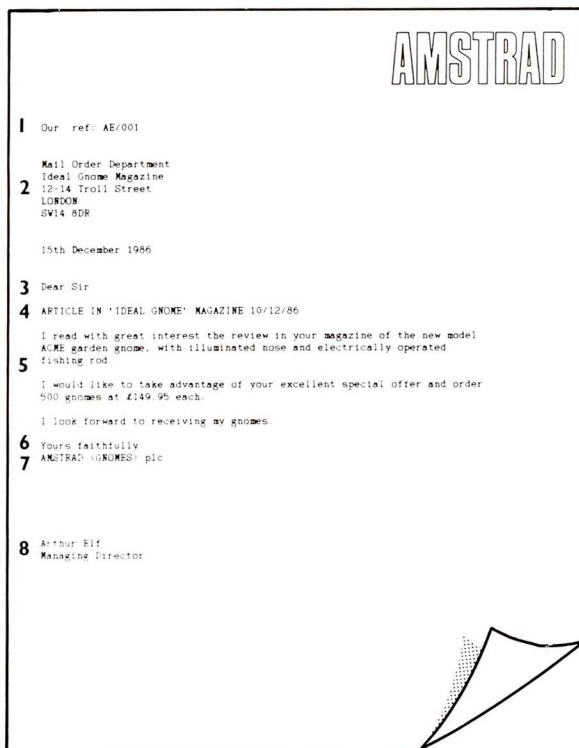
There's an old Disney cartoon which ought to be compulsory viewing for all driving test candidates. It stars placid, mild mannered Goofy, and follows him on his route to work one day. In the morning he ambles out to the car and gets in; at the instant he turns the ignition key, a miraculous transformation overtakes him. His eyes bulge, his face goes red, he hits the horn, screeches off with a plume of burning rubber and careers down the road leaving a trail of wreckage behind him.

This all goes to show how ordinary people become mentally unbalanced when faced with a car. In some respects, word processors can have the same effect; people who chat to you on the phone quite normally suddenly get hold of a word processor and forget the basic principles of putting one word after another on a page.

The business of writing

Different secretarial colleges teach different styles of letter writing, and if you work in a big company there may well be a 'company standard' which you have to use anyway. But for general use, here are some simple guidelines for writing single page letters. Following it are notes which go through each part of the letter.

Overall, you need to aim for a friendly but professional feel to the letter. This means using the parts of the



standard layout which are appropriate to your reader, but avoiding irrelevant bits (like the 'Our Ref' part) where they aren't needed. Don't get seduced into wordy officialese, like 'With reference to the conversation between the aforementioned and the undersigned.' When you've finished a letter, imagine that you are on the phone to the person and reading out the body of the letter text. If it doesn't sound natural, you've probably gone over the top.

1 References. The idea of a reference on a letter is to help you trace it later on. If somebody writes to you saying, "Thank you for your letter of 16th December, reference AE/001" you will be able to quickly find your copy of the original text and remember what's going on.

With LocoScript, one good idea is to have your reference as the name of the file in which the letter is stored. If you have several discs numbered 1, 2, 3 ... then you can add this in too. For example, if you have written a letter in a file called IGM.001 on disc number 2, you might use the reference 2/IGM.001

If you don't keep formal records of letters then leave out the word 'Ref.' altogether – it looks pompous to have irrelevant headings flying around. Also, don't use this if you are writing as a private individual rather than as a company, since that looks silly too.

2 Address. This is positioned so that when an A4 letter is folded into 3, the address will show properly through a standard envelope. In keeping with the modern punctuation style, there are no commas at the ends of lines.

You may have to experiment a little to find the best position for the address on your envelopes, but it will probably start around the 11th or 12th line on the page. Bear in mind that the PCW automatically leaves 6 blank lines at the top of a page when you use single sheet paper, so the first line of your LocoScript letter appears on line 7 of the page.

Leave two blank lines above and below the address, so that you don't get rogue lines showing through the address window of the envelope.

3 Greeting. Obviously, the whole tone of a letter is determined by how well you know somebody. If you are writing to an organisation where you don't know who will read the letter, begin off 'Dear Sir' (or 'Dear Sir or Madam' if you want to keep your options open). With this greeting, the letter should be ended 'Yours faithfully'.

On the other hand, if you know the name of the person you are writing to (perhaps you have spoken on the phone), begin off 'Dear Mr. Elf', 'Dear Arthur' or whatever. End these letters with 'Yours sincerely'.

4 Title. If the occasion demands it, put the subject of the letter in capitals at the start. Not all letters need this line, so use your judgement to tell whether it looks silly or not. Generally, it is used when you are writing to someone for the first few times, but not once you get into an involved correspondence.

5 Text body. The modern style for text is that paragraphs start hard against the left margin (no indentation), and there is a blank line between each paragraph. This helps to break up long pages of text and make them easier to read.

The final line of the letter should give some clue as to what will happen next. If you expect a reply from someone, say so here. Equally, if they have to wait for you to do something, say so.

6 Close. What form of close you use depends on what you wrote in the 'Dear ...' slot. Often, to a fairly close business acquaintance, 'Best wishes' is a good informal close – it is friendly without being too familiar.

7 Affiliation. If you are writing on behalf of a

company, put the name in capitals here, otherwise forget this bit.

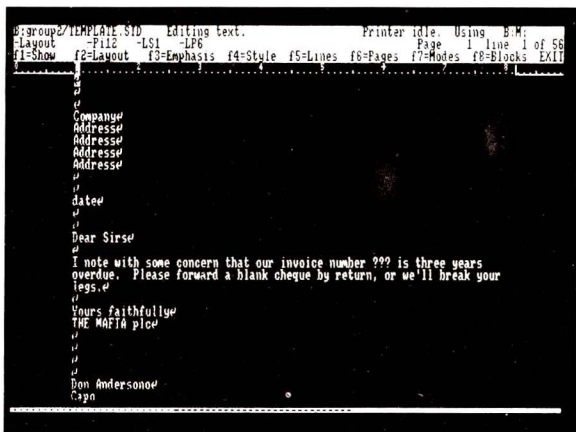
8 Signature Block. Your name and position, which allow you to indulge in a flowery signature since no-one actually has to read it to discover who you are.

Where LocoScript's templates come in

LocoScript is tailor made for letter writing. By now you must have written at least one letter with it – all the manuals and teach-yourself courses start by showing you how to create a document, type in simple sentences and print it out. However, there are several ways to use LocoScript to make repetitive letters simpler.

First of all, a quick refresher course on disc organisation and 'templates', starting from the familiar Disc Management screen. The highlight bar shows you which group you are currently in. Each group should have a document in it called TEMPLATE.STD; when you press the 'C' key to Create a document, your new document starts off as a copy of its group's TEMPLATE.STD.

This means that if you write a lot of similar letters, like credit chasing, you can write a standard form letter with the names and amounts left blank and store this as the TEMPLATE.STD for your credit chasing group. Then when you create a new document in that group, you always start with that basic form letter, and you just go through filling in the blanks. TEMPLATE.STD is a normal LocoScript document; you can edit it like any letter, rename it or delete it as you wish.



▲ A sample TEMPLATE.STD for credit chasing

Grouping it all together

Discs can have up to 8 groups for you to organise your files into – perhaps one group for invoices, one for final demands, and so on. You just use the cursor keys to move between groups and documents.

One catch is that groups with no documents in them are not shown on the screen at all, so you can't move the highlight bar into those groups. In the top third of the Disc Management screen is a list of the groups, with a highlight bar over the current one. To move to a new group, you hold the [SHIFT] key down and use the arrow keys; if you are in group 0 of a disc, [SHIFT]→ takes you into group 4, even if there are no files in group 4.

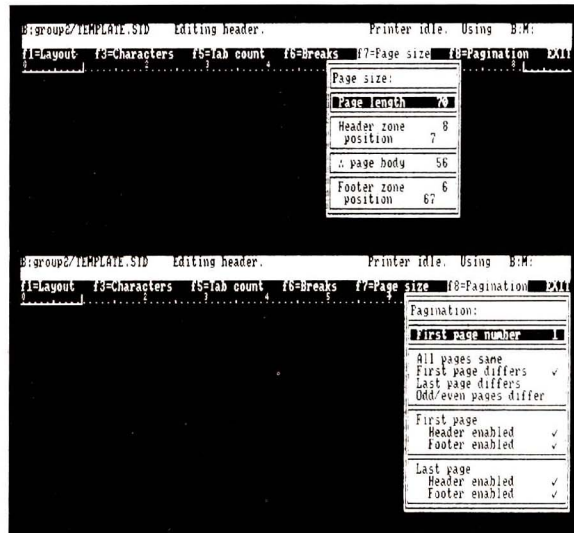
Each group ought to have a TEMPLATE.STD. The best way to create a new group is to place the cursor bar over any existing TEMPLATE.STD file and press the 'f3=Copy' key. The screen now asks you to select the 'destination group', so use [SHIFT] and the cursor keys to get into the group you want, press [ENTER], and bingo! Your new group opens up on the screen with its TEMPLATE.STD, which you can then edit as you want.

Headers and footers for letters

LocoScript's headers and footers are at times complex to

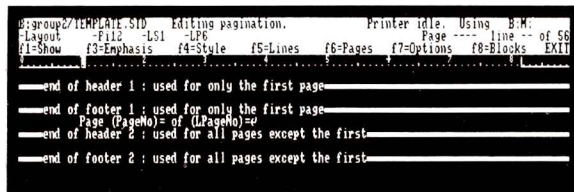
use, and were covered fully in issue 2 of 8000 Plus. They are particularly useful when writing letters, as you can number continuation sheets automatically.

For multiple page letters, you may want to have at the top of each sheet (except the first) 'Page 2 of 4', or whatever the numbering actually is. LocoScript can handle this very nicely.



Edit your TEMPLATE.STD file in the group you keep letters in, and from the Editing screen, use the [f7] key, the 'Edit Header' option, [f7] again and finally [f7] yet again to get to the page size menu. Set this up to the sizes shown, which are good values for A4 paper. When this menu is OK, press [ENTER] to clear it off and then [f8] for the Pagination menu. Set this up too as shown.

Press [ENTER] to clear the menu off, and then [EXIT] which takes you back into the 'Editing Pagination' screen. Now you can set up the headers and footers.



For the first page, leave both the header and footer blank. Use the cursor down key to get into the header region of 'All pages except first', and type

Page = of =
Now place the cursor on the first equals sign, and put a 'This Page Number' code there, either with the [f6] key or by the short-cut \square PN. Place the cursor on the second =, and put a 'Last Page Number' code there, again either with the [f6] menu or by typing \square LPN.

All this magic means that when the pages are printed out, the first equals sign is replaced by the current page number, and the second equals sign by the number of the last page in the letter (ie. the total number of pages in the letter). If you have a letter longer than 9 pages, this method will go wrong – you will need to use == instead of just =, but otherwise it works in the same way.

Caveat Scriptor!

So now you can all write beautifully laid out, grammatically perfect letters. We at 8000 Plus expect to receive nothing but excellence in our postbag – we may start offering prizes for the worst offenders each month. You have been warned! A final piece of advice; don't forget to use a spelling checker for letters more than a paragraph or so long. EXIT

Boffin note

LocoScript's Groups are really the same thing as CP/M's User Numbers. If you start CP/M up, put a LocoScript disc into the drive and type DIR, all the document names in the first group will appear. If you type USER 1 and then DIR again, you will see all the documents in the next group. CP/M has 16 user areas numbered 0 to 15; LocoScript's 8 groups are in user areas 0 to 7. If you have deleted a letter with the [f6] key in LocoScript, it is marked as a 'Limbo' file; Limbo files for groups 0 to 7 are stored in the user areas 8 to 15 respectively.

INDEX

A5 paper	7	Esc character	26	PIP command	
Accents	5,16	Editing		error messages explained	3
Alarm call	13	Basic command lines	10	to avoid LocoScript	8
ALT key	11	using RPED	28-9	converting WordStar files	19
ASCII codes	26-7	Envelopes		Printing	16
ASCII files		printing	12	POKEing programs	10
editing in LocoScript	7	feeding through printer	20	Printer	
creating from CP/M	25, 280000	ERASEing from Limbo	13	changing defaults	27
Auto-start discs	17,23	Fill command	14	controlling from CP/M	26-7
Automating printer changes	27	FontGem (SuperType))	17	non-standard	12
B drive	2,22	Footers	33-5,43	resetting	27
Backing up discs	18,22	on single page document	4, 19	Printing	
Backslash	24	Footnotes	33-5	alternative typefaces	13, 20
Base layout	38	typeface	20	echoing CP/M commands	2
Basic		Foreign characters	5	from CP/M	26-7
clock	10	Formatting discs on 8512	2	from Basic	10
editing command lines	10	French characters	16	from SuperCalc	13
editing with RPED	29	using in CP/M	7	headers and footers	35
loading quickly	4	Full disc	19	labels	7
moving and copying lines	20	Game of Life	11	multicolumn printing	15
PEEKing and POKing	8,10	German characters	16	multiple copies	12
printing from	10	Greek characters	16	screen dump	2
program to examine memory	8	Groups in LocoScript	43	stopping a printout	4
running from CP/M	18	Hacking	8	overprinting	5
saving files	6	Hard spaces	12,18	while editing	5
typing shortcuts	6	Headers	33-5,43	Punctuation symbols	16
underlining, reverse video and beeping	9	Headers tricks	19	PROFILE.SUB file	23,25
Batman	13	Hitchhikers Guide	19	Random numbers	16
Beeping		Highlighting text	6	Resetting printer	27
from Basic	9	Hyphenation	12	Reverse video	
from LocoScript	13	Indent tabs	36	from Basic	9
Bold text, highlighting	6	INPUT command	14	to improve contrast	9
Boot discs	17,23	Italic text	6	to mark italics	6
Business letters	42-3	Keys, defining	13, 24-5	Ribbon reinking	8
Business stationery	5	Keyboard feet	13	RPED utility	28-9
Cancelling auto-start	17	Label printing	7,12,14	Rulers	36-8
Caps lock key	2	Layouts	36-41	Screen	
Cardbox sorting	6	Letter templates	4,6,43	better contrast	9
Centring text	12	Letter headings	5, 42-3	page scrolling	8
Centronics port	30	Life, free game of	11	Second drive on 8512	2
Character set, full	16	Limbo files	13	Serial interface	30-1
Character pitches	16	Lines		SETDEF utility	8
Clock in Basic	10	per page	33	SETKEYS utility	13, 24-5
Columns	4,15	vertical	39-41	SETSIO utility	31
Command lines		LocoScript	see individual entries	Shell program	15,17
editing in CP/M	4	Long lines in LocoScript	7	Short cuts in LocoScript	18
single key-stroke using SETKEYS	25	Long files in LocoScript	7	Spool	14
Control key, PCW name for	2	LPRINT command	26-7	Speed	
Copying to drive B	20	Mailmerge	14	LocoScript long files	7
Copying safely	16	M drive, use of	22	STOP key	12
CP/M	see individual entries	Margins	16, 35	SUBMIT utility	23, 25
CP/M software, how to run	18,22-3	Margin notes	19	SuperCalc print styles	13, 18
Cracker II	6	Mathematical expressions	9,16	printing in LocoScript	11
Cursor		Monetary symbols	16	SuperType	17
switching off and on	10	Moving files while editing	5, 19	Tables	39-40
Czech characters	20	Multicolumn printing	4,15, 19	Tabs	36-41
Dash symbol	16	NewWord		saving disc space	6
Defining keys	24-5	boot disc	3	TAB command in Basic	70
DEVICE utility	30-1	using with Spool	14	Templates	
DIRectory		quick line moves	17	letter headings	4,5,6
Obtaining printout of	2	converting files	19	margin settings for	7
When to use	18	Overprinting	5	using with Layouts	40
Directory full	16	PAGE key		Timing in Basic	10
Disc drives, differences between	22	alternative to	8	Touch typing tips	13,17
Disc		typing ahead	5	Typefaces, alternative	13, 20
backing up	22	Page layout	34-5	Typestyles from CP/M	26-7
auto-start from	23	Page numbers	18, 35,43	Typing ahead	5
errors	19	Pagination menu	34	Underlining	
Disc Manager	5	Paper		from Basic	9
DISCKIT utility		feeding continuous	6	improved in LocoScript	9
Avoiding disc swaps	3	length	35	Unit key	2
Backing up master discs	18, 22	Passwords, bypassing	11	VERIFY command	16
Copying to drive B	20	PEEKing at programs	8,10	Vertical lines	39-41
Document layout menu	36-8	Phrases		Word breaks	12
Double density discs	2	use of	2,4	Wordcount automation	15
Esc key, PCW name for	2	deleting	7	WordStar file conversion	19
		those supplied	8		

