AMSTRAD
DMP2000



USER INSTRUCTIONS

Introduction

AMSTRAD DMP2000 Dot Matrix Printer

The DMP2000 is a further milestone in the AMSTRAD range of low cost, high performance computer products.

It combines the versatility of an industry standard software instruction set with AMSTRAD 's expertise in quality engineering at an exceptionally low price.

Single cut sheet or continuous paper may be used, and the ingenious 'flatbed ' design allows the easy insertion and alignment of both tractor and friction feed paper. Printing speeds of over 100 characters per second will make rapid work of even the most lengthy drafts.

The extremely wide choice of sizes and typefaces coupled with a complete ASCII and international character set should provide a solution to any printing problem. In addition, the implementation of dot addressable graphics and standard Epson compatible command codes will allow the DMP2000 to operate directly with most computer software, including word processing programs, graphics, screen dumps etc.

The DMP2000 will operate with any AMSTRAD (or other) computer which incorporates a standard Centronics parallel interface.

AMSOFT

A DIVISION OF

AMSTRAD

CONSUMER ELECTRONICS PLC.

© Copyright 1985 AMSOFT, AMSTRAD Consumer Electronics plc.

Neither the whole nor any part of the information contained herein, nor the product described in this manual, may be adapted or reproduced in any material form except with the prior written approval of AMSTRAD Consumer Electronics plc. ('AMSTRAD ').

The product described in this manual, and products for use with it are subject to continuous development and improvement. All information of a technical nature and particulars of the product and its use (including the information and particulars in this manual) are given by AMSTRAD in good faith.

AMSOFT welcome comments and suggestions relating to the product or to this manual.

All correspondence should be addressed to:

AMSOFT,
Brentwood House,
169 Kings Road,
BRENTWOOD,
Essex CM14 4EF.

All maintenance and service on the product must be carried out by AMSOFT authorised dealers. Neither AMSOFT nor AMSTRAD can accept any liability whatsoever for any loss or damage caused by service or maintenance by unauthorised personnel. This guide is intended only to assist the reader in the use of the product, and therefore AMSOFT and AMSTRAD shall not be liable for any loss or damage whatsoever arising from the use of any information or particulars in, or any error or omission in, this guide or any incorrect use of the product.

CP/M is the trademark of Digital Research Inc.

Acknowledgements to Acorn, Apple, BBC, CBM, Centronics, Epson, Microsoft, MSX, Oric, and Sinclair

First Published 1985
Second Edition Autumn 1985

Written by Ivor Spital
Contributions by Alexander Martin

Published by AMSTRAD
Typeset by KAMSET typesetting graphics (Brentwood)

AMSTRAD is a registered trademark of AMSTRAD Consumer Electronics plc.
Unauthorised use of the trademark or word AMSTRAD is strictly forbidden.

IMPORTANT

You must read this....

1. Always connect the mains lead of the printer to a 3-pin plug following the instructions in Chapter 1.

2. Do not attempt to connect the printer to any mains supply other than 220-240V~ 50Hz.

3. There are no user serviceable parts inside the printer - do not attempt to gain access inside the casing. Refer all servicing to qualified service personnel.

4. Do not operate the printer with its ribbon removed.

5. Do not operate the printer with no paper loaded.

6. Do not switch on or operate the printer with the print head stabiliser in position.

7. Do not bring drinks or any other liquids near the printer. If you do accidentally spill liquid on the printer, immediately remove the mains plug from the supply socket and consult your dealer.

8. Do not block or cover the ventilation slots in the cabinet.

9. Do not use or store the printer in excessively hot, cold, damp, or dusty areas.

Contents

Chapter 1
Open the box....

How to wire up the mains plug
Preparing the printer
Fitting the ink ribbon
Connecting the printer to your computer
Loading the paper
How the controls work
First steps in printing

Chapter 2
Simple printing exercises....

Printing and listing in BASIC and CP/M
AMSTRAD BASIC aids to printer operation
The print buffer
How to print 'foreign' characters
The DIP switches
How to change to an alternative typeface
Control codes

Chapter 3
Selecting print styles....

Choice of styles
Cancelling your choice
Combined styles
Illegal combinations

Chapter 4
Print formatting control....

Print head movement
Form feed
Margin setting
Page length setting
Perforation skipping
Tabulation
Paper feed rate adjustment

Chapter 5
Graphics printing....

Introduction
Single, double, and quadruple density graphics
Bit image graphics modes

Chapter 6
Extra functions....

Incremental print
8 bit characters
Control code printing
Reset, paper-out, and bleeper
Deleting characters from the buffer
Print head control
International character sets
User defined characters
Hexadecimal dump

Chapter 7
For your reference....

Reference information for the user, including:

Technical specification
Interface
Signal timing
DIP switch functions

Appendix 1
Table of control codes

Appendix 2
Some programs for AMSTRAD computer users....

'Electric typewriter'
'Text screen dump'
'Graphics screen dump'

Appendix 3
Notice for USA users

Appendix 4
Index

Chapter 1
Open the box....

Subjects covered in this chapter:

- How to wire up the mains plug
- Preparing the printer
- Fitting the ink ribbon
- Connecting the printer to your computer
- Loading the paper
- How the controls work
- First steps in printing

Unpacking your printer

In addition to the DMP2000 printer and this manual, you will also find a small box within the packing carton containing the ink ribbon, *PL-1 printer cable (for AMSTRAD computers), and the paper guide bar. Make sure that you do not accidentally discard any of these items during unpacking.

*Not supplied in some countries.

Fitting a mains plug to your printer

The DMP2000 operates from a 220-240 Volt 50Hz AC mains supply. Fit a proper mains plug to the mains lead of the DMP2000. If a 13 Amp (BS1363) plug is used, a 3 Amp fuse must be fitted. The 13 Amp fuse supplied in a new plug must NOT be used. If any other type of plug is used, a 5 Amp fuse must be fitted either in the plug or adaptor or at the distribution board.

WARNING: THIS APPARATUS MUST BE EARTHED

IMPORTANT - The wires in the mains lead are coloured in accordance with the following code:

Green-and-yellow	**: Earth**
Blue	**: Neutral**
Brown	**: Live**

As the colours of the wires in the mains lead of this apparatus may not correspond with the coloured markings identifying the terminals in your plug, proceed as follows:

The wire which is coloured **GREEN-AND-YELLOW** must be connected to the terminal in the plug which is marked by the letter '**E**' or by the safety earth symbol ⏚ or coloured green or green-and-yellow.

The wire which is coloured **BLUE** must be connected to the terminal which is marked with the letter '**N**' or coloured black.

The wire which is coloured **BROWN** must be connected to the terminal which is marked with the letter '**L**' or coloured red.

Disconnect the mains plug from the supply socket when not in use.

Do not attempt to remove any screws, nor open the casing of the DMP2000. Always obey the warning on the rating label located on the underside of the unit:

WARNING

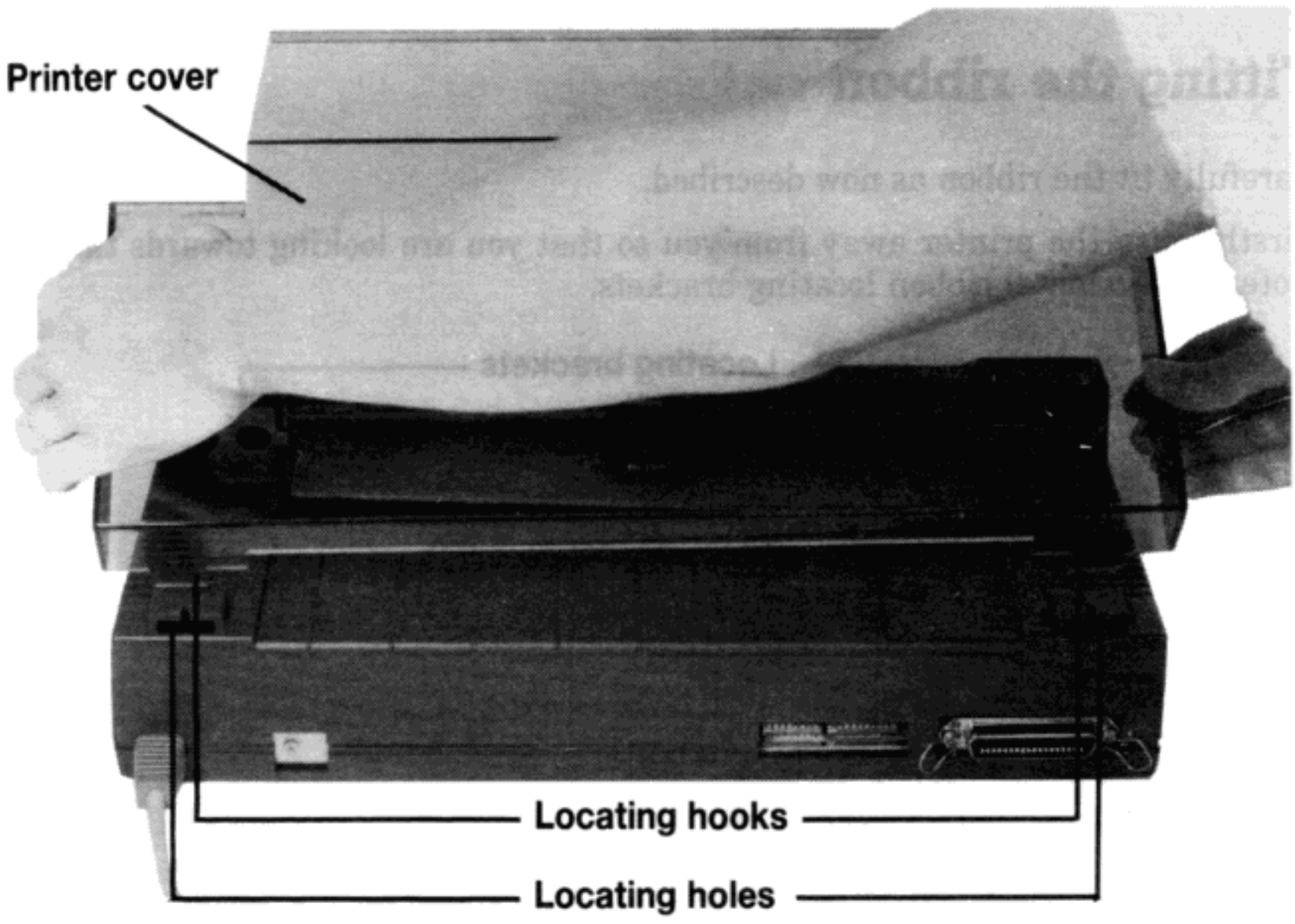
LIVE PARTS INSIDE. DO NOT REMOVE ANY SCREWS.

Preparing your printer

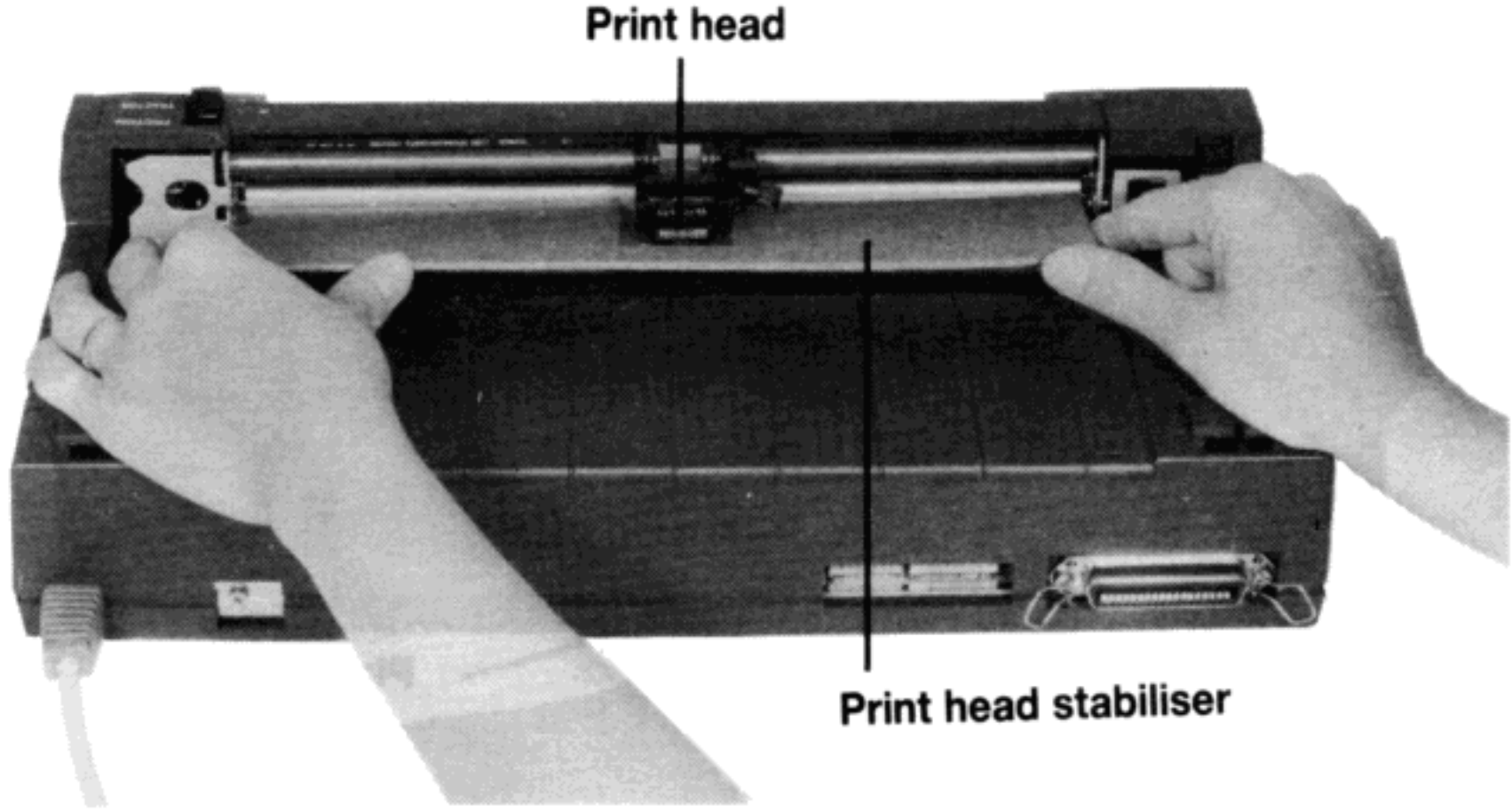
Now that you've fitted the mains plug, you'll no doubt be anxious to switch on and start printing.... **DON'T!** There are a number of important preparations yet to be made, and it is vital that you carry out the following instructions first:

Removing the print head stabiliser

Firstly, remove the clear plastic cover at the top of the printer by hinging the cover backwards, then lifting it away from the printer. (To replace the cover, simply lower the locating hooks at the bottom of the cover into the holes towards the rear of the printer.)



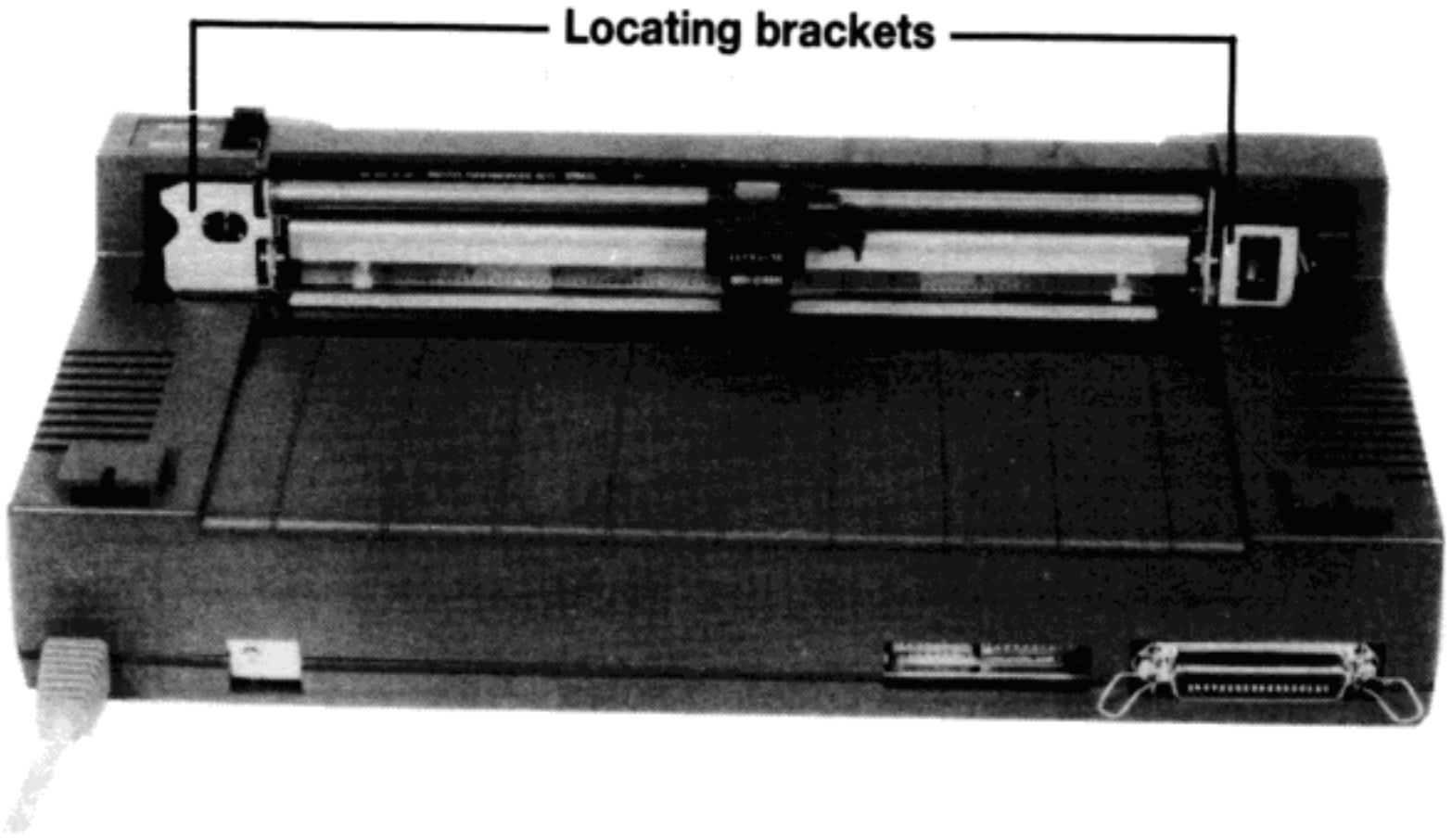
Now remove the cardboard print head stabiliser by sliding it towards the rear of the printer. Also remove the cardboard inserts at the front of the printer and on top of the print head.



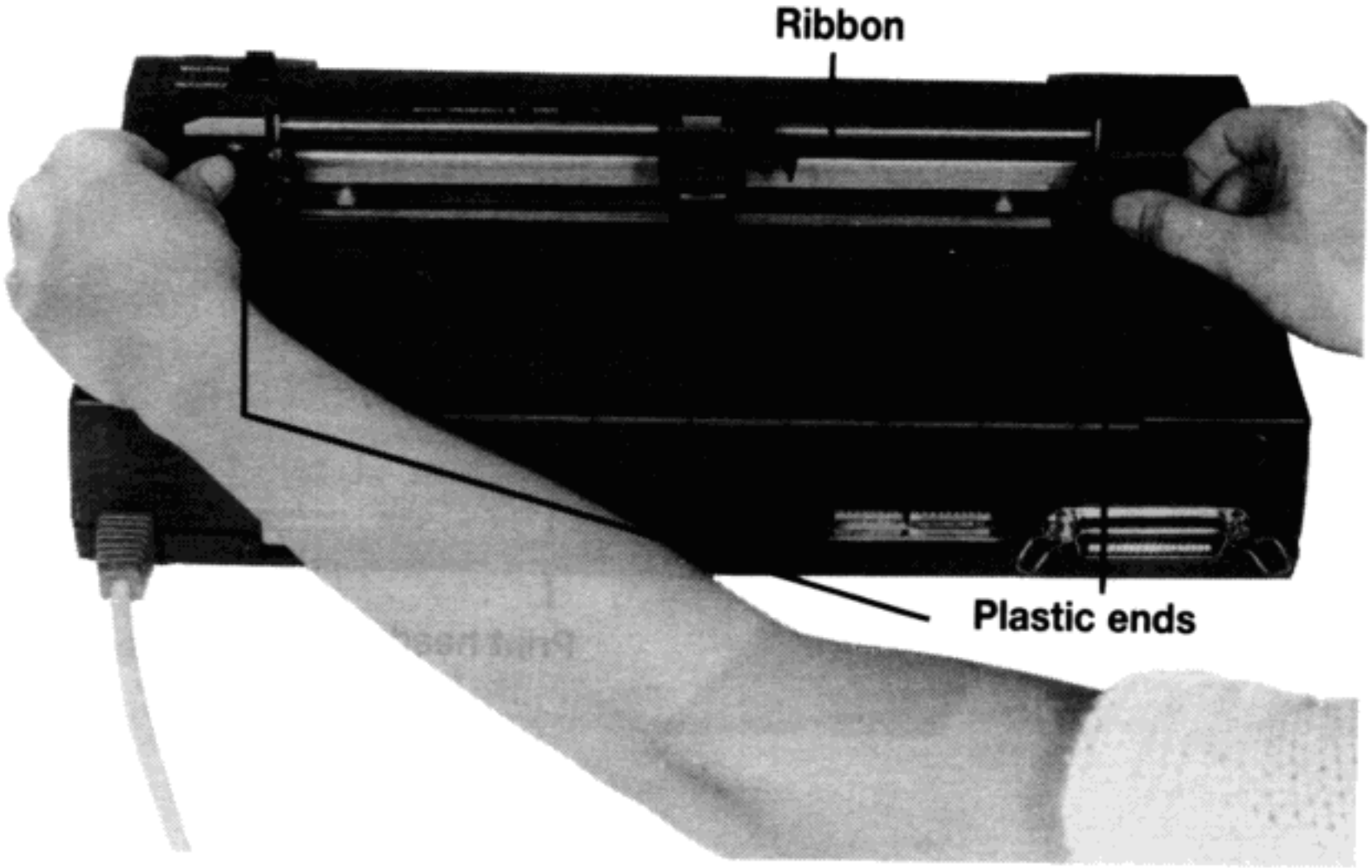
Fitting the ribbon

Carefully fit the ribbon as now described.

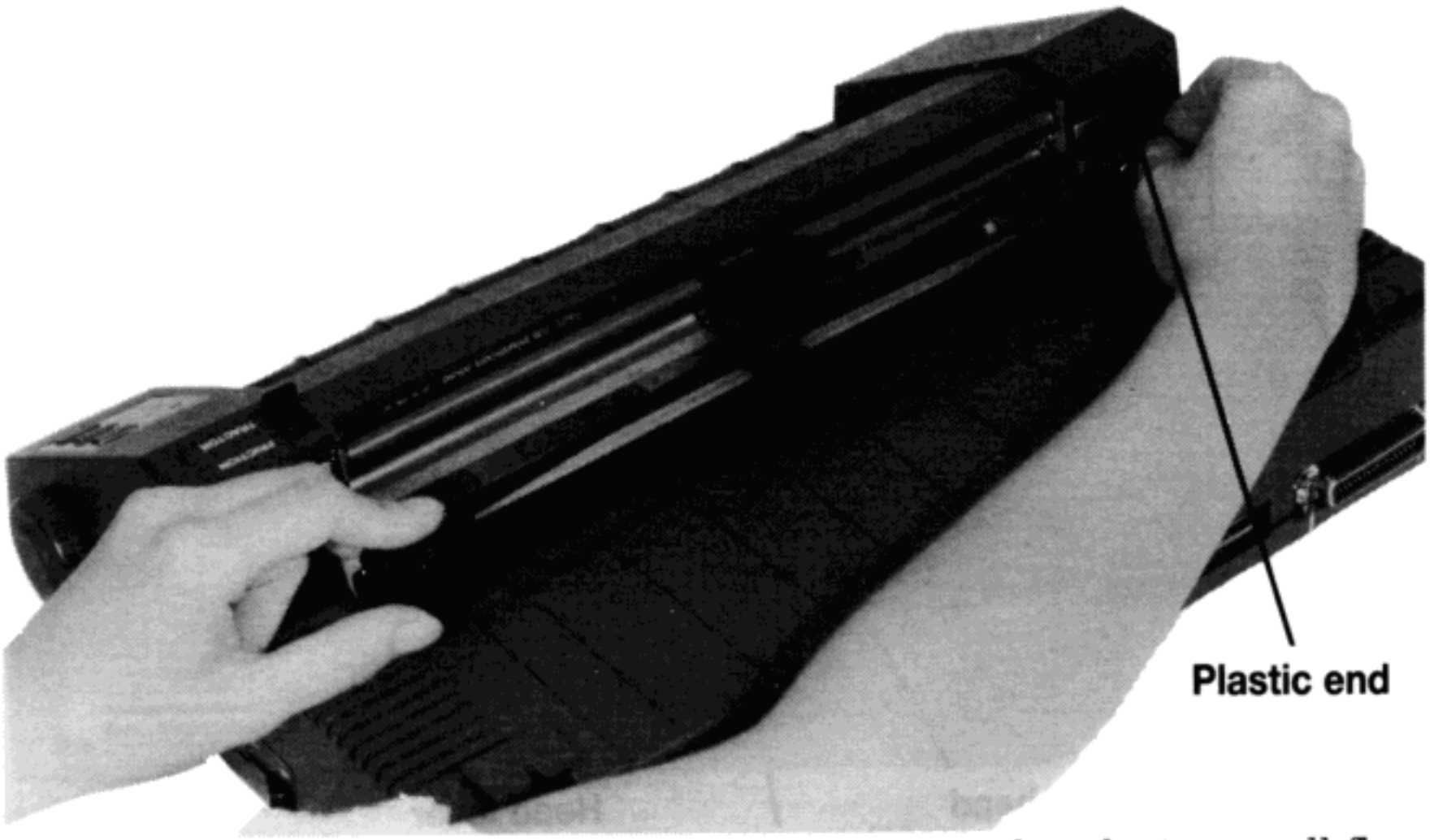
Firstly, turn the printer away from you so that you are looking towards the rear. Note the two metal ribbon locating brackets.



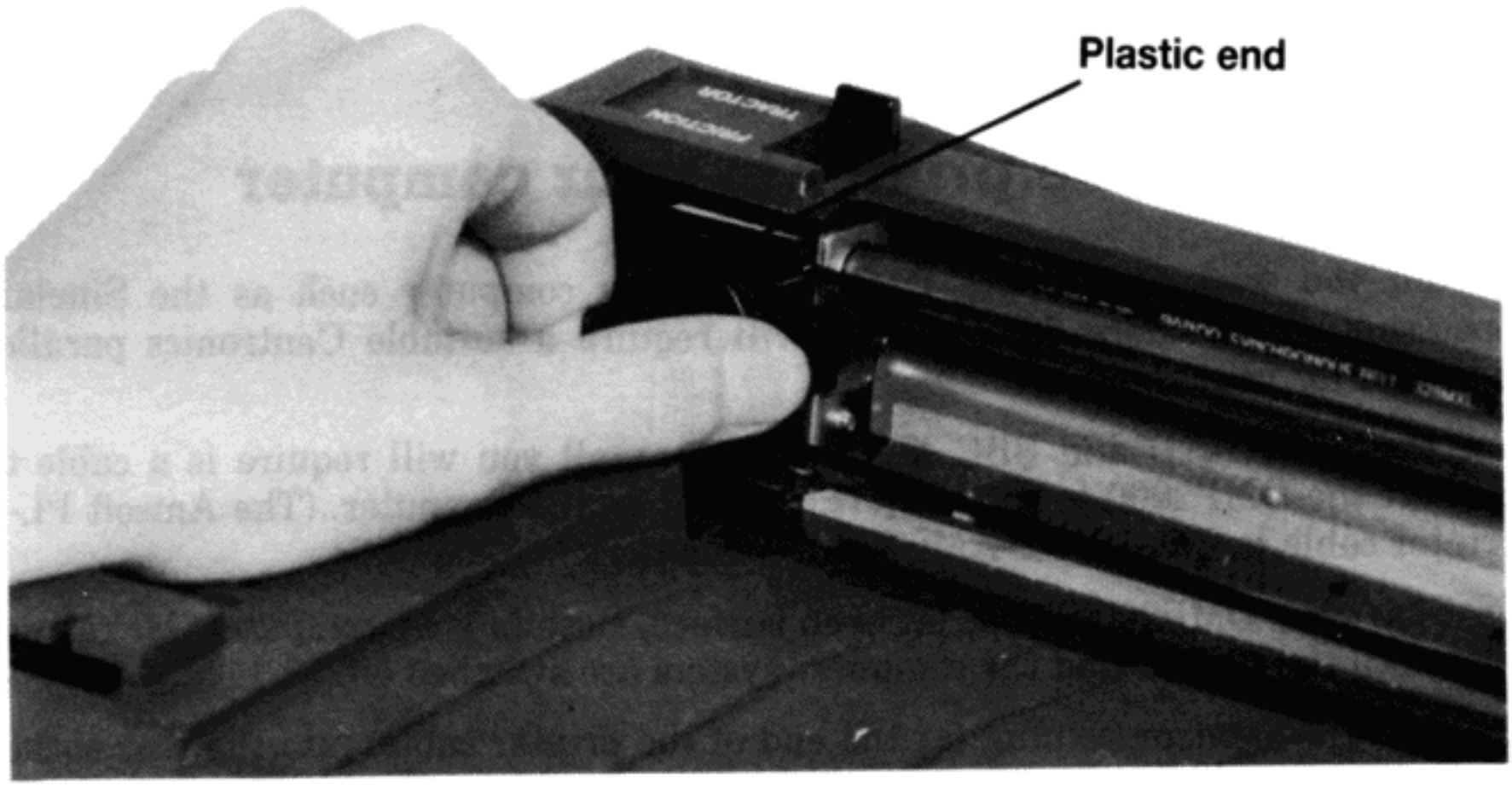
Carefully remove the ribbon from its packing, and prepare to fit the two plastic ends of the ribbon assembly into the metal locating brackets on the printer.



Start by fitting the plastic end which is in your right hand. Place the bottom of the plastic end into the square hole in the locating bracket. Now clip the top of the plastic end fully into position.



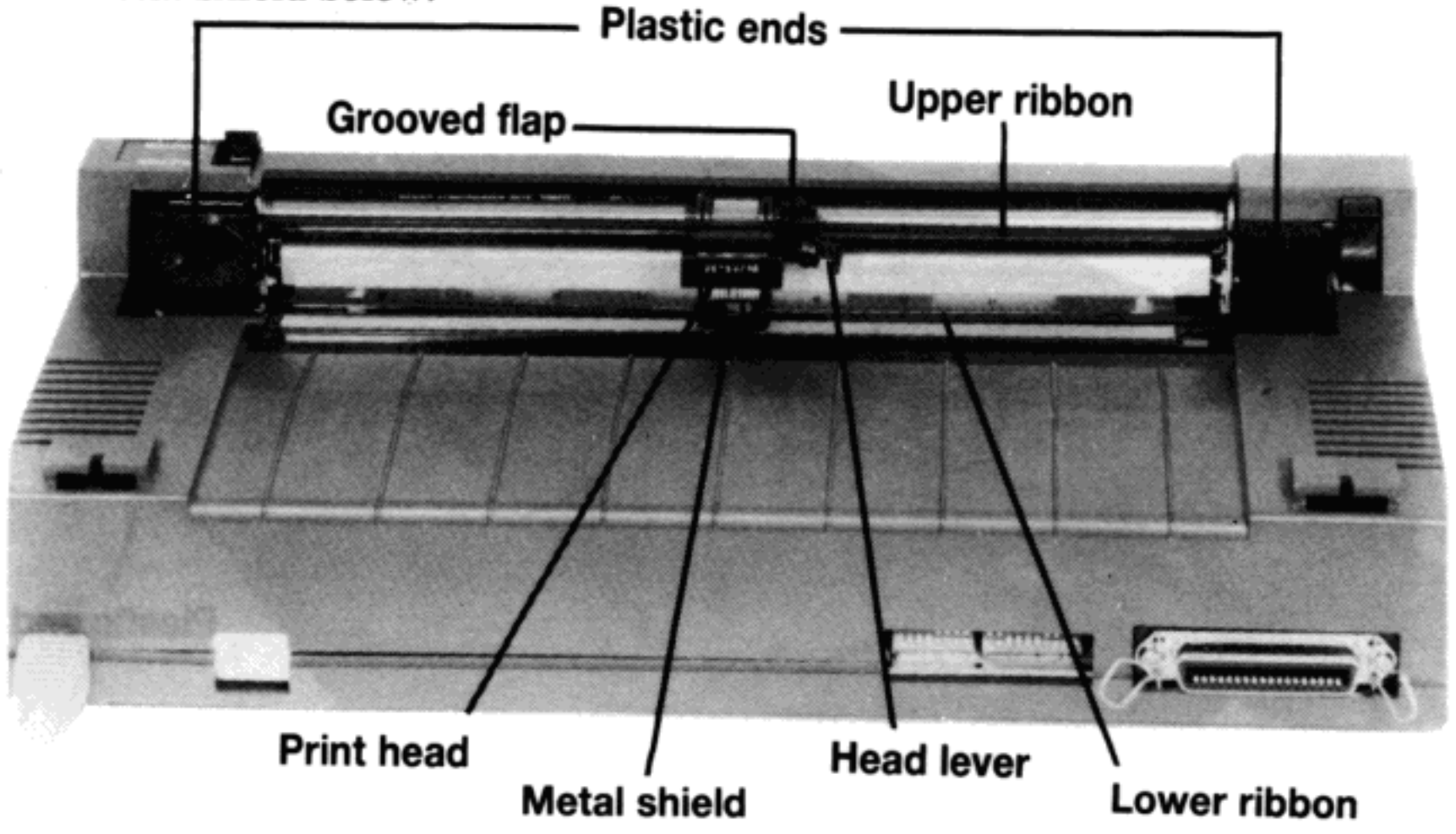
Take hold of the other plastic end. Gently pinching together the two small flaps, place the plastic end fully home into the other locating bracket, and release the flaps.



Both plastic ends should now be locked fully into place.

The ribbon which runs between the plastic ends must now be correctly positioned. Gently raise the small head lever to the right of the print head, then slide the upper ribbon under the grooved flap above. Now release the head lever.

Finally, slide the lower ribbon in between the bottom of the print head and the small metal shield below.



Ribbon removal and replacement

Should you wish to remove or replace the ribbon at any time, simply reverse the fitting process just described.

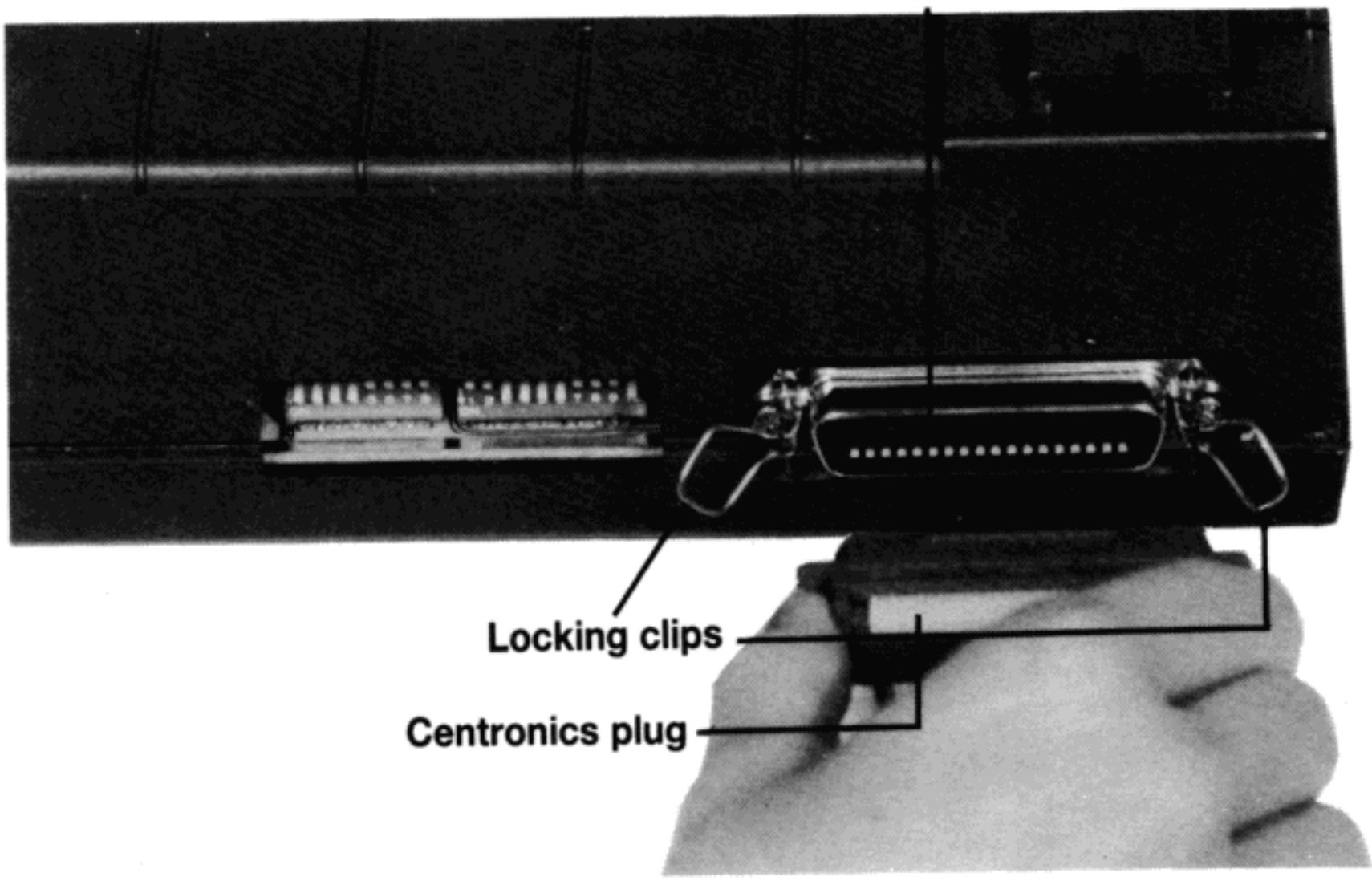
Connecting the printer to your computer

Should you wish to use the DMP2000 with a computer such as the Sinclair Spectrum, Commodore 64, etc., you will require a suitable Centronics parallel interface for your computer.

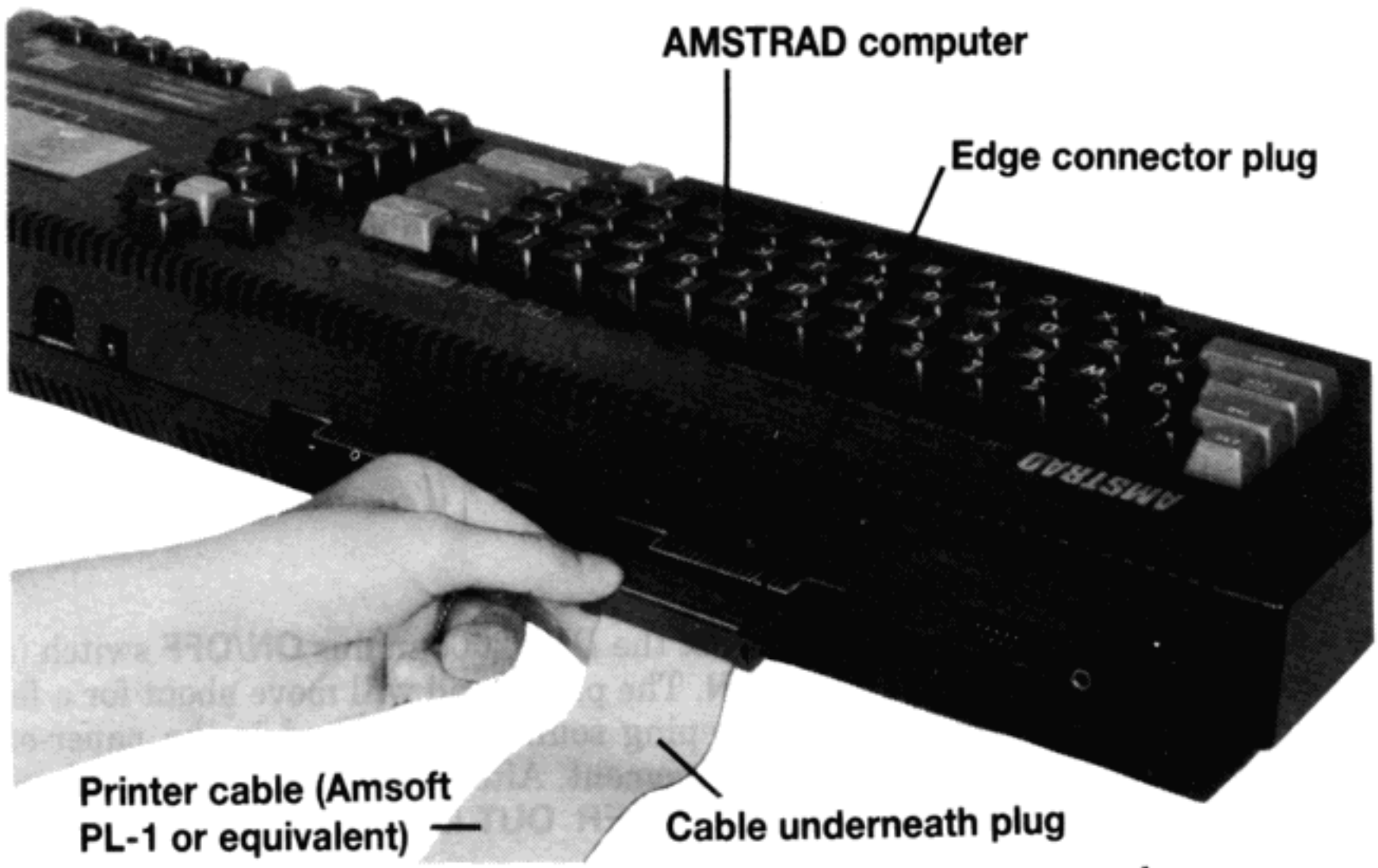
For the AMSTRAD and BBC Microcomputers, all you will require is a cable to connect the DMP2000 to the **PRINTER** socket on the computer. (The Amsoft PL-1 printer cable is suitable for AMSTRAD computers.)

Position your DMP2000 on a flat even surface, close to your computer. Make sure that both the printer and the computer system are switched off.

Connect the Centronics plug (at the end of the printer cable) into the rear socket on the printer. If the Centronics plug has a cut-out in each side, you may fasten the printer's locking-clips into the cut-outs.



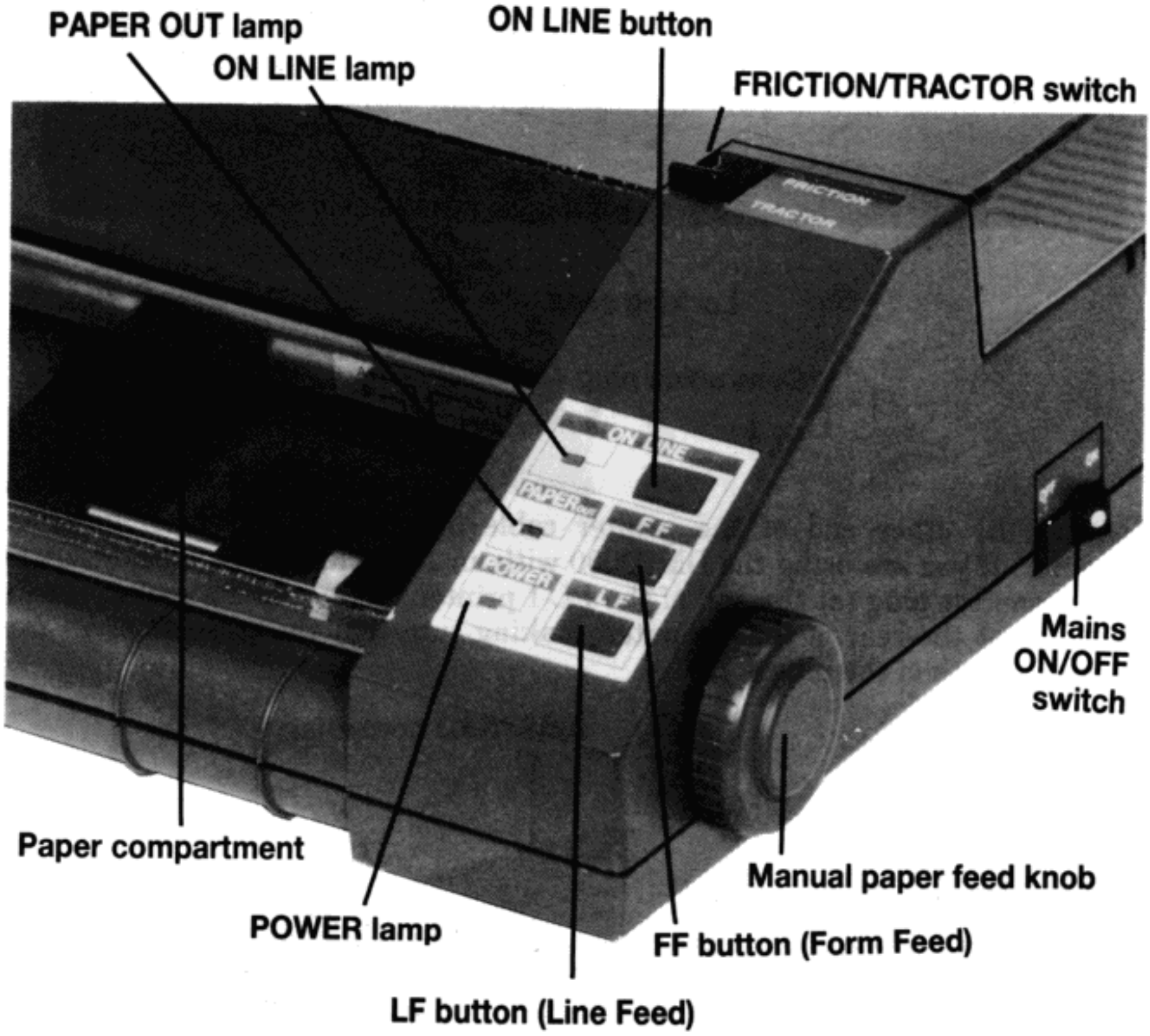
Connect the other end of the printer cable (or parallel interface) into your computer. If you are using the DMP2000 with an AMSTRAD computer, insert the edge connector plug (at the end of the PL-1 printer cable) into the socket marked **PRINTER** at the rear of the computer, making sure that the cable emerges from the underneath of the plug.



You may now replace the printer cover and turn the printer towards you.

Getting to know your printer

Take a good look at your printer. You'll be switching it on very shortly now, and it's as well to get acquainted with the names of all the buttons, knobs, switches, and lamps beforehand.



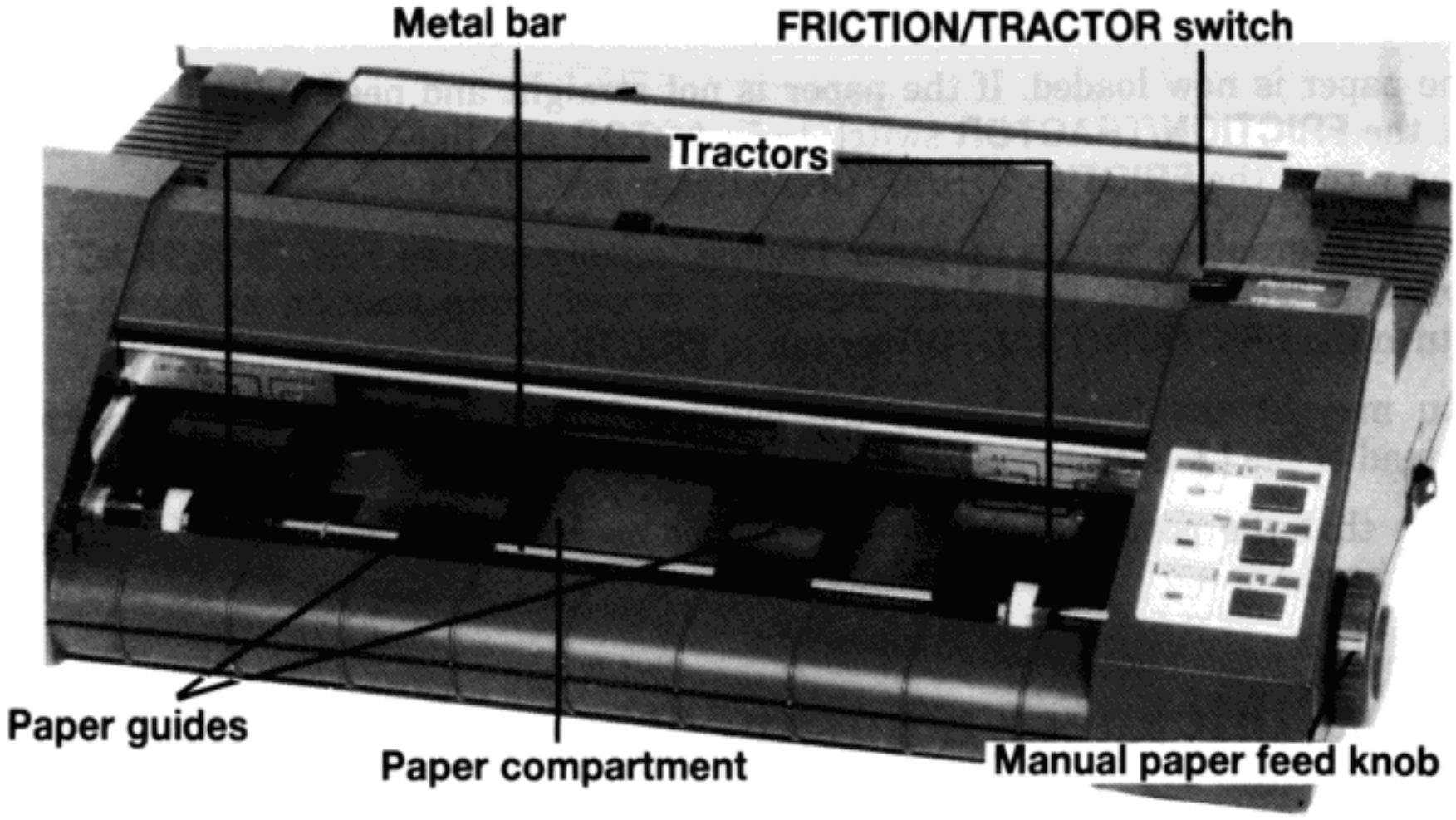
Switching on

Switch on your computer system, then set the DMP2000 mains **ON/OFF** switch (at the right hand side of the printer) to **ON**. The print head will move about for a few seconds, and you will then hear a bleeping sound. This sound is the paper-out alarm, but do not worry about it for the moment. After the alarm has stopped, you will note that the **POWER** lamp and the **PAPER OUT** lamp are both on.

All that remains to do now is load the paper, then you can start printing.

Loading the paper

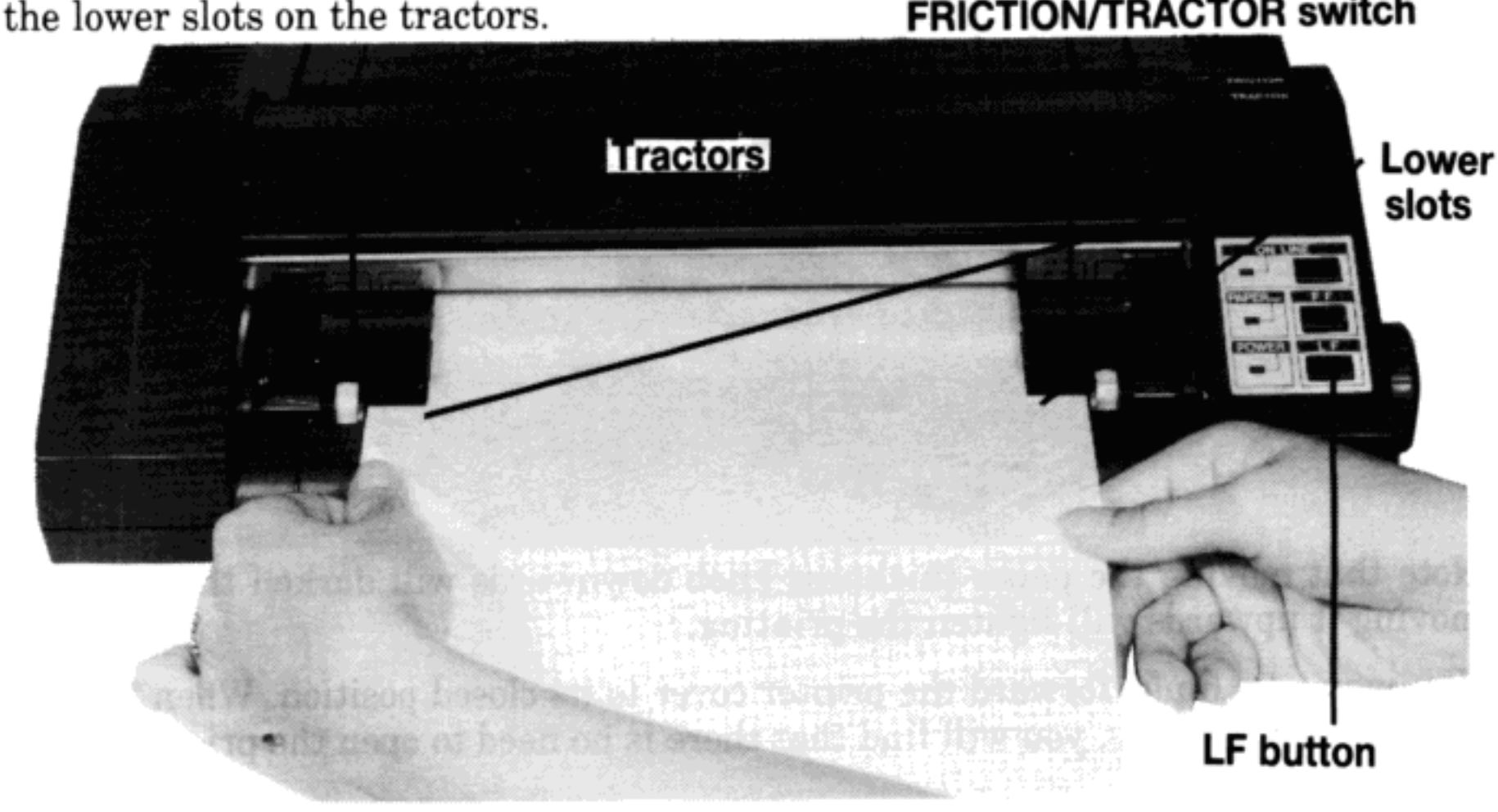
Hinge back the printer cover to its open position. Look inside the paper compartment, and you will see 4 sliding plastic blocks attached to a metal bar.



The two outer plastic blocks are called tractors, and they may be moved left or right to accomodate the size of paper you are using.

The DMP2000 may be used with plain paper or tractor feed paper (fan fold paper with holes at the side). 'How to load tractor feed paper' will be dealt with at the end of this chapter.

For now, take a piece of plain (A4 or similar) paper and slide it fully home into the lower slots on the tractors.



Make sure that the **FRICTION/TRACTOR** switch is set to **FRICTION**.

Now, gently feeding the paper into the printer, hold down the **LF** button, and the printer will gradually 'take in' the paper. When you see the paper coming out of the top of the printer (just underneath the print head), release the **LF** button.

The paper is now loaded. If the paper is not straight and needs aligning, simply set the **FRICTION/TRACTOR** switch to **TRACTOR** and line-up the paper by hand; then return the **FRICTION/TRACTOR** switch to **FRICTION**.

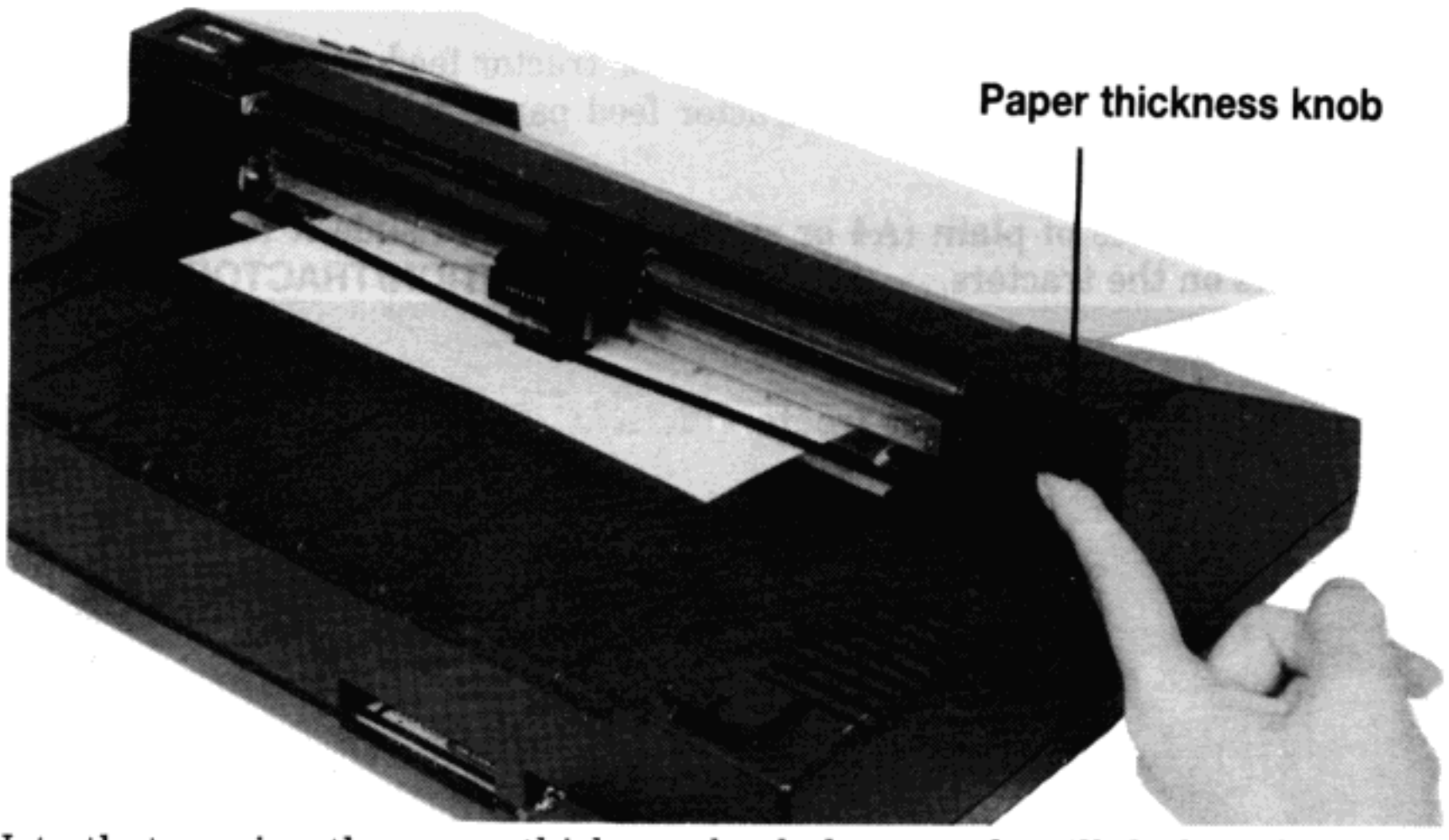
As an alternative means of paper loading, simply set the **FRICTION/TRACTOR** switch to **TRACTOR** and slide the paper straight into position by hand; then return the **FRICTION/TRACTOR** switch to **FRICTION**.

You may at any time advance the paper using the manual paper feed knob instead of the **LF** button.

(Note that the position printing takes place on the paper, is directly under the print head.)

Paper thickness adjustment

Once the paper is loaded, you may adjust the paper thickness knob to a suitable setting - upwards for thicker paper (or two sheets); downwards for thinner paper.



Note that moving the paper thickness knob downwards will darken the printing; moving it upwards will lighten the printing.

You may now hinge forward the printer cover to its closed position. When loading plain paper in future, you will find that there is no need to open the printer cover.

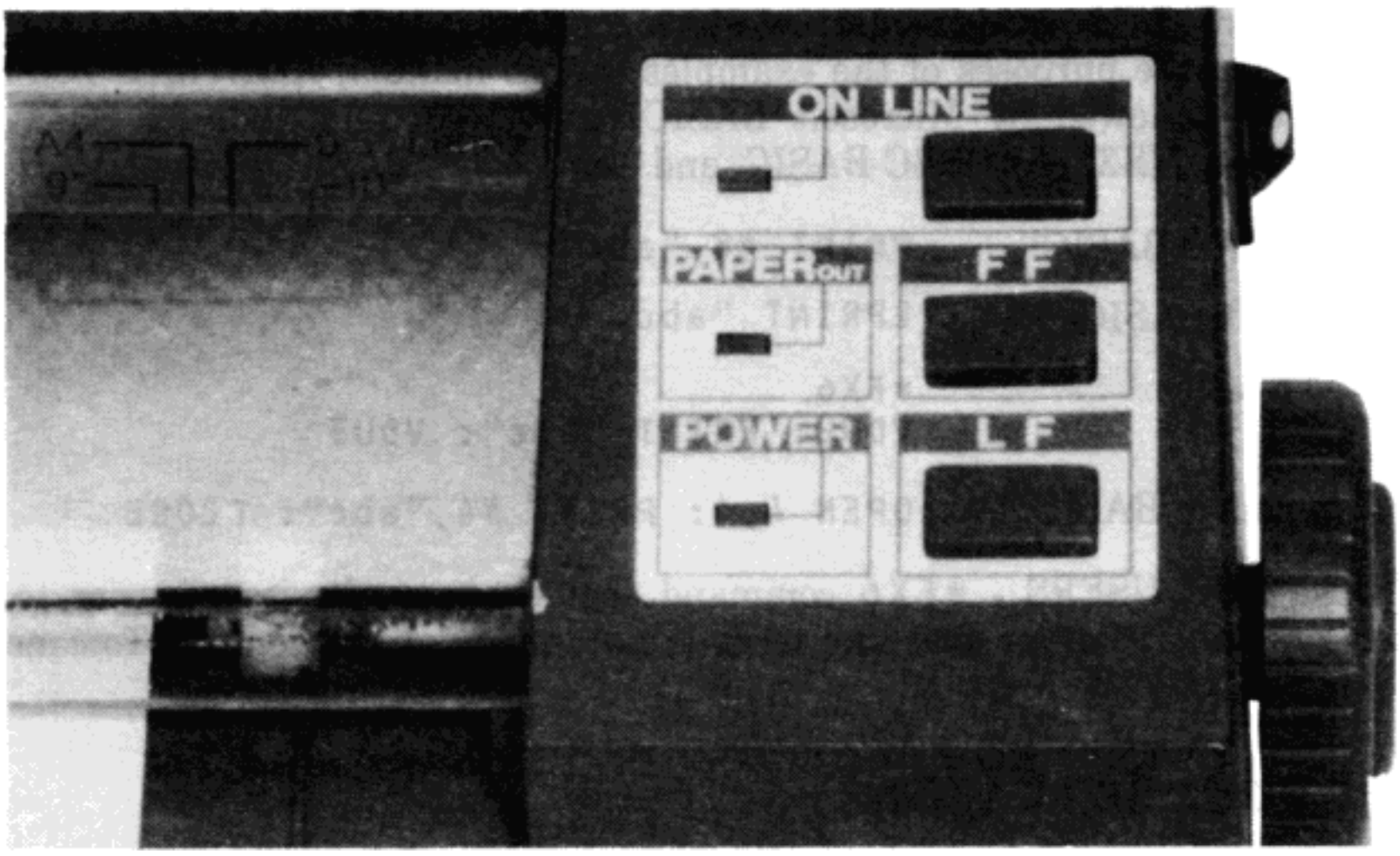
Let's print!

The DMP2000 incorporates a 'self test' facility whereby all the ASCII characters are automatically printed. To try this out, first set the mains **ON/OFF** switch to **OFF**. Now, holding down the **LF** button, set the mains **ON/OFF** switch to **ON**. Release the **LF** button, and self test printing will start and continue until either the paper runs out, or the mains is switched off.

After four or five lines of self test printing, set the mains **ON/OFF** switch to **OFF**.

Have a look at what's been printed. If the characters have not been printed clearly and evenly, check that you have installed the ribbon and paper correctly.

Now switch the printer back on again. Assuming that you still have some paper loaded in the printer, notice that when you switch on this time, the paper-out alarm does **NOT** come on, and the **ON LINE** lamp is lit.



What does ON LINE mean?

ON LINE means that the printer is ready to print (as soon as it receives a command from the computer). Note that when the printer is **ON LINE**, you won't be able to move the paper using the **LF** or **FF** buttons. What you must then do is press the **ON LINE** button once, so that the **ON LINE** lamp goes out. This is known as setting the printer 'off line', and you will then be able to advance the paper using, for example, the **LF** button. So briefly, the rule is: 'on line' to print; 'off line' to stop printing and move the paper.

The FF button

The **FF** (Form Feed) button may be used to advance the paper through the printer by one whole page.

To demonstrate, set the printer off line, then press the **FF** button. Watch what happens to the paper.

Form feed is very useful if, for example, you've just printed a letter to somebody and you wish to 'run it out' from the printer.

If you are using tractor feed paper (explained at the end of this chapter) you may use the **FF** button to advance the paper by one page's length.

Which computer have you got?

The BASIC commands that you should use to tell the computer to send output to the printer will obviously depend upon which computer you have connected to the DMP2000. For the purposes of the examples in this manual, AMSTRAD BASIC will be used, followed by Microsoft BASIC (use these commands for Sinclair Spectrum, Oric, MSX, etc.), BBC BASIC, and Commodore BASIC. For example:

AMSTRAD BASIC:
```
PRINT #8,"abc"
```

Microsoft BASIC:
```
LPRINT "abc"
```

BBC BASIC:
```
*FX6
VDU2: PRINT "abc": VDU3
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,"abc": CLOSE
```

NOTE TO BBC USERS - `*FX6` command should always be issued prior to a printer command in order that line feeds are automatically generated. To cancel `*FX6` use `*FX6,10`.

Print your first word

Load a piece of paper into the printer.
Press the **ON LINE** button, and make sure that the **ON LINE** lamp is lit.
Now instruct the computer to send a word to the printer:

AMSTRAD BASIC:
```
PRINT #8,"Ivor"
```

Microsoft BASIC:
```
LPRINT "Ivor"
```

BBC BASIC:
```
*FX6
VDU2: PRINT "Ivor": VDU3
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,"Ivor": CLOSE
```

Whatever you just typed (inside the quotation marks) should have immediately been printed out by the DMP2000. If it hasn't, then check that the cable between the printer and the computer is correctly fitted at both ends, that the paper is correctly loaded, and that the printer is on line.

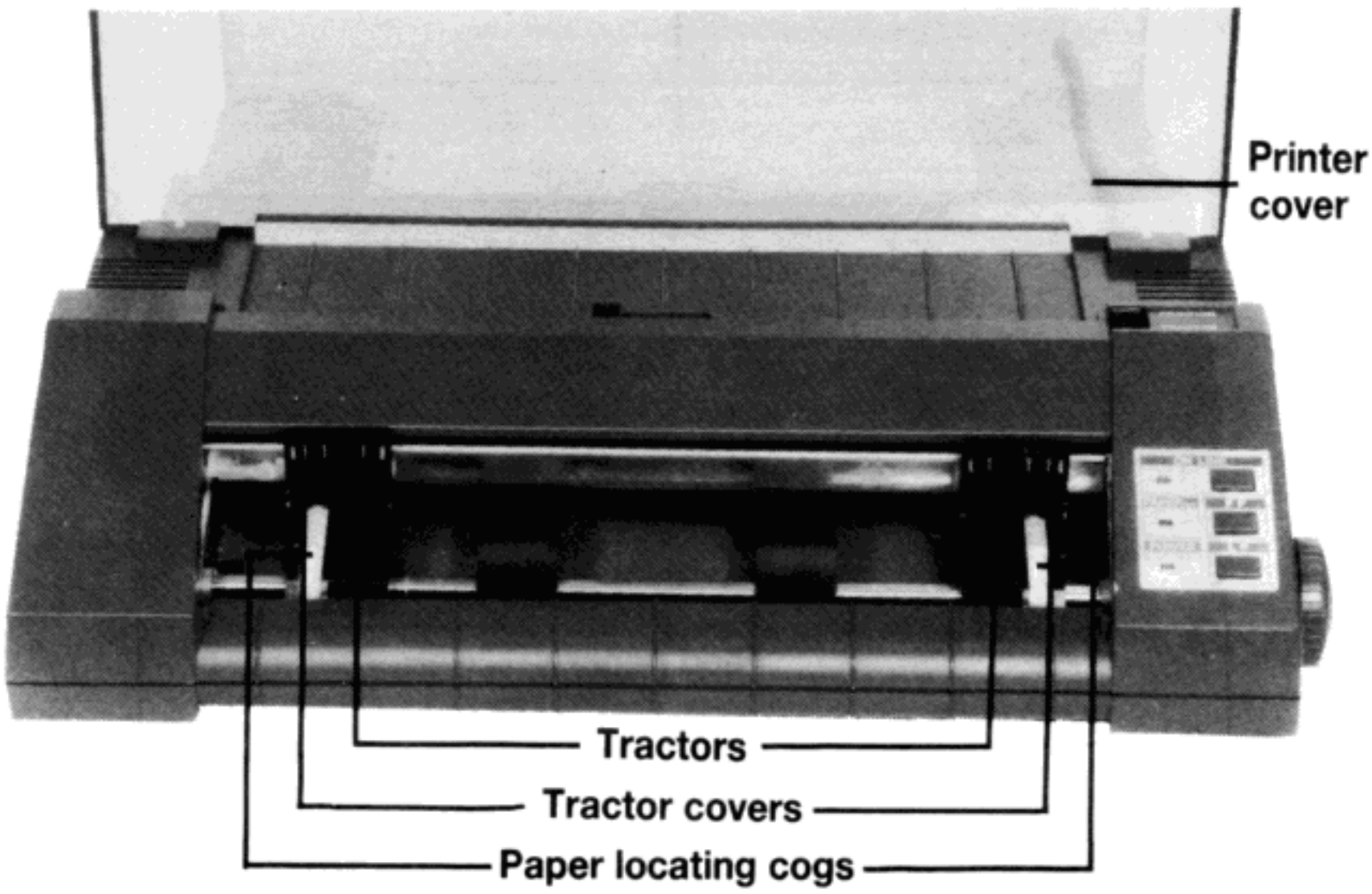
Well, you have now set up the DMP2000 and printed your first word. The next chapter goes on to describe some of the simpler printing functions, and will introduce you to some of the alternative styles of printing that the DMP2000 is capable of.

Finally in this chapter....

How to load tractor feed paper

Tractor feed paper is useful if you want to print program listings or long continuous texts. Load the paper as follows:

Open the printer cover, and hinge back the flaps on top of the tractors (these are known as the tractor covers):

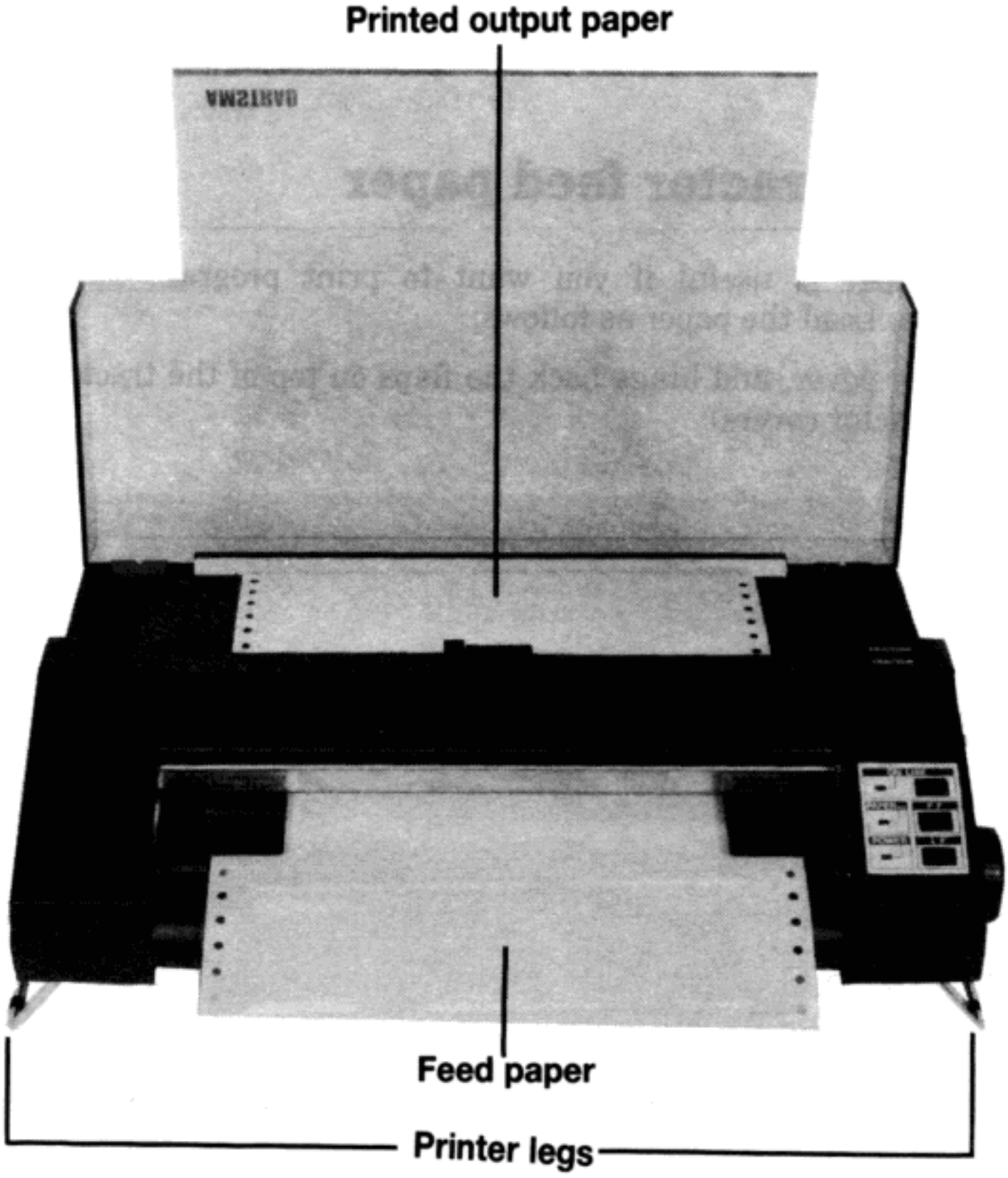


When you have opened the tractor covers, the paper locating cogs will be fully exposed. Now line up the paper holes over the cogs, and if necessary, move the tractors to accomodate the paper being used.

Place the paper over the cogs and close the tractor covers. Set the **FRICTION/TRACTOR** switch to **TRACTOR**, and carefully feed the paper into the printer (using the **LF** button or the manual paper feed knob) ensuring that the paper feeds in smoothly without creasing or tearing at the tractors.

When you see the paper coming out of the top of the printer (just underneath the print head) the paper is loaded, and you may then close the printer cover.

If you wish, you may position the feed paper underneath the printer by extending the printer legs.



The paper guide bar may optionally be fitted between the left and right front feet of the printer.

Make sure that the printed output paper is allowed to run out freely, and pile directly behind the printer. The paper should fold one way, then the other, as it piles.

Chapter 2
Simple printing exercises....

Subjects covered in this chapter:

Printing and listing in BASIC and CP/M
AMSTRAD BASIC aids to printer operation
The print buffer
How to print 'foreign' characters
The DIP switches
How to change to an alternative typeface
Control codes

General printing

Having set up the DMP2000 and printed a word or so, I hope that you'll by now have gathered that to send text to the printer, you simply direct it to the appropriate 'stream' (for example, `#8` in AMSTRAD BASIC). This not only applies to constant strings (such as that shown in the previous example), but also to any combination of string variables, numbers, numeric variables, or control codes (more about these later).

It can be seen, therefore, that if you repeatedly input a line of text into a string variable or string array, then print out the current value of the string, you have the basis for a simple typewriter program. For example (AMSTRAD BASIC only):

```
10 CLS
20 LINE INPUT "",a$
30 PRINT #8,a$
40 GOTO 20
```

Listing a program

BASIC programs may be listed to the printer. Simply type in:

AMSTRAD BASIC: `LIST #8`

Microsoft BASIC: `LLIST`

BBC BASIC:

```
*FX6
VDU2
LIST
VDU3
```

Commodore BASIC:

```
OPEN 4,4
LIST #4
CLOSE
```

You may list a specific line (or range of lines) to the printer as you would to the screen. However, you should bear in mind that in AMSTRAD BASIC, you must state the #8 stream expression AFTER the line number(s), for example:

```
LIST 20-50,#8
```

NOTE - From here on, the BBC BASIC command *FX6 will not be shown.

AMSTRAD BASIC aids to printer operation

The following section applies only if you are using your printer with an AMSTRAD computer. If you aren't, then skip to the next section ('Printing under CP/M').

Print formatting

BASIC print format commands such as PRINT USING, PRINT TAB, *and* PRINT SPC *can be directed to the printer simply by adding the* #8 *stream director. Equally, the use of the semicolon and comma in* PRINT #8 *statements will enable successive expressions to be printed adjacent to one another, or in adjacent print zones. The* ZONE *command applies to both the screen and the printer. Example commands:*

```
10 ZONE 20
20 PRINT #8,"text ";"semicolon","comma"
30 PRINT #8,TAB(30)"column 30"
40 PRINT #8,"leave";SPC(10)"ten spaces"
50 PRINT #8,USING "**$##.## to the pound";1.3975
```

The WIDTH command

You may use the WIDTH *command to specify the number of characters per line (in the range 1 to 254) to be printed (the computer defaults to 132). Example command:*

```
10 WIDTH 50
20 PRINT #8,STRING$(200,42)
```

The command WIDTH 255 *selects 'unlimited' line wrapping, and you should use this setting when performing graphics printing (explained later in this manual).*

Note that the `MODE` *in which the computer is operating (i.e. 20, 40, or 80 column) bears no relationship to the size or number of characters per line on the printer.*

The POS function

The form `POS(#8)` *may be used to determine the next print-position on the paper. Note that this does not necessarily correspond to the physical position of the print head. Example command:*

```
10 CLS
20 PRINT #8,"123456789";
30 PRINT POS(#8): REM display the print-position
   on the screen
40 PRINT #8: REM flush buffer
```

Printing under CP/M

When the CP/M operating system is loaded, pressing **[CTRL]P** will echo screen output to the printer until **[CTRL]P** is pressed again to cancel the assignment.

Note that pressing **[CTRL]P** does NOT produce a printed character on the screen.

Example:

```
A>[CTRL]P TYPE FILENAME.TXT
```

Typing the above command (after the `A>` prompt) will send the contents of the file `FILENAME.TXT` to both the screen and the printer. Similarly, typing `DIR` after **[CTRL]P** will output the disc directory (catalog) to the printer.

Don't forget - until you cancel the **[CTRL]P** assignment, everything sent to the screen will be echoed to the printer.

The print buffer

Before printing any characters on the paper, the printer stores incoming information (from the computer) in an area of memory called the print buffer. In previous examples, the reason that the printer has printed out everything that you've instructed it to (rather than holding it in the buffer) is because each `PRINT` statement that you have issued has been automatically followed by a carriage return (this is executed by default) which has the action of emptying (or 'flushing') the buffer.

To explain the above, let's forget the printer for a moment and concentrate on the screen.

Comparing the results of the following two programs:

```
10 PRINT 123
20 PRINT 456
30 PRINT 789
run
 123
 456
 789
```

....and....

```
10 PRINT 123;
20 PRINT 456,
30 PRINT 789
run
 123 456     789
```

....you can see that the semicolon and comma at the end of lines `10` and `20` suppress the 'carriage return' on the screen.

Now instruct your computer to print a string (or a number) followed by a semicolon or comma, to the printer. Note what happens (or what doesn't happen!).

You'll probably have gathered that the reason nothing has been printed out is because you've suppressed the carriage return, and your expression to be printed is held inside the print buffer. The buffer will be flushed either when it receives the next carriage return, or when it's so full that it needs to spill out its contents to make room for some more. Now type in:

AMSTRAD BASIC:	`PRINT #8`
Microsoft BASIC:	`LPRINT`
BBC BASIC:	`VDU2: PRINT: VDU3`
Commodore BASIC:	`OPEN 4,4: PRINT #4: CLOSE`

Although you've instructed the computer to print 'nothing' to the printer, the action of sending the carriage return alone is enough to flush the previous contents out of the buffer.

Foreign characters

What you see on the screen is not always what you get on the printer! To illustrate this point, first bear in mind that on conventional typewriters, **[SHIFT]3** produces a £ pound sign. Typing **[SHIFT]3** on your computer, however, will produce a # hash symbol. If you now type in:

AMSTRAD BASIC:	`PRINT #8,"#100"`
Microsoft BASIC:	`LPRINT "#100"`
BBC BASIC:	`VDU2: PRINT "#100": VDU3`
Commodore BASIC:	`OPEN 4,4: PRINT #4,"#100": CLOSE`

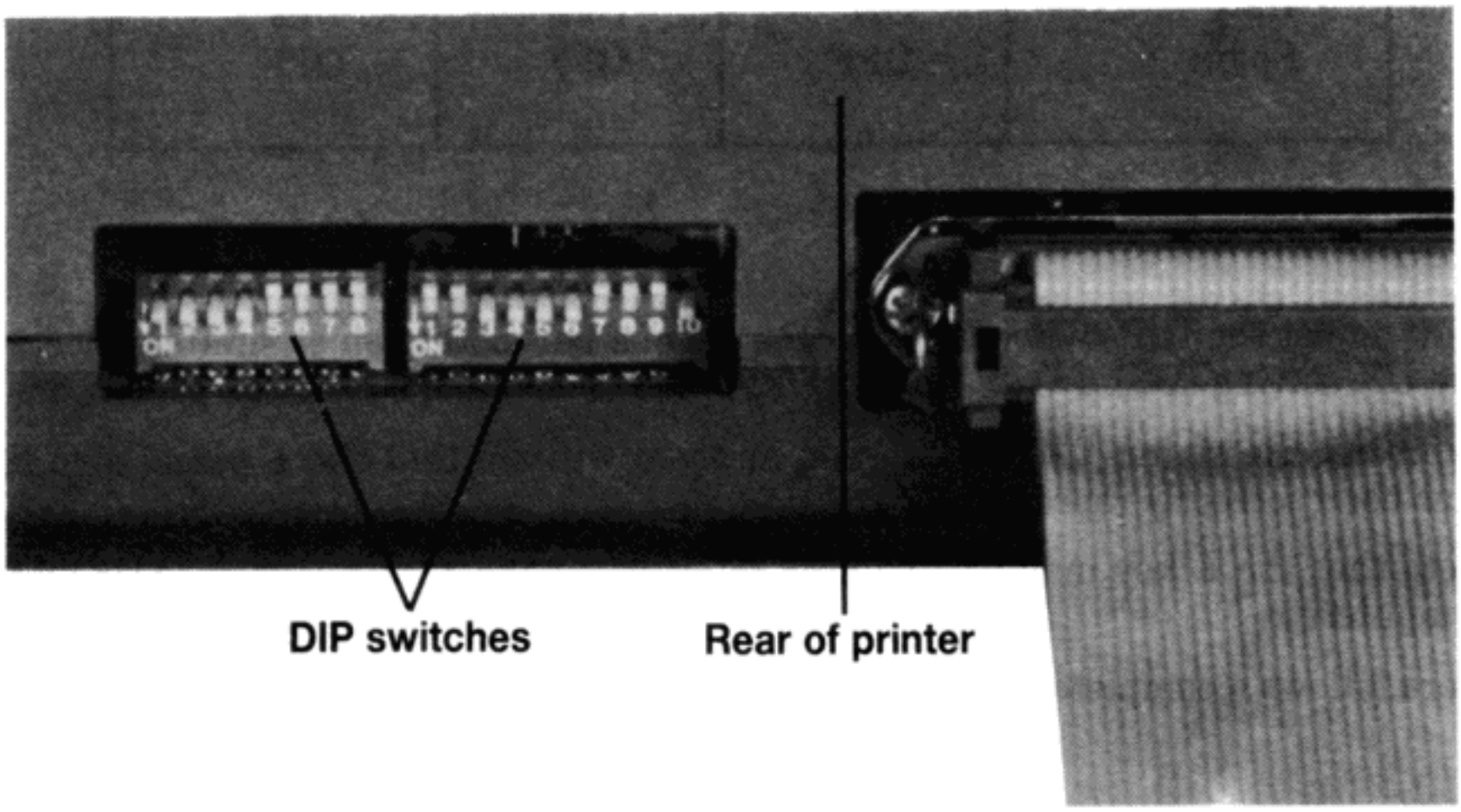
....you will see that the # hash symbol has also been reproduced by the printer. This is because the DMP2000 is factory set to reproduce the standard (American) ASCII character set.

Fortunately, for those of you who wish to print £ pound instead of # hash, the printer may be adjusted to do so.

The way that you may set the DMP2000 to print 'foreign' characters such as the £ pound sign is by using a series of miniature switches (called DIP switches) at the rear of the printer.

How to adjust the DIP switches

Always switch the printer off before adjusting the DIP switches.



If you look carefully, you will note that there are 2 blocks (or banks) of switches. The first bank (called DS1) contains 8 switches, while the second bank (called DS2) contains 10. Each of the switches are numbered, and on the corner of each bank, you will see the word ON (showing you the direction to switch on).

The 3 switches that control the foreign print characters are numbers 1, 2, and 3 on the first bank (DS1). From now on, we will refer to these switches as DS1-1, DS1-2, and DS1-3.

The other DIP switches (DS1-4 to DS1-8, and DS2-1 to DS2-10) will be dealt with later in this manual.

The following table indicates the various DIP switch settings for the required foreign characters:

COUNTRY	DS1-1	DS1-2	DS1-3
USA	ON	ON	ON
UK	OFF	OFF	ON
France	OFF	ON	ON
Germany	ON	OFF	ON
Denmark	ON	ON	OFF
Sweden	OFF	ON	OFF
Italy	ON	OFF	OFF
Spain	OFF	OFF	OFF

Now adjust the DIP switches (using the tip of a ball-point pen or similar object) to the UK setting (DS1-1 OFF, DS1-2 OFF, DS1-3 ON) and switch the printer on.

Print out the string "#100" again, and you will see that the £ pound sign has now been correctly reproduced.

Note that the £ key may also be used to print out the £ or # sign on AMSTRAD computers.

Here follows a complete table of the available foreign character sets:

CHARACTER CODE (HEX)

	&23	&24	&40	&5B	&5C	&5D	&5E	&60	&7B	&7C	&7D	&7E
USA	#	$	@	[	\	]	^	'	{	¦	}	~
UK	£	$	@	[	\	]	^	'	{	¦	}	~
France	#	$	à	°	ç	§	^	'	é	ù	è	¨
Germany	#	$	§	Ä	Ö	Ü	^	'	ä	ö	ü	ß
Denmark	#	$	@	Æ	ø	Å	^	'	æ	ø	å	~
Sweden	#	¤	É	Ä	Ö	Å	Ü	é	ä	ö	å	ü
Italy	#	$	@	°	\	é	^	ù	à	ò	è	ì
Spain	₧	$	@	¡	Ñ	¿	^	'	¨	ñ	}	~

How to change to an alternative typeface

The DMP2000 is capable of reproducing many combinations of different print style, or 'typeface'.

Here's one to try out. Type in:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"x";CHR$(1);"good
quality"
```

Microsoft BASIC:
```
LPRINT CHR$(27);"x";CHR$(1);"good
quality"
```

BBC BASIC:
```
VDU2,1,27,1,ASC("x"),1,1: PRINT "good
quality": VDU3
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"x";
CHR$(1);"g ood  quality":CLOSE
```

Look at the printing. You have selected the NLQ (Near Letter Quality) typeface.

(If NLQ printing hasn't been produced, check that you correctly typed in the above command, using a lower-case 'x'.)

To cancel the NLQ setting, type in:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"x";CHR$(Ø);
"standard"
```

Microsoft BASIC:
```
LPRINT CHR$(27);"x";CHR$(Ø);"standard"
```

BBC BASIC:
```
VDU2,1,27,1,ASC("x"),1,0:PRINT
"standard": VDU3
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"x";CHR$(0);
"standard":CLOSE
```

To select (enable) and cancel (disable) the NLQ setting, we have used what's known as a control code.

What is a control code?

A control code is used in order to activate a function of the computer, while the code itself is not generally printed. `CHR$(7)`, for example, is a control code that makes the computer 'beep'. Just try:

```
PRINT CHR$(7)
```

Now let's have a look at the above commands. Notice that the `PRINT` statement has four parts:

```
CHR$(27)
"x"
CHR$(1)  ....or.... CHR$(0)
```
‹string expression›

The `CHR$(27)` part is known as an 'escape code' (sometimes shortened to **ESC**), and tells the printer that what follows is NOT to be printed out, but is to be used for enabling or disabling one of the printer's functions. A sequence of instructions starting with `CHR$(27)` (such as in the above examples) is known as an 'escape sequence'.

The 'x' part of the command is the individual code-letter appertaining to NLQ operation. As you work through this guide, you will see that each particular printer function has its own code-letter.

The `CHR$(1)` or `CHR$(0)` part can be thought of as a switch, turning the particular print function on or off. As you will see from the above examples, `CHR$(1)` switches the function ON; `CHR$(0)` switches it OFF.

Finally, the ‹string expression› part contains the item(s) that you want printed. The ‹string expression› can be a constant string enclosed by double quotes (as in the above examples), a string variable (such as `A$`), a numeric constant (such as `3.142`), a numeric variable (such as `X` or `Y%`), or any combination of these.

The numerous control codes for each of the DMP2000's functions will be given in the next few chapters. You will also be shown how you may select a combination of two or more control codes together.

Chapter 3
Selecting print styles....

Subjects covered in this chapter:

Choice of styles
Cancelling your choice
Combined styles
Illegal combinations

What's available?

The DMP2000 is capable of over 100 different print style combinations. There are six main typefaces, known as:

Standard (sometimes known as 'Pica')
Mini (sometimes known as 'Elite')
Proportional
Condensed
NLQ-standard
NLQ-proportional

To these main typefaces, you can apply the following additional functions:

Subscript
Superscript
Double-strike
Italics
Bold

Finally, to any of the above combinations of typeface, you may apply:

Underline
Double-width

As you can see, there are lots of typefaces for you to choose from, and it's easy to get lost in a maze of combinations! It is perhaps worth remembering, therefore, that you can always get back to Standard typeface (with no modifications) by switching the printer off, then on again.

The printer has a built-in 'memory' of its own, so switching off or resetting the computer will NOT alter the printer's settings.

Selecting one of the main typefaces

Before we discuss all the different possible combinations of typeface and how to select them, let's start with the 6 main typefaces.

In each case, you will be shown the control codes required to select or cancel the typeface, together with an example.

Standard typeface

Standard typeface is automatically selected when the printer is first switched on, or when any combination of other typeface settings are cancelled. It is the one typeface that you do not have to explicitly 'select'.

Mini typeface

TO SELECT: **ESC M**

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"M";"this is mini"
```

Microsoft BASIC:
```
LPRINT CHR$(27);"M";"this is mini"
```

BBC BASIC:
```
VDU2,1,27,1,ASC("M"): PRINT "this
is mini": VDU3
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"M";"this
is mini": CLOSE
```

TO CANCEL: **ESC P**

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"P";"back to normal"
```

Microsoft BASIC: `LPRINT CHR$(27);"P";"back to normal"`

BBC BASIC:
```
VDU2,1,27,1,ASC("P"): PRINT "back to
normal": VDU3
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"P";"back to
normal": CLOSE
```

Proportional typeface

TO SELECT: **ESC** p + 1

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"p";CHR$(1);"this is
proportional"
```

Microsoft BASIC:
```
LPRINT CHR$(27);"p";CHR$(1);"this is
proportional"
```

BBC BASIC:
```
VDU2,1,27,1,ASC("p"),1,1: PRINT "this
is proportional": VDU3
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"p";CHR$(1);
"this is proportional": CLOSE
```

TO CANCEL: **ESC** p + Ø

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"p";CHR$(Ø);"back
to normal"
```

Microsoft BASIC:
```
LPRINT CHR$(27);"p";CHR$(Ø);"back to
normal"
```

BBC BASIC:
```
VDU2,1,27,1,ASC("p"),1,Ø: PRINT "back to
normal": VDU3
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"p";CHR$(Ø);
"back  to normal": CLOSE
```

Condensed typeface

TO SELECT: **SI** or **ESC SI**

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(15);"this is condensed"
```
Microsoft BASIC:
```
LPRINT CHR$(15);"this is condensed"
```
BBC BASIC:
```
VDU2,1,15: PRINT "this is condensed": VDU3
```
Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(15);"this is
condensed": CLOSE
```

TO CANCEL: **DC2**

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(18);"back to normal"
```
Microsoft BASIC:
```
LPRINT CHR$(18);"back to normal"
```
BBC BASIC:
```
VDU2,1,18:PRINT "back to normal": VDU3
```
Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(18);"back to
normal": CLOSE
```

NLQ-standard typeface

TO SELECT: **ESC** x + 1

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"x";CHR$(1);"this is
NLQ-standard"
```
Microsoft BASIC:
```
LPRINT CHR$(27);"x";CHR$(1);"this is
NLQ-standard"
```
BBC BASIC:
```
VDU2,1,27,1,ASC("x"),1,1: PRINT "this
is NLQ-standard": VDU3
```
Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"x";CHR$(1);
"t his is NLQ-standard": CLOSE
```

NOTE - The NLQ-standard typeface may be manually selected by holding down the **LF** and **ON LINE** buttons while switching the printer on.

TO CANCEL: **ESC** x + Ø

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"x";CHR$(Ø);"back
to normal"
```

Microsoft BASIC:
```
LPRINT CHR$(27);"x";CHR$(Ø);"back to
normal"
```

BBC BASIC:
```
VDU2,1,27,1,ASC("x"),1,Ø: PRINT "back to
normal": VDU3
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"x";CHR$(Ø);
"back to normal": CLOSE
```

NLQ-proportional typeface

TO SELECT: **ESC** x + 1 + **ESC** p + 1

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"x";CHR$(1);CHR$(27 );
"p";CHR$(1);"this is NLQ-proportional"
```

Microsoft BASIC:
```
LPRINT CHR$(27);"x";CHR$(1);CHR$(27);
"p";CHR$(1);"this is NLQ-proportional"
```

BBC BASIC:
```
VDU2,1,27,1,ASC("x"),1,1,1,27,1,ASC("p"),
1,1: PRINT "this is NLQ-proportional":
VDU3
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"x";CHR $(1);
CHR$(27);"p";CHR$(1) ;"this is NLQ-
proportional": CLOSE
```

TO CANCEL: **ESC** x + Ø + **ESC** p + Ø

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"x";CHR$(Ø);CHR$(27);
"p";CHR$(Ø);"back to normal"
```

Microsoft BASIC:
```
LPRINT CHR$(27);"x";CHR$(Ø);CHR$(27);"p";
CHR$(Ø);"back to normal"
```

BBC BASIC:
```
VDU2,1,27,1,ASC("x"),1,Ø,1,27,1,ASC("p"),1,
Ø: PRINT "back to normal": VDU3
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"x";CHR$(Ø);
CHR$(27);"p";CHR$(Ø);"back to normal":
CLOSE
```

Selecting additional functions

Having selected your main typeface, you may now modify it using the additional options available.

Subscript option

TO SELECT: **ESC** S + 1

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"S";CHR$(1);"this is
a subscript"
```

Microsoft BASIC:
```
LPRINT CHR$(27);"S";CHR$(1);"this is a
subscript"
```

BBC BASIC:
```
VDU2,1,27,1,ASC("S"),1,1: PRINT "this is
a subscript": VDU3
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"S";CHR$(1);
"this is a subscript": CLOSE
```

TO CANCEL: **ESC** T

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"T";"back to normal"
```

Microsoft BASIC:
```
LPRINT CHR$(27);"T";"back to normal"
```

BBC BASIC:
```
VDU2,1,27,1,ASC("T"): PRINT "back to
normal": VDU3
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"T";"back to
normal": CLOSE
```

Superscript option

TO SELECT: **ESC** S + Ø

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"S";CHR$(Ø);"this is
a superscript"
```

Microsoft BASIC: `LPRINT CHR$(27);"S";CHR$(0);"this is a superscript"`

BBC BASIC: `VDU2,1,27,1,ASC("S"),1,0: PRINT "this is superscript": VDU3`

Commodore BASIC: `OPEN 4,4: PRINT #4,CHR$(27);"S";CHR$(0);"this is a superscript": CLOSE`

TO CANCEL: **ESC** T

Examples:

AMSTRAD BASIC: `PRINT #8,CHR$(27);"T";"back to normal"`

Microsoft BASIC: `LPRINT CHR$(27);"T";"back to normal"`

BBC BASIC: `VDU2,1,27,1,ASC("T"): PRINT "back to normal": VDU3`

Commodore BASIC: `OPEN 4,4: PRINT #4,CHR$(27);"T";"back to normal": CLOSE`

Double strike option

TO SELECT: **ESC** G

Examples:

AMSTRAD BASIC: `PRINT #8,CHR$(27);"G";"this is double-strike"`

Microsoft BASIC: `LPRINT CHR$(27);"G";"this is double-strike"`

BBC BASIC: `VDU2,1,27,1,ASC("G"): PRINT "this is double-strike": VDU3`

Commodore BASIC: `OPEN 4,4: PRINT #4,CHR$(27);"G";"this is double-strike": CLOSE`

TO CANCEL: **ESC** H

Examples:

AMSTRAD BASIC: `PRINT #8,CHR$(27);"H";"back to normal"`

Microsoft BASIC: `LPRINT CHR$(27);"H";"back to normal"`

BBC BASIC:
```
VDU2,1,27,1,ASC("H"): PRINT "back to
normal": VDU3
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"H";"back to
normal": CLOSE
```

Italics option

TO SELECT: **ESC** 4

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"4";"this is italicised"
```

Microsoft BASIC:
```
LPRINT CHR$(27);"4";"this is italicised"
```

BBC BASIC:
```
VDU2,1,27,1,ASC("4"): PRINT "this is
italicised": VDU3
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"4";"this is
italicised": CLOSE
```

TO CANCEL: **ESC** 5

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"5";"back to normal"
```

Microsoft BASIC:
```
LPRINT CHR$(27);"5";"back to normal"
```

BBC BASIC:
```
VDU2,1,27,1,ASC("5"): PRINT "back to
normal": VDU3
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"5";"back
to normal": CLOSE
```

Bold option

TO SELECT: **ESC** E

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"E";"this is bold"
```

Microsoft BASIC:
```
LPRINT CHR$(27);"E";"this is bold"
```

BBC BASIC: `VDU2,1,27,1,ASC("E"): PRINT "this is bold": VDU3`

Commodore BASIC: `OPEN 4,4: PRINT #4,CHR$(27);"E";"this is bold": CLOSE`

TO CANCEL: **ESC** F

Examples:

AMSTRAD BASIC: `PRINT #8,CHR$(27);"F";"back to normal"`

Microsoft BASIC: `LPRINT CHR$(27);"F";"back to normal"`

BBC BASIC: `VDU2,1,27,1,ASC("F"): PRINT "back to normal": VDU3`

Commodore BASIC: `OPEN 4,4: PRINT #4,CHR$(27);"F";"back to normal": CLOSE`

Selecting underline or double-width printing

Both underline and/or double-width printing may be added to any of the available combinations of typeface.

Underline option

TO SELECT: **ESC** - + 1

Examples:

AMSTRAD BASIC: `PRINT #8,CHR$(27);"-";CHR$(1);"this is underlined"`

Microsoft BASIC: `LPRINT CHR$(27);"-";CHR$(1);"this is underlined"`

BBC BASIC: `VDU2,1,27,1,ASC("-"),1,1: PRINT "this is underlined": VDU3`

Commodore BASIC: `OPEN 4,4: PRINT #4,CHR$(27);"-";CHR$(1);"this is underlined": CLOSE`

TO CANCEL: **ESC - + Ø**

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"-";CHR$(Ø);"back
to normal"
```

Microsoft BASIC:
```
LPRINT CHR$(27);"-";CHR$(Ø);"back to
normal"
```

BBC BASIC:
```
VDU2,1,27,1,ASC("-"),1,Ø: PRINT "back to
normal": VDU3
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"-";CHR$(Ø);
"back to normal": CLOSE
```

Double-width option

TO SELECT: **ESC W** + 1 or **SO**

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"W";CHR$(1);"this is
double-width"
```

Microsoft BASIC:
```
LPRINT CHR$(27);"W";CHR$(1);"this is
double-width"
```

BBC BASIC:
```
VDU2,1,27,1,ASC("W"),1,1: PRINT "this is
double-width": VDU3
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"W";CHR$(1);
"this is double-width": CLOSE
```

TO CANCEL: **ESC W** + **Ø** or **DC4**

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"W";CHR$(Ø);"back
to normal"
```

Microsoft BASIC:
```
LPRINT CHR$(27);"W";CHR$(Ø);"back
to normal"
```

BBC BASIC:
```
VDU2,1,27,1,ASC("W"),1,Ø: PRINT "back to
normal": VDU3
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"W";CHR$(Ø);"
back to normal": CLOSE
```

Combining styles

Now that you know how to access each of the different typefaces, you might like to combine a few of them. Note that this time, in the examples that follow, I will not differentiate between the various dialects of BASIC, but will simply give the essential body of the `PRINT` statement.

Make sure that your printer is set to Standard typeface with no additional style options selected - (if in doubt, switch the printer off, then on again).

Example `PRINT` statement -
(Mini typeface incorporating italics and underline):

```
CHR$(27);"M";"You can emphasise a point using ";CHR$(27);
"4";"italics ";CHR$(27);"5";"or ";CHR$(27);"-";CHR$(1);
"underline";CHR$(27);"-";CHR$(Ø);CHR$(27);"P"
```

Note that each of the typefaces and options selected in this example are cancelled after use. If you don't cancel them, they will be used in the next print statement.

Studying this example, you could be forgiven for thinking that it looks muddled, and that it is difficult to pick out your typeface selections.

A solution to this problem is to assign the escape codes into string variables as follows (don't type this in just yet)....

```
mini.on$=CHR$(27)+"M": mini.off$=CHR$(27)+"P"
ital.on$=CHR$(27)+"4": ital.off$=CHR$(27)+"5"
und.on$=CHR$(27)+"-"+CHR$(1): und.off$=CHR$(27)+"-"+CHR$(Ø)
```

....and thereafter use the string variables' names to select or cancel a particular typeface.

(Note that when assigning a mixture of strings to string variables as above, the strings must be appended to each other using the + operator.)

The variables' names used above are a bit 'long-winded' in order to illustrate their functions. However, if you use short names such as....

`ms$` (mini select)
`mc$` (mini cancel)
`is$` (italics select)
`ic$` (italics cancel)
`us$` (underline select)
`uc$` (underline cancel)

....you can see that the example text would become much clearer. In addition you could use the variables in all subsequent print statements. It's a good idea to have a ready-written program or routine like this to call upon when you're printing in various styles - and no doubt you'll incorporate something along these lines into your own text creation utility.

So let's create the beginnings of such a program. We'll add a few more typefaces and options.

Don't forget to line number this program if you wish to run it more than once or save it (Sinclair Spectrum users should type LET before every variable assignment).

```
ms$=CHR$(27)+"M": mc$=CHR$(27)+"P"
is$=CHR$(27)+"4": ic$=CHR$(27)+"5"
us$=CHR$(27)+"-"+CHR$(1): uc$=CHR$(27)+"-"+CHR$(0)
ws$=CHR$(27)+"W"+CHR$(1): wc$=CHR$(27)+"W"+CHR$(0)
cs$=CHR$(15): cc$=CHR$(18)
```

Now try this PRINT statement:

```
ms$;"You can emphasise a point using ";is$;"italics ";ic$;
"or ";us$;"underline. ";uc$;"You can also spread out a bit
using ";ws$;"double width characters, ";wc$;mc$;cs$;"or
even hide something in the small print!";cc$
```

Notice that before the last phrase is printed (in Condensed typeface) the Mini typeface is first cancelled (by mc$). This is because Mini typeface and Condensed typeface cannot be used together - it is an 'illegal combination'. The final section in this chapter (see ahead) illustrates which typeface combinations are permitted and which are illegal.

Subscripts and superscripts

Add the following to your list of escape code variables:

```
nss$=CHR$(27)+"x"+CHR$(1): nsc$=CHR$(27)+"x"+CHR$(0)
sbs$=CHR$(27)+"S"+CHR$(1): sc$=CHR$(27)+"T"
sps$=CHR$(27)+"S"+CHR$(0)
```

(Note that the string variable sc$ cancels both the Subscript and the Superscript options.)

Example PRINT statement -
(NLQ-standard typeface incorporating subscripts and superscripts):

```
nss$;"Subscripts include H";sbs$;"2";sc$;"Ø and Log";sbs$;
"1Ø";sc$;", while superscripts include 1Ø";sps$;"-3";sc$;
" and 1ØØ";sps$;"o";sc$;"C.";nsc$
```

A few more assorted typefaces:

```
ps$=CHR$(27)+"p"+CHR$(1): pc$=CHR$(27)+"p"+CHR$(Ø)
ds$=CHR$(27)+"G": dc$=CHR$(27)+"H"
bs$=CHR$(27)+"E": bc$=CHR$(27)+"F"
```

Example PRINT statements -
(proportional, double strike, and bold typefaces):

```
ps$;"Proportional printing has non-uniform character widths
";pc$

"Trade unionists will favour the ";ds$;"double strike ";dc$
;"option"

"Your mission - ";bs$;"to boldly print like no other printe
r";bc$
```

Illegal combinations - what you can and can't do

Not all typefaces can be combined with all additional options. You cannot, for example, choose NLQ-proportional with bold italic subscripts. The table on the next page illustrates the permitted typeface combinations available, and more importantly -the illegal ones.

NOTES:

1. A blank square indicates an illegal combination.
2. All options may include Double-width and/or Underlining.
3. When using Standard typeface, you may select both the Bold and Italic options together.

	NORMAL (OFF)	DOUBLE STRIKE	SUB-SCRIPT	SUPER-SCRIPT	
STANDARD TYPEFACE	OK	OK	OK	OK	NORMAL (OFF)
	OK	OK	OK	OK	BOLD
	OK	OK	OK	OK	ITALICS
MINI TYPEFACE	OK	OK	OK	OK	NORMAL (OFF)
					BOLD
	OK	OK	OK	OK	ITALICS
PROPORTIONAL TYPEFACE	OK	OK			NORMAL (OFF)
					BOLD
	OK	OK			ITALICS
CONDENSED TYPEFACE	OK	OK	OK	OK	NORMAL (OFF)
					BOLD
	OK	OK	OK	OK	ITALICS
NLQ-STANDARD TYPEFACE	OK		OK	OK	NORMAL (OFF)
					BOLD
					ITALICS
NLQ-PROPORTIONAL TYPEFACE	OK				NORMAL (OFF)
					BOLD
					ITALICS
	NORMAL (OFF)	DOUBLE STRIKE	SUB-SCRIPT	SUPER-SCRIPT	

Chapter 4
Print formatting control....

Subjects covered in this chapter:

Print head movement
Form feed
Margin setting
Page length setting
Perforation skipping
Tabulation
Paper feed rate adjustment

Note that in many of the examples which follow, the carriage return following a control code is suppressed (note also that there are no `VDU3` (BBC BASIC) or `CLOSE` (Commodore BASIC) statements) - it is assumed that most of the codes will be followed by your own further instructions to the printer.

Carriage return

This code sends the print head back to the beginning of the line, ready to start printing at the left-hand margin.

TO SELECT: **CR**

Examples:

AMSTRAD BASIC: `PRINT #8,CHR$(13);`

Microsoft BASIC: `LPRINT CHR$(13);`

BBC BASIC: `VDU2,1,13`

Commodore BASIC: `OPEN 4,4: PRINT #4,CHR$(13);`

Line feed

This code feeds the paper up by one line so that the print head is ready to start printing on the next line. Line feed has the additional effect of outputting the contents of the buffer.

TO SELECT: **LF**

Examples:

AMSTRAD BASIC: `PRINT #8,CHR$(10);`

Microsoft BASIC: `LPRINT CHR$(10);`

BBC BASIC: `VDU2,1,10`

Commodore BASIC: `OPEN 4,4: PRINT #4,CHR$(10);`

Backspace

This code moves the print head one space to the left.

TO SELECT: **BS**

Examples:

AMSTRAD BASIC: `PRINT #8,CHR$(8);`

Microsoft BASIC: `LPRINT CHR$(8);`

BBC BASIC: `VDU2,1,8`

Commodore BASIC: `OPEN 4,4: PRINT #4,CHR$(8);`

Form feed

This code moves the print head to the start of the next page. It can be thought of as 'start a new page'.

TO SELECT: **FF**

Examples:

AMSTRAD BASIC: `PRINT #8,CHR$(12);`

Microsoft BASIC: `LPRINT CHR$(12);`

BBC BASIC: `VDU2,1,12`

Commodore BASIC: `OPEN 4,4: PRINT #4,CHR$(12);`

Margins

The width of the page can be set by altering the margins outside of which the printer will not print.

Right margin setting

This code sets the right hand margin to the value of n (in the range 1 to 255). The value n represents the number of character columns from the left hand edge of the printer.

TO SELECT: **ESC** Q + n

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"Q";CHR$(50)
```

Microsoft BASIC:
```
LPRINT CHR$(27);"Q";CHR$(50)
```

BBC BASIC:
```
VDU2,1,27,1,ASC("Q"),1,50
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"Q";CHR$(50)
```

Left margin setting

This code sets the left hand margin to the value of n (in the range 0 to 255). The value n represents the number of character columns from the left hand edge of the printer.

TO SELECT: **ESC** l + n

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"l";CHR$(20)
```

Microsoft BASIC:
```
LPRINT CHR$(27);"l";CHR$(20)
```

BBC BASIC:
```
VDU2,1,27,1,ASC("l"),1,20
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"l";CHR$(20)
```

Note that if the right hand margin is set to a value which is less than (or equal to) the left hand margin, then the right hand margin setting will be ignored.

Page length setting (by lines)

This code sets the page length to the value of n (in the range 1 to 127). The value n represents the number of lines per page. Note that changing the paper feed rate (described ahead) will not alter the page dimensions set by this command.

TO SELECT: **ESC** C + n

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"C";CHR$(15)
```

Microsoft BASIC:
```
LPRINT CHR$(27);"C";CHR$(15)
```

BBC BASIC:
```
VDU2,1,27,1,ASC("C"),1,15
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"C";CHR$(15)
```

Page length setting (by inches)

This code sets the page length to the value of n (in the range 1 to 22). The value n represents the number of inches per page.

TO SELECT: **ESC** C + Ø + n

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"C";CHR$(Ø);CHR$(4)
```

Microsoft BASIC:
```
LPRINT CHR$(27);"C";CHR$(Ø);CHR$(4)
```

BBC BASIC:
```
VDU2,1,27,1,ASC("C"),1,Ø,1,4
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"C";
CHR$(Ø);CHR$(4)
```

Skip perforation setting

When using continuous stationery (such as fan fold/tractor feed paper) the printer can be set to skip a number of lines when it reaches the foot of a page in order to avoid printing directly over the perforations in the paper.

This code sets skip perforation to the value of n (in the range 1 to 127). The value n represents the number of lines to be skipped at the foot of a page. This value is deducted from the page length setting (in lines or inches) if previously set.

TO SELECT: **ESC N** + n

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"N";CHR$(5)
```

Microsoft BASIC:
```
LPRINT CHR$(27);"N";CHR$(5)
```

BBC BASIC:
```
VDU2,1,27,1,ASC("N"),1,5
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"N";CHR$(5)
```

TO CANCEL: **ESC** 0 (Note that this is the capital letter 0)

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"O"
```

Microsoft BASIC:
```
LPRINT CHR$(27);"O"
```

BBC BASIC:
```
VDU2,1,27,1,ASC("O")
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"O"
```

Tabulation

It is possible to set up variable positions to which the print head can be moved. These are called tabulation settings or 'tabs' for short. Tabs can be set for vertical and horizontal positions. Once the tab positions have been set, the print head is sent to that position by the appropriate tab jump command.

The DMP2000 has an additional feature allowing vertical tabs to be set in different 'channels'. A channel can then be selected and the tab settings for that particular channel used.

Horizontal tab setting

This code sets the horizontal tab positions. Up to 32 tab positions may be specified (each in the range 1 to 137).

TO SELECT: **ESC** D + n1 + n2 + n3 ....etc.... n32 + Ø

The value of n2 should be greater than the value of n1, n3 should be greater than n2, n4 should be greater than n3, and so on. The sequence of tab numbers must be terminated by a zero.

(When the printer is first switched on, horizontal tabs default to every 8 character positions.)

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"D";CHR$(10);CHR$(20);
CHR$(0)
```

Microsoft BASIC:
```
LPRINT CHR$(27);"D";CHR$(10);CHR$(20);
CHR$(0)
```

BBC BASIC:
```
VDU2,1,27,1,ASC("D"),1,10,1,20,1,0
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"D";CHR$(10);
CHR$(20);CHR$(0)
```

Horizontal tab jump

This code sends the print head to the next horizontal tab position.

TO SELECT: **TAB**

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(9);
```

Microsoft BASIC:
```
LPRINT CHR$(9);
```

BBC BASIC:
```
VDU2,1,9
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(9);
```

Vertical tab setting

This code sets the vertical tab positions. Up to 16 tab positions may be specified (each in the range 1 to 254).

TO SELECT: **ESC** B + n1 + n2 + n3 ....etc.... n16 + Ø

The value of n2 should be greater than the value of n1, n3 should be greater than n2, n4 should be greater than n3, and so on. The sequence of tab numbers must be terminated by a zero.

(When the printer is first switched on, vertical tabs default to single line feeds.)

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"B";CHR$(10);CHR$(20);
CHR$(0)
```

Microsoft BASIC:
```
LPRINT CHR$(27);"B";CHR$(10);CHR$(20);
CHR$(0)
```

BBC BASIC:
```
VDU2,1,27,1,ASC("B"),1,10,1,20,1,0
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"B";CHR$(10);
CHR$(20);CHR$(0)
```

Vertical tab jump

This code sends the print head to the next vertical tab position.

TO SELECT: **VT**

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(11);
```

Microsoft BASIC:
```
LPRINT CHR$(11);
```

BBC BASIC:
```
VDU2,1,11
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(11);
```

Tab channel setting

There are eight channels for which it is possible to set up to 16 vertical tab positions each. This feature is intended for applications where more than one type of page format is being used within a document. To use this feature, you should first set up the channel and tab settings (using **ESC** b + channel + n1 ....etc.... + **Ø**) for each of the required channels. You may thereafter select the appropriate channel (using **ESC** / + channel) when you wish to call up a new set of tabs, and use the vertical tab jump (**VT**) code to move to the next tab position.

NOTE - If no channel tabs are set up or selected, channel 0 is assumed.

TO SELECT: **ESC** b + channel + n1 + n2 + n3 ...etc... n16 + Ø

The channel parameter must be in the range 0 to 7.

The value of n2 should be greater than the value of n1, n3 should be greater than n2, n4 should be greater than n3, and so on. The sequence of tab numbers must be terminated by a zero.

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"b";CHR$(7);CHR$(10);
CHR$(20);CHR$(0)
```

Microsoft BASIC:
```
LPRINT CHR$(27);"b";CHR$(7);CHR$(10);
CHR$(20);CHR$(0)
```

BBC BASIC:
```
VDU2,1,27,1,ASC("b"),1,7,1,10,1,20,1,0
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"b";CHR$(7);
CHR$(10);CHR$(20);CHR$(0)
```

Channel selection

This code selects the tab channel to be used (in the range 0 to 7).

TO SELECT: **ESC /** + channel

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"/";CHR$(7)
```

Microsoft BASIC:
```
LPRINT CHR$(27);"/";CHR$(7)
```

BBC BASIC:
```
VDU2,1,27,1,ASC("/"),1,7
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"/";CHR$(7)
```

All subsequent vertical tab jump (**VT**) codes will move the print head to the next vertical tab position for that particular channel.

Paper feed rates

When the printer is switched on, the paper feed rate defaults to $^1/_6$ inch per line. However, the amount of paper fed per line can be altered using the following commands.

$^1/_8$ inch paper feed

TO SELECT: **ESC Ø**

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"Ø"
```

Microsoft BASIC:
```
LPRINT CHR$(27);"Ø"
```

BBC BASIC:
```
VDU2,1,27,1,ASC("Ø")
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"Ø"
```

$^7/_{72}$ inch paper feed

TO SELECT: **ESC** 1

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"1"
```

Microsoft BASIC:
```
LPRINT CHR$(27);"1"
```

BBC BASIC:
```
VDU2,1,27,1,ASC("1")
```

Commodore BASIC:
```
OPEN 4,4: PRINT #4,CHR$(27);"1"
```

$^1/_6$ inch paper feed (default)

TO SELECT: **ESC** 2

Examples:

AMSTRAD BASIC:
```
PRINT #8,CHR$(27);"2"
```

Microsoft BASIC:
```
LPRINT CHR$(27);"2"
```

BBC BASIC: `VDU2,1,27,1,ASC("2")`

Commodore BASIC: `OPEN 4,4: PRINT #4,CHR$(27);"2"`

Variable $^{n}/_{216}$ inch paper feed

This code sets the paper feed rate to $^{n}/_{216}$ inch. The value of n may be in the range 0 to 255.

TO SELECT: **ESC** 3 + n

Examples:

AMSTRAD BASIC: `PRINT #8,CHR$(27);"3";CHR$(27)`

Microsoft BASIC: `LPRINT CHR$(27);"3";CHR$(27)`

BBC BASIC: `VDU2,1,27,1,ASC("3"),1,27`

Commodore BASIC: `OPEN 4,4: PRINT #4,CHR$(27);"3";CHR$(27)`

Variable $^{n}/_{72}$ inch paper feed

This code sets the paper feed rate to $^{n}/_{72}$ inch. The value of n may be in the range 0 to 85.

TO SELECT: **ESC** A + n

Examples:

AMSTRAD BASIC: `PRINT #8,CHR$(27);"A";CHR$(18)`

Microsoft BASIC: `LPRINT CHR$(27);"A";CHR$(18)`

BBC BASIC: `VDU2,1,27,1,ASC("A"),1,18`

Commodore BASIC: `OPEN 4,4: PRINT #4,CHR$(27);"A";CHR$(18)`

Variable $^{n}/_{216}$ inch one-shot forward feed

This code executes a once-only forward paper feed of $^{n}/_{216}$ inch. The value of n may be in the range 0 to 255.

TO SELECT: **ESC** J + n

Examples:

```
AMSTRAD BASIC:   PRINT #8,CHR$(27);"J";CHR$(216)

Microsoft BASIC:   LPRINT CHR$(27);"J";CHR$(216)

BBC BASIC:   VDU2,1,27,1,ASC("J"),1,216

Commodore BASIC:   OPEN 4,4: PRINT #4,CHR$(27);"J";CHR$(216)
```

Variable $^{n}/_{216}$ inch one-shot reverse feed

This code executes a once-only reverse paper feed of $^{n}/_{216}$ inch. The value of n may be in the range 0 to 255.

TO SELECT: **ESC** j + n

Examples:

```
AMSTRAD BASIC:   PRINT #8,CHR$(27);"j";CHR$(108)

Microsoft BASIC:   LPRINT CHR$(27);"j";CHR$(108)

BBC BASIC:   VDU2,1,27,1,ASC("j"),1,108

Commodore BASIC:   OPEN 4,4: PRINT #4,CHR$(27);"j";CHR$(108)
```

Chapter 5
Graphics printing....

Subjects covered in this chapter:

Introduction
Single, double, and quadruple density graphics
Bit image graphics modes

What is graphics printing?

The printer can interpret each code that it receives in different ways, depending on the mode it happens to be in. When set into one of the graphics modes, instead of a complete character being printed, the code received by the printer is used to directly control each of the tiny individual 'pins' inside the print head. For each code sent, only one line of dots is drawn vertically. Whether a dot is drawn or not depends on the byte of data sent. Each bit in the byte can be a 1 or a 0; a 1 indicates that a dot should be printed, a 0 indicates that a space should be left. This is known as 'bit image graphics'.

(NOTE TO AMSTRAD COMPUTER USERS - If you're not too sure about the terms 'byte' and 'bit', read the section entitled 'Bits and bytes' towards the end of your computer manual.)

As usual, the graphics mode is entered using an escape code. Two further parameters are also included and these tell the printer how much data it should turn into graphics images before reverting to character printing once again. The parameters are in the form of a two-byte integer (n1 and n2) with the low byte first. Using a seven bit printer port (as available on AMSTRAD computers) it will only be possible to specify a value between 0 and 127 for either of these parameters -(values 128 to 255 of the byte cannot be specified). This should not cause problems however, as the graphics mode can be set to repeat during a line. This will be become clearer as the command is used.

It should be noted that while in the graphics mode, all control codes are printed as bit image graphics (rather than being executed). This means that the effect of sending, for example, a carriage return or line feed will NOT be to execute the appropriate function, but to actually print out a series of dots on the paper.

There is a maximum number of dots that can be printed on one line. If the maximum number is exceeded, then the additional data is ignored. The graphics mode does not automatically execute a line feed and carriage return when it reaches the end of a line.

The following short example (AMSTRAD BASIC only) shows how to print some graphics. If you are not using an AMSTRAD computer, adapt the program to suit your own particular computer:

```
10 WIDTH 255
20 PRINT #8,CHR$(27);"K";CHR$(126);CHR$(0);
30 FOR n=1 TO 126
40   PRINT #8,CHR$(&X1010111);
50 NEXT
```

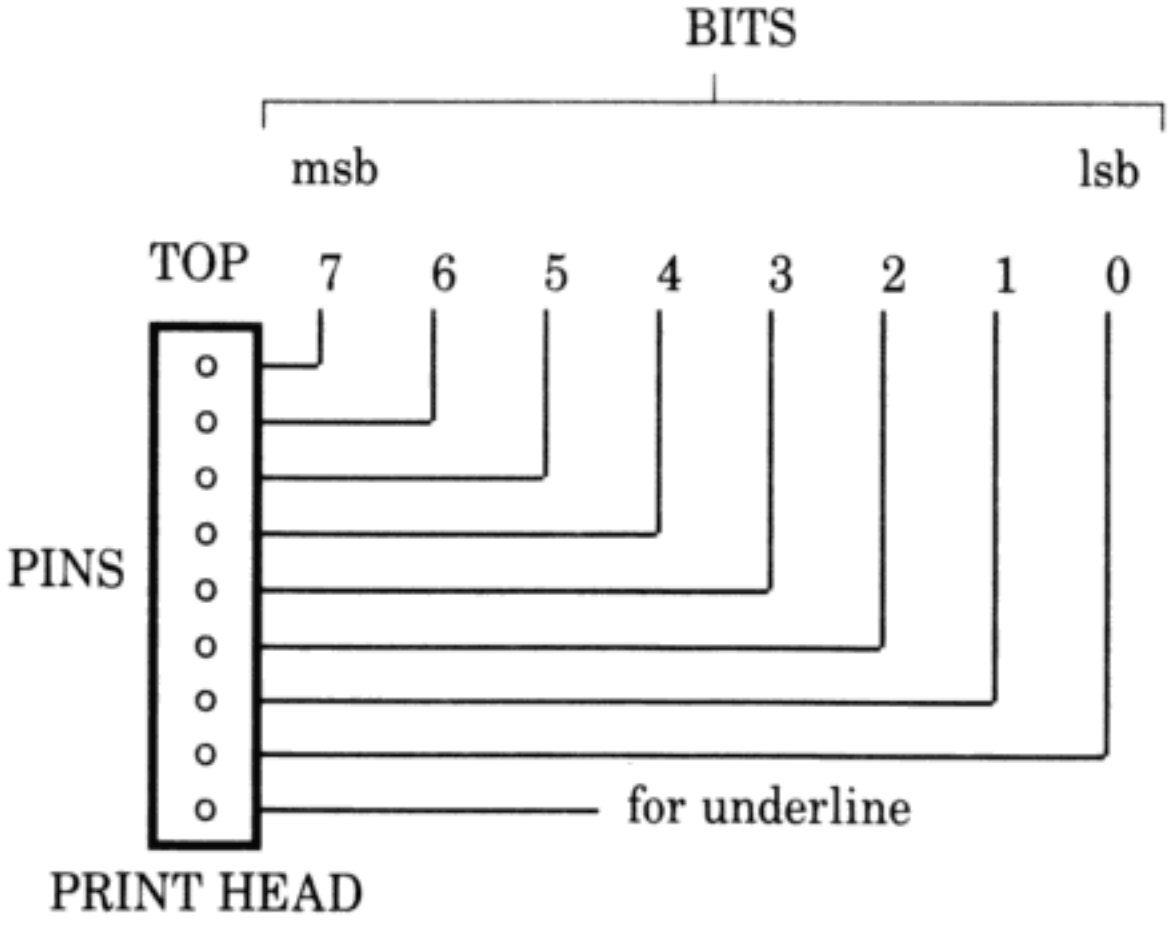
There are a few points that should be noted about this program:

Firstly, a `WIDTH 255` command has been included. This suppresses the automatic generation of carriage return/line feed after a certain number of characters are printed out. If carriage return/line feed were not suppressed, spurious dot patterns in the printed output graphics would result.

Secondly, the use of a semicolon to terminate the print statement (in line `20`) which selects graphics mode is essential, as this suppresses the carriage return/line feed normally executed after a print statement, which again would otherwise result in spurious dot patterns.

We have used &X1010111 as our data in the above example. You could have typed 87 (decimal) instead of &X1010111 (which is the binary representation of 87), but in binary form you can see the effect of each 1 and 0 on the result. Try using different numbers within the above example to understand the effect.

The following diagram illustrates how each bit of data relates to a corresponding pin on the print head, which in turn produces a corresponding dot on the paper.



Graphics modes

There is more than one graphics mode. The others allow denser graphics and faster speeds. These are summarised below.

NOTE - For each of the following commands, the parameters n1 and n2 are in the range 0 to 255 for 8 bit printer ports, and in the range 0 to 127 for 7 bit printer ports as used on AMSTRAD computers.

Single density graphics

(Maximum printable positions on a line - 480)

TO SELECT: **ESC** K + n1 + n2

Double density graphics

(Maximum printable positions on a line - 960)

TO SELECT: **ESC** L + n1 + n2

Double speed double density graphics

(Maximum printable positions on a line - 960)

TO SELECT: **ESC** Y + n1 + n2

Quadruple density graphics

(Maximum printable positions on a line - 1920)

TO SELECT: **ESC** Z + n1 + n2

Bit image mode

TO SELECT: **ESC *** + mode + n1 + n2

....where the mode parameter is the required graphics mode (see the following table):

mode	Number of dots/8 inch	Connecting dot density/8 inch
0	480 single density	480
1	960 double density	960
2	960 double speed/double density	960
3	1920 quadruple density	1920
4	640 CRT graphic	640
5	576 plotter graphic	576
6	720 CRT graphic	720

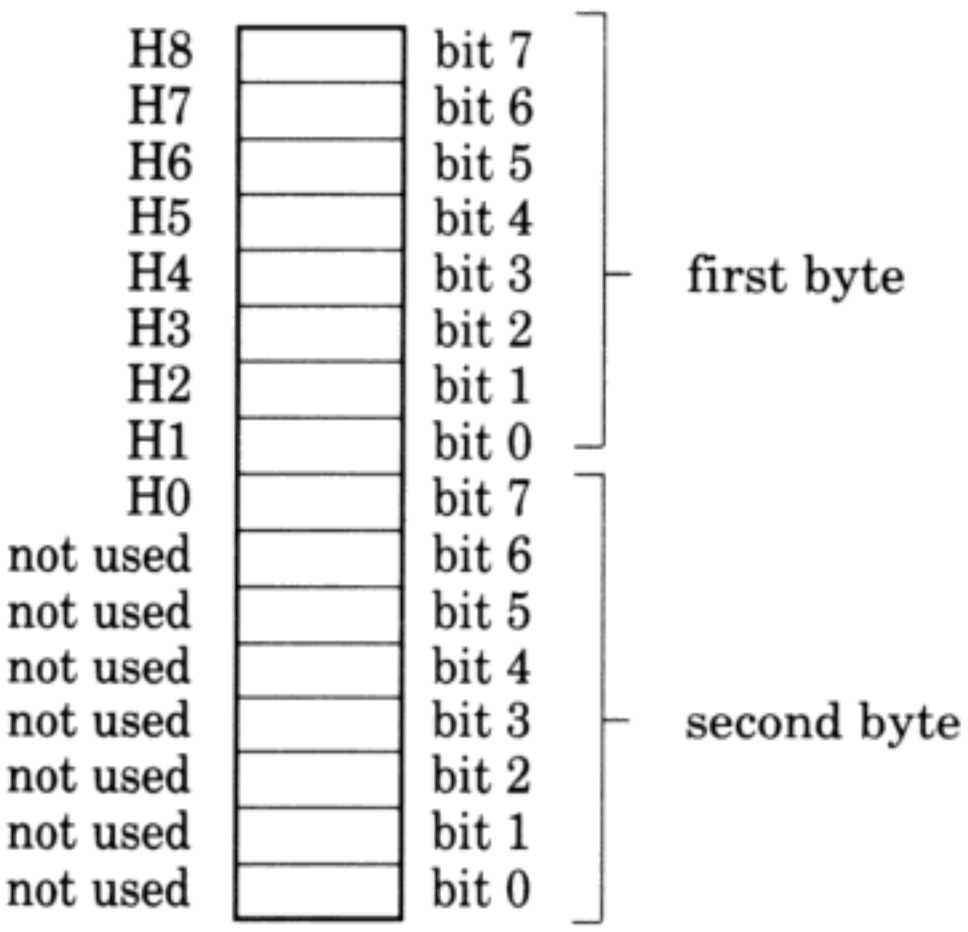
9-pin bit image mode (8 bit printer output only)

TO SELECT: **ESC** ↑ + mode + n1 + n2

....where the mode parameter is the required graphics mode (see the following table):

mode	Maximum number of dots	Density
0	480	single density
1	960	double density

Two bytes of data should be sent for each printable position. The first is used as data for the top eight pins. Bit 7 of the second byte is used as data for the bottom pin:



Bit image mode selection/change

TO SELECT: **ESC ?** + code + mode

....where the code parameter is one of the control code letters K, L, Y, or Z (described earlier in this chapter), and where the mode parameter is in the range 0 to 6 (as previously described for the **ESC *** code).

Chapter 6
Extra functions....

Subjects covered in this chapter:

Incremental print
8 bit characters
Control code printing
Reset, paper-out, and bleeper
Deleting characters from the buffer
Print head control
International character sets
User defined characters
Hexadecimal dump

Incremental print

TO SELECT: **ESC** `i + 1`

This code allows characters sent to the printer to be printed out immediately. After printing, the paper is fed forward to allow the character(s) to be seen. The paper returns to the previous print position when it receives the next character.

TO CANCEL: **ESC** `i + 0`

Printable code area expansion (8 bit printer output only)

This code allows the printing out of the eighth bit characters 128 to 159 and 255 (&80 to &9F and &FF HEX) which are by default set to produce 'un-printable' control characters.

TO SELECT: **ESC** 6

If you have connected the printer to a computer with an 8 bit printer port and you wish to view the available characters, simply select the above control code, then run the following program (adapt the `PRINT` command to suit your particular computer):

```
10 FOR n=128 TO 159
20 PRINT (to the printer) CHR$(n);n
30 NEXT n
40 PRINT (to the printer) CHR$(255);255
```

TO CANCEL: **ESC** 7

Set eighth bit

This code sets the eighth bit of every code sent to the printer to 1.

TO SELECT: **ESC** >

Unset eighth bit

This code sets the eighth bit of every code sent to the printer to 0.

TO SELECT: **ESC** =

Accept eighth bit

This code cancels the above (set and unset) commands, directing the printer to accept all eight bits as they are received from the computer.

TO SELECT: **ESC** #

Control code printing

By default, control codes are executed when sent to the printer. This code allows instead, the printing out of any following control codes (between 0 and 31).

Note that not all codes in this range can be set to produce printable characters.

TO SELECT: **ESC** I + 1

TO CANCEL: **ESC** I + 0

Reset printer

This code resets the printer. The printer will then perform as if it had just been switched on.

TO SELECT: **ESC** @

Paper-out detect disable

This code disables the action of the PAPER OUT sensor which halts the printer when there is no paper.

TO SELECT: **ESC** 8

Paper-out detect enable

This code re-enables the action of the PAPER OUT sensor.

TO SELECT: **ESC** 9

Bleeper

This code sounds the printer's internal bleeper.

TO SELECT: **BEL** (send `CHR$(7)` to the printer)

Delete

This code deletes the last character from the printer's buffer.

TO SELECT: **DEL** (send `CHR$(127)` to the printer)

Note that the above code will not operate in the incremental print mode, nor will it delete any control codes in the buffer.

Cancel

This code deletes the entire contents of the printer's buffer.

TO SELECT: **CAN** (send `CHR$(24)` to the printer)

Note that the above code will not operate in the incremental print mode, nor will it delete any control codes in the buffer.

Home head

This code moves the print head to the home (leftmost) position.

TO SELECT: **ESC** <

Uni-directional printing

By default, the DMP2000 prints in both directions - left to right and right to left. This code selects printing from left to right only.

TO SELECT: **ESC** U + 1

TO CANCEL: **ESC** U + Ø

Half speed

This code halves the speed of the print head's movement.

TO SELECT: **ESC** s + 1

TO CANCEL: **ESC** s + Ø

International character set selection

The setting of the DIP switches (described in Chapter 2) facilitates 'hardware' selection of the international character set to be used when the printer is switched on. This code allows the DIP switch settings to be overriden by 'software'.

TO SELECT: **ESC** R + n

....where n is in the range 0 to 8 for the following countries:

n	COUNTRY
0	USA
1	France
2	Germany
3	UK
4	Denmark
5	Sweden
6	Italy
7	Spain
8	Japan

User defined characters

The printer has the ability to print user defined characters. The following section outlines how to to go about defining (downloading) a character.

There are four main stages to go through before a download character appears on the paper:

1. Download character definition
2. Download character set selection
3. Control code printing selection
4. Printing of the character

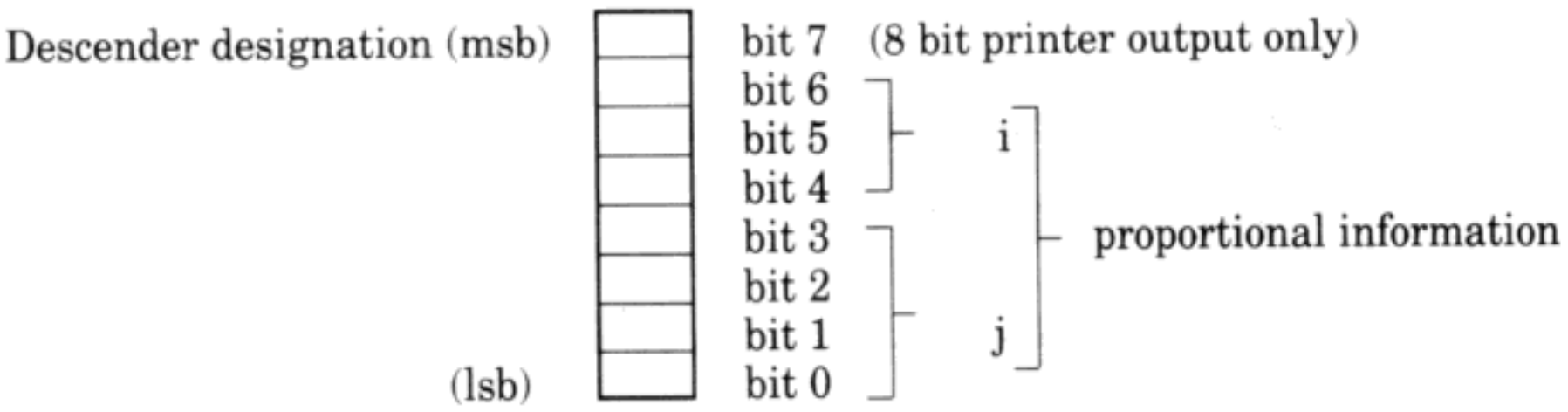
1. Download character definition

TO SELECT: **ESC &** + **Ø** + first + last +attribute + d1 + d2 ....etc.... + d11
+attribute + d1 + d2 ....etc.... + d11
+attribute + d1 + d2 ....etc.... + d11
....etc....

It is possible to define any of the characters in the range 0 to 31, though some of these cannot be displayed, as the control code takes precedence.

The 'first' and 'last' parameters specify the range of character numbers which are to be defined. First should be less than (or equal to) last. If first is less than last, then the data for the additional characters (attribute + d1 .... d11) should be added onto the statement.

The attribute parameter is a bit significant number corresponding as follows:



....where i is the starting position (in the range 0 to 7), and j is the finishing position (in the range i+4 to 11).

Bit 7 sets the descender designation - 0 to descend, 1 not to descend (usable only on a computer supplying 8 bit printer output).

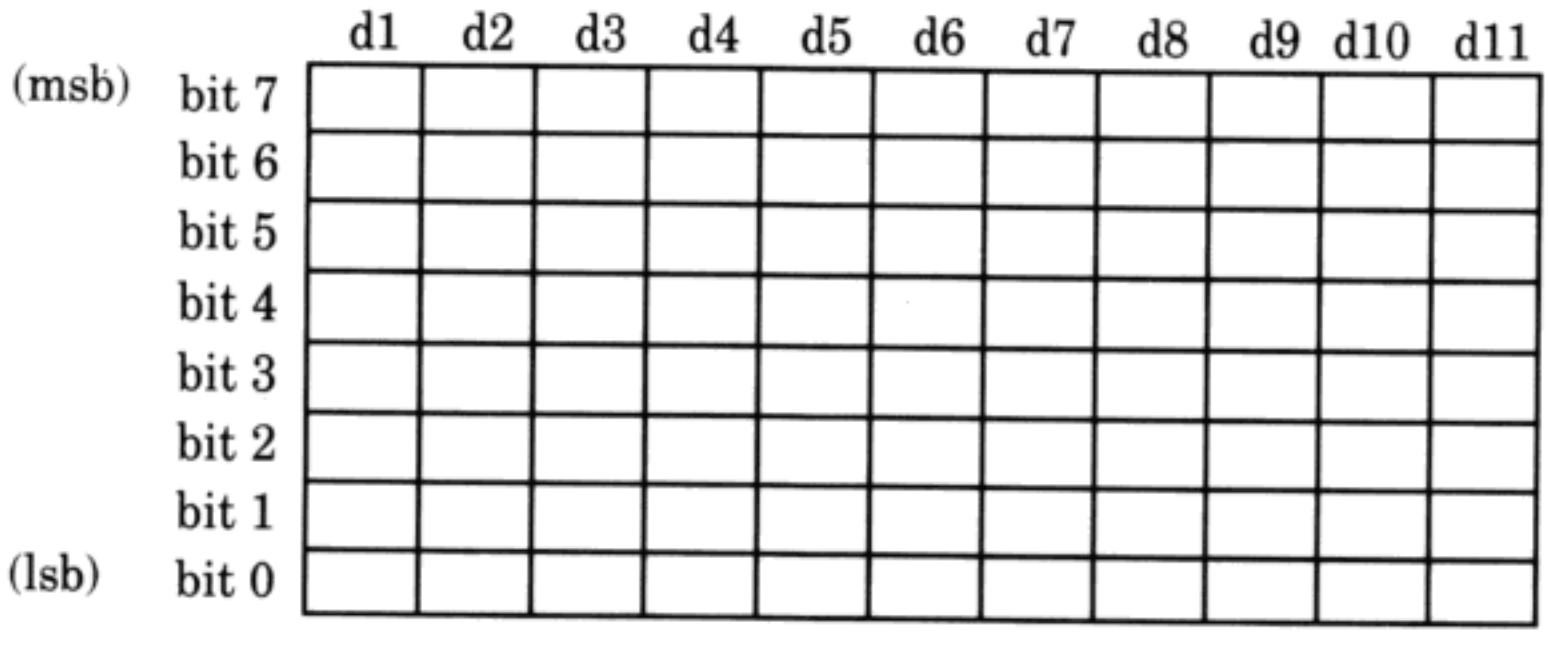
Bits 0 to 7 specify proportional information about the character.

The horizontal position of the character in the 11 column grid must be specified by giving start and finish positions.

Bits 4 to 7 contain the starting position which may be in the range 0 to 7. Bits 0 to 3 contain the finishing position which can be in the range (starting position + 4) to 11.

The minimum character size is 5.

The parameters d1 to d11 are the bit significant data for each vertical line of dots, corresponding as follows:



The character is defined from left to right with the highest bit printed at the top of the character. Please note that the printer is incapable of printing contiguous dots on the same horizontal line, and every other horizontally adjacent bit will be ignored if this is attempted. This should become clearer if you study the example at the end of this section.

2. Download character set selection

TO SELECT: **ESC** % + 1 + Ø

The above code selects the download character set for use.

TO CANCEL: **ESC** % + Ø + Ø

The above code de-selects the download character set and re-selects the internal character set.

(For your reference, the following code copies the internal character set into the download set:

TO SELECT: **ESC : + Ø + Ø + Ø**

Note that when the printer is switched on, the download set is undefined.)

3. Control code printing selection

As described earlier in this chapter (**ESC I + 1**).

4. Printing of the character

Simply send the download character to the printer.

Example program

The following program (AMSTRAD BASIC) uses the above sequence of operations to create a user defined character (a square box). The parameters d1 to d11 are read from the data at the end of the program; binary numbers have been used to show the relationship between bits and dots.

```
10 PRINT #8,CHR$(27);"&";CHR$(Ø);CHR$(5);CHR$(5);CHR$(11);
20 FOR d=1 TO 11
30   READ c$
40   cell$=CHR$(VAL("&X"+c$))
50   PRINT #8,cell$;
60 NEXT
70 PRINT #8,CHR$(27);"%";CHR$(1);CHR$(Ø)
80 PRINT #8,CHR$(27);"I";CHR$(1)
90 PRINT #8,CHR$(5)
100 '
110 'data for a square box character
120 DATA 1111111
130 DATA ØØØØØØØ
140 DATA 1ØØØØØ1
150 DATA ØØØØØØØ
160 DATA 1ØØØØØ1
170 DATA ØØØØØØØ
180 DATA 1ØØØØØ1
190 DATA ØØØØØØØ
200 DATA 1ØØØØØ1
210 DATA ØØØØØØØ
220 DATA 1111111
```

NOTE - For this 'download character' example program to work, DIP switch DS2-4 must be set to the **ON** position. (See Chapter 7 page 5 for a table of all DIP switch functions.)

REMEMBER - Always switch the printer off before adjusting the DIP switches.

Hexadecimal dump

The DMP2000 has the ability to print (in hexadecimal format) all codes sent to it. To select this mode, the printer must be switched on while the **LF** and **FF** buttons are held down together.

To illustrate this, load a piece of paper into the printer, switch the printer off, then hold down the **LF** and **FF** buttons together while switching the printer on again. The printer is now in the hex dump mode.

Reset your computer and type in:

```
10 REM this is a one-line test program
```

Now `LIST` the program to the printer. The computer will send the program to the DMP2000's print buffer. In the hex dump mode, the buffer's contents will only be printed out when the buffer is full. However, with a short, one-line program (such as above), the buffer will obviously NOT be full, and nothing will be printed out. Under such circumstances, you should set the printer off line, and the contents of the partially-full buffer will be printed out.

To cancel the hex dump mode, simply switch the printer off.

Chapter 7
For your reference....

Subjects covered in this chapter:

Reference information for the user, including:

Technical Specification
Interface
Signal timing
DIP switch functions

Technical Specification

Print system: Impact dot-matrix

Print speed : 105 CPS (normal character)
: 52 CPS (double width character)

Printing characteristics
(vertical x horizontal): 9 x 9 (normal character)
9 x 10 (double width character)
8 x chosen amount (bit-image)
9 x chosen amount (9 pin bit-image)

96 characters ASCII + italics + international character sets

Normal character size: 2.1 (width) x 2.55 (height) mm

Print sizes:	Standard (Pica)	- 10	CPI /	80	CPL
	Mini (Elite)	- 12	CPI /	96	CPL
	Condensed	- 17	CPI /	137	CPL
	Double width Standard	- 5	CPI /	40	CPL
	Double width Mini	- 6	CPI /	48	CPL
	Double width Condensed	- 8.5	CPI /	68	CPL

Number of columns: 80 (standard)
40 (double width)
132 (condensed)
66 (double width condensed)

Line feed rates: 1/6, 1/8, 7/72, n/216 (programmable), n/72 inch (programmable)

Line feed speed: 200 mS (1/6 inch)

Paper type: 4.5 to 10 inch fan-fold (tractor feed)
4 to 9.5 inch cut sheet or roll paper (friction feed)

Number of copies: 2 sheets (incl. original)
40 g/m^2 pressure-sensitive paper train

Interface: Parallel (Centronics compatible)

Mains supply: 220-240 Volt AC 50 Hz

Dimensions: 16 (width) x 10 (depth) x 4 (height) inch (400 x 250 x 100 mm)

Weight: 4.2 kg

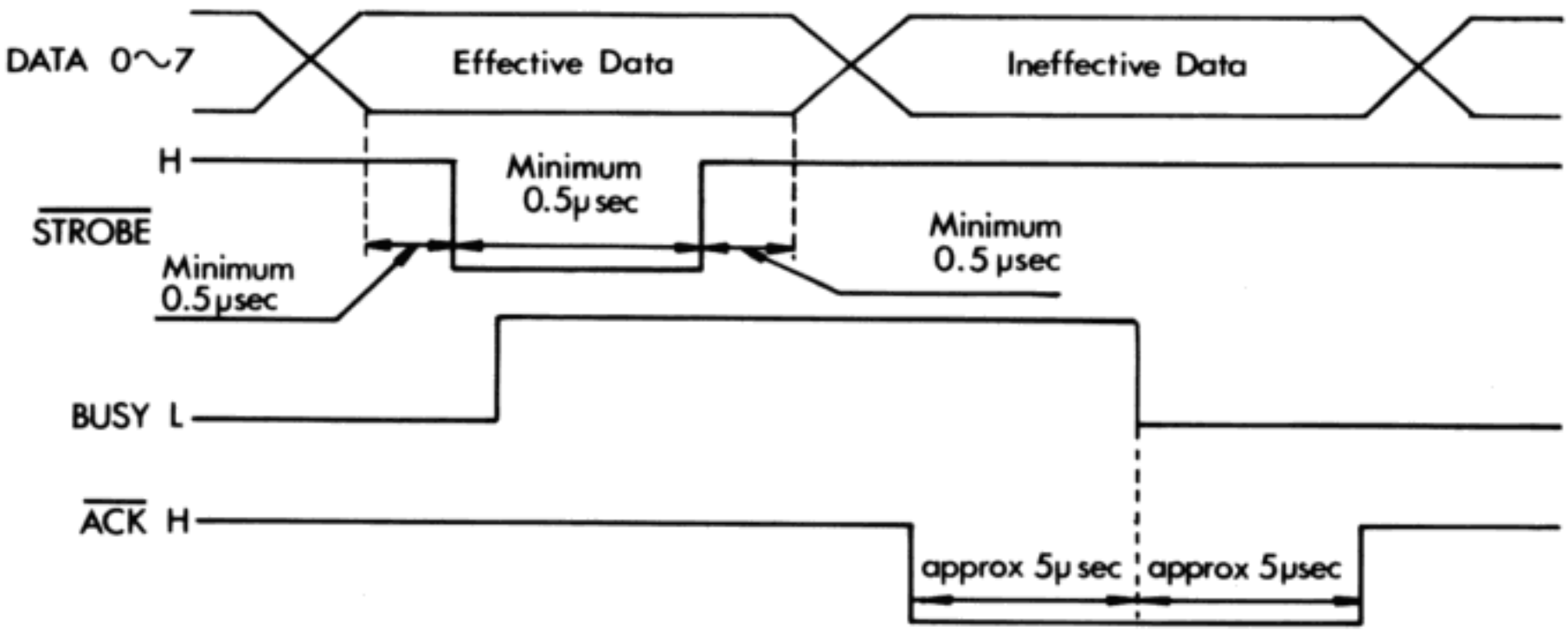
Interface

PIN	DESIGNATION	I/O	DESCRIPTION
1	$\overline{\text{DATA}}$ $\overline{\text{STROBE}}$	IN	Taking pin low enables receiving of DATA 0 to DATA 7. Minimum necessary pulse width is 0.5 μS.
2	DATA 0 (LSB)	IN	8-bit data signal. Taking pin high or low corresponds to 1 and 0 respectively.
3	DATA 1		
4	DATA 2		
5	DATA 3		
6	DATA 4		
7	DATA 5		
8	DATA 6		
9	DATA 7 (MSB)		
10	$\overline{\text{ACKNOWLEDGE}}$	OUT	Active low output pulse generated when data entry and processing are completed. After this signal, subsequent data will be accepted. This signal is also generated when changing from off line to on line.
11	BUSY	OUT	Output high under any of the following conditions: a. Going off line. b. Paper feed or printing operation c. When a control code is received

PIN	DESIGNATION	I/O	DESCRIPTION
12	PE	OUT	Output high when paper is out. (When on line, paper out is sensed after executing a paper feed command. When off line, paper out is always sensed.)
13	SELECT	OUT	On line and off line correspond to high and low respectively. When off line, DATA 0 to DATA 7 are not receivable.
14	NC		
15	NC		
16	OV		
17	CHASSIS GND		
18	+5V	OUT	+5V (50mA max) power supply output.
19	GND		Signal ground.
20	GND		Signal ground.
21	GND		Signal ground.
22	GND		Signal ground.
23	GND		Signal ground.
24	GND		Signal ground.
25	GND		Signal ground.
26	GND		Signal ground.
27	GND		Signal ground.
28	GND		Signal ground.
29	GND		Signal ground.
30	GND		Signal ground.
31	$\overline{\text{INPUT PRIME}}$	IN	Taking pin low initialises printer. Minimum necessary pulse width is 100μS.
32	$\overline{\text{FAULT}}$	OUT	Output low when off line.
33	GND		
34	NC		
35	+5V	OUT	

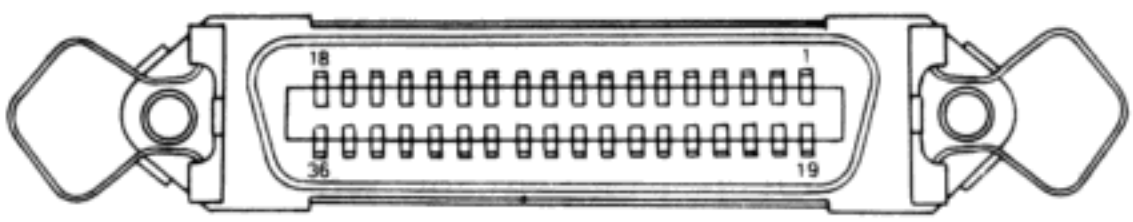
PIN	DESIGNATION	I/O	DESCRIPTION
36	$\overline{\text{SLCT IN}}$	IN	Taking pin low or high sets printer on line or off line respectively (when the printer is not in error condition).

Signal timing



DATA input waveform

Printer socket



Rear view of printer

DIP switch functions

Chapter 2 described how to adjust DIP switches DS1-1, DS1-2, and DS1-3 to select one of the international character sets. The following table indicates the functions of the remaining DIP switches:

SWITCH	FUNCTION	OFF	ON
DS1-1	International characters	See Chapter 2	See Chapter 2
DS1-2	International characters	See Chapter 2	See Chapter 2
DS1-3	International characters	See Chapter 2	See Chapter 2
DS1-4	CR function	CR only	CR & LF
DS1-5	Paper out sensor	Enable	Disable
DS1-6	Page length	11 inch	12 inch
DS1-7	Code unit	8 units	7 units *
DS1-8	Default character set	Standard	NLQ-standard
DS2-1	Zero character	Unslashed	Slashed
DS2-2	Default skip perforation	Disable	Enable
DS2-3	Buffer mode	Character	Graphics
DS2-4	Buffer mode	Character/graphics	Download
DS2-5	SLCT IN signal	Not sent	Automatically sent
DS2-6	Alarm bleeper	Disable	Enable
DS2-7	Default typeface	Bold off	Condensed & Bold on
DS2-8	Default typeface	Condensed off	Bold on
DS2-9	Do not use		
DS2-10	Do not use		

* Set DS1-7 ON when using the printer with an Apple computer.

REMEMBER - Always switch the printer off before adjusting the DIP switches.

Appendix 1
Table of control codes

CODE	DECIMAL	HEX	FUNCTION
BEL	7	&07	Sound bleeper
BS	8	&08	Back space
TAB	9	&09	Horizontal tab jump
LF	10	&0A	Line feed
VT	11	&0B	Vertical tab jump
FF	12	&0C	Form feed
CR	13	&0D	Carriage return
SO	14	&0E	Select double width
SI	15	&0F	Select condensed
DC1	17	&11	Device control 1
DC2	18	&12	Cancel condensed
DC3	19	&13	Device control 3
DC4	20	&14	Cancel double width
CAN	24	&18	Clear buffer
DEL	127	&7F	Delete last character from buffer
ESC SO	27 14	&1B &0E	Select double width
ESC SI	27 15	&1B &0F	Select condensed
ESC ! + parameter	27 33 n	&1B &21 n	Select print mode
ESC #	27 35	&1B &23	Accept eighth bit
ESC % + **Ø** + parameter	27 37 0 n	&1B &25 &00 n	Select internal character set

CODE	DECIMAL	HEX	FUNCTION
ESC % + 1 + parameter	27 37 0 n	&1B &25 &00 n	Select download character set
ESC & + Ø + parameters	27 38 0 n..	&1B &26 &00 n..	Define download character
ESC * + parameters	27 42 n..	&1B &2A n..	Select bit image graphics
ESC - + parameter	27 45 n	&1B &2D n	Select or cancel underline
ESC / + parameter	27 47 n	&1B &2F n	Select tab channel
ESC Ø	27 48	&1B &30	Select 1/8 inch paper feed
ESC 1	27 49	&1B &31	Select 7/72 inch paper feed
ESC 2	27 50	&1B &32	Select 1/6 inch paper feed
ESC 3 + parameter	27 51 n	&1B &33 n	Select variable n/216 inch paper feed
ESC 4	27 52	&1B &34	Select italics
ESC 5	27 53	&1B &35	Cancel italics
ESC 6	27 54	&1B &36	Select printable code area expansion
ESC 7	27 55	&1B &37	Cancel printable code area expansion
ESC 8	27 56	&1B &38	Disable paper-out detection
ESC 9	27 57	&1B &39	Enable paper-out detection
ESC : + Ø + parameters	27 58 0 n..	&1B &3A &00 n..	Copy internal character set into download set

CODE	DECIMAL	HEX	FUNCTION
ESC <	27 60	&1B &3C	Home head
ESC =	27 61	&1B &3D	Unset eighth bit to 0
ESC >	27 62	&1B &3E	Set eighth bit to 1
ESC ? + parameters	27 63 n..	&1B &3F n..	Select/change bit image mode
ESC @	27 64	&1B &40	Reset printer
ESC A + parameter	27 65 n	&1B &41 n	Select variable n/72 inch paper feed
ESC B + parameters + Ø	27 66 n.. 0	&1B &42 n.. &00	Set vertical tabs
ESC C + parameter	27 67 n	&1B &43 n	Set page length (by lines)
ESC C + Ø + parameter	27 67 0 n	&1B &43 &00 n	Set page length (by inches)
ESC D + parameters + Ø	27 68 n.. 0	&1B &44 n.. &00	Set horizontal tabs
ESC E	27 69	&1B &45	Select bold
ESC F	27 70	&1B &46	Cancel bold
ESC G	27 71	&1B &47	Select double strike
ESC H	27 72	&1B &48	Cancel double strike
ESC I + parameter	27 73 n	&1B &49 n	Select or cancel control code printing
ESC J + parameter	27 74 n	&1B &4A n	Select variable n/216 inch one-shot forward feed
ESC K + parameters	27 75 n..	&1B &4B n..	Select single density graphics
ESC L + parameters	27 76 n..	&1B &4C n..	Select double density graphics

CODE	DECIMAL	HEX	FUNCTION
ESC M	27 77	&1B &4D	Select mini
ESC N + parameters	27 78 n..	&1B &4E n..	Select skip perforation
ESC O	27 79	&1B &4F	Cancel skip perforation
ESC P	27 80	&1B &50	Cancel mini
ESC Q + parameter	27 81 n	&1B &51 n	Set right margin
ESC R + parameter	27 82 n	&1B &52 n	Select international character set
ESC S + parameter	27 83 n	&1B &53 n	Select subscript or superscript
ESC T	27 84	&1B &54	Cancel subscript and superscript
ESC U + parameter	27 85 n	&1B &55 n	Select or cancel uni-directional printing
ESC W + parameter	27 87 n	&1B &57 n	Select or cancel double-width
ESC Y + parameters	27 89 n..	&1B &59 n..	Select double speed double density graphics
ESC Z + parameters	27 90 n..	&1B &5A n..	Select quadruple density graphics
ESC ↑ + parameters	27 94 n..	&1B &5E n..	Select 9-pin bit image mode
ESC b + parameters + Ø	27 98 n.. 0	&1B &62 n.. &00	Set tabs in tab channel
ESC i + parameter	27 105 n	&1B &69 n	Select incremental print
ESC j + parameter	27 106 n	&1B &6A n	Select variable n/216 inch one-shot reverse feed
ESC l + parameter	27 108 n	&1B &6C n	Set left margin

CODE	DECIMAL	HEX	FUNCTION
ESC p + parameter	27 112 n	&1B &70 n	Select or cancel proportional
ESC s + parameter	27 115 n	&1B &73 n	Select or cancel half speed printing
ESC x + parameter	27 120 n	&1B &78 n	Select or cancel NLQ

Appendix 2
Some programs for AMSTRAD computer users....

Program 1: 'Electric typewriter'

This program sets up the DMP2000 as a simple typewriter. You may select a range of typefaces and options for the current line being typed. Carriage return/line feeds are performed automatically at the appropriate word breaks, so you can just type away without looking at the screen.

Instructions:

* Make sure that printer is **ON LINE**.

* To start the program: `RUN`.

* Select printer-paper width (in the range `20` to `70`).

* Start typing.

  You can edit the current line using **[DEL]**.
  You can press **[ENTER]** (or **[RETURN]**) to force a carriage return/line feed.
  When a line becomes full (or you issue a carriage return/line feed), the line will be printed out.

* The current typeface is indicated at top of screen.

* To change to an alternative typeface, press **[COPY]**.

  Typeface combinations:

  Standard, Standard subscript, Standard underlined, Subscript & underlined
  Bold, Bold subscript, Bold underlined, Bold subscript & underlined
  Italics, Italics subscript, Italics underlined, Italics subscript & underlined
  NLQ, NLQ subscript, NLQ underlined, NLQ subscript & underlined
  The 'Cancel' option resets to Standard typeface.

* To stop program, press **[ESC]** twice.

```
10 ' DMP2000 ELECTRIC TYPEWRITER by IVOR SPITAL
20 ' copyright (c) AMSOFT 1985
30 '
40 ' SET-UP
50 '
60 MODE 2
70 DEFINT b,p,w
80 WINDOW 1,80,24,24: WINDOW #1,1,80,5,23: WINDOW #2,1,
   80,1,1
90 INPUT #2,"ENTER WIDTH (20-70) ";w: IF w<20 OR w>70
   THEN 90
100 b=w-15: REM sets line wrapping point
110 GOSUB 670
120 LOCATE #1,1,24
130 LOCATE #7,1,25: PRINT #7,STRING$(w,46): WIDTH 255
140 CALL &BB81: REM cursor on
150 '
160 ' GET CHAR
170 '
180 c$=INKEY$: IF c$="" THEN 180
190 IF ASC(c$)=224 THEN GOSUB 730: GOSUB 670
200 IF ASC(c$)>127 THEN 180
210 IF c$=CHR$(127) THEN 590
220 IF c$=CHR$(13) THEN GOSUB 300
230 IF POS(#0)>w THEN GOSUB 360
240 PRINT c$;
250 lin$=rt$+lin$+c$: rt$=""
260 GOTO 180
270 '
280 ' CARRIAGE RETURN
290 '
300 lt$=lin$: rt$=""
310 GOSUB 480
320 RETURN
330 '
340 ' WRAP
350 '
360 IF c$=CHR$(32) THEN 400
370 FOR p=w TO b STEP-1
380   IF MID$(lin$,p,1)=CHR$(32) THEN 420
390 NEXT
400 GOSUB 300
410 RETURN
420 lt$=LEFT$(lin$,p): rt$=RIGHT$(lin$,w-p)
```

```
430 GOSUB 480
440 RETURN
450 '
460 ' END OF LINE
470 '
480 PRINT #2,"PRINTER IS NOT ON LINE"
490 CLS
500 PRINT #8,sx$;sy$;sz$;lt$;cx$;cy$;cz$
510 GOSUB 670
520 PRINT #1,lt$: PRINT rt$;
530 lin$=""
540 IF c$=CHR$(32) THEN c$=""
550 RETURN
560 '
570 ' DELETE CHAR
580 '
590 IF POS(#0)=1 THEN 180
600 IF RIGHT$(lin$,1)=CHR$(32) THEN c$=""
610 PRINT CHR$(8);CHR$(16);
620 lin$=LEFT$(lin$,LEN(lin$)-1)
630 GOTO 180
640 '
650 ' STATUS
660 '
670 IF x$="" THEN x$="STANDARD "
680 PRINT #2,x$;y$;z$
690 RETURN
700 '
710 ' CHANGE TYPEFACE
720 '
730 PRINT #2,CHR$(24);" SELECT:   B=Bold   I=Italics
      N=NLQ   S=Subscript   U=Underline   C=Cancel";CHR$(24)
740 ON INSTR(" binsuc",LOWER$(INKEY$)) GOTO 740,760,
    770,780,790,800,810
750 GOTO 740
760 sx$=CHR$(27)+"E": cx$=CHR$(27)+"F": x$="BOLD ": RETURN
770 sx$=CHR$(27)+"4": cx$=CHR$(27)+"5": x$="ITALICS ": RETURN
780 sx$=CHR$(27)+"x"+CHR$(1): cx$=CHR$(27)+"x"+CHR$(0): x$=
    "NLQ ": RETURN
790 sy$=CHR$(27)+"S"+CHR$(1): cy$=CHR$(27)+"T": y$=
    "SUBSCRIPT ": RETURN
800 sz$=CHR$(27)+"-"+CHR$(1): cz$=CHR$(27)+"-"+CHR$(0):
    z$="UNDERLINED ": RETURN
810 sx$="": sy$="": sz$="": cx$="": cy$="": cz$="": x$="":
    y$="": z$="": RETURN
```

Program 2: 'Text screen dump'

This program allows you to 'dump' the contents of the text screen to the printer. You can use any screen mode (though `MODE 2` may not be too legible). Various colours (or 'shades' on a green monitor) will not be resolved so if you wish to dump a graphics screen, you will probably achieve better results using Program 3 (ahead).

Instructions:

* Before each dump, it is recommended that you switch the printer off, then on again.
* Make sure that printer is **ON LINE**.
* Load the following dump program into the computer, and `RUN`.
  Once run, the program will delete itself from memory.
* Now type in (or load) the text that you wish to dump to the printer.
* When the screen is ready to dump, hold down **[CTRL]** and press **[COPY]**.

  The screen contents will be sent to the printer.
  If you wish to terminate the dump before the entire screen is reproduced, hold down the **[SHIFT]** key until the dump stops.
* You may repeat the above procedure any number of times to reproduce different screens.

```
10 ' TEXT SCREEN DUMP by CLIFF LAWSON
20 ' copyright (c) AMSOFT 1985
30 '
40 ZONE 3: MODE 2: LOCATE 12,10: PRINT "Please wait .";
50 MEMORY HIMEM-353
60 addr=HIMEM+1
70 lin=180: REM first DATA line
80 ON ERROR GOTO 160
90 ps=1: sum=0
100 READ a$
110 n=VAL("&"+MID$(a$,ps,2))
120 ps=ps+3
130 IF ps<26 THEN POKE addr,n: addr=addr+1: sum=(sum+n)
    MOD 256 ELSE IF sum<>n THEN PRINT: PRINT: PRINT
    "DATA error in line",lin: PRINT CHR$(7): MEMORY
    HIMEM+353: END
140 IF ps<27 THEN GOTO 110
150 lin=lin+10: PRINT ".";: GOTO 90
```

```
160 IF ERR=4 AND ERL=100 THEN MODE 2: LOCATE 22,10: PRINT
    "Push [CTRL] and [COPY] to start dump": LOCATE 25,12:
    PRINT "Hold down [SHIFT] to stop dump": CALL HIMEM+1:
    NEW ELSE PRINT "Error : ",ERR,"in",ERL
170 END
180 DATA    21 E1 E9 22 30 00 F7 EB 1F
190 DATA    21 49 01 19 4E 23 46 79 B4
200 DATA    B0 28 15 E5 60 69 19 E5 99
210 DATA    4E 23 46 60 69 19 44 4D 2A
220 DATA    E1 71 23 70 E1 23 18 E4 E5
230 DATA    21 49 00 06 81 0E 00 11 10
240 DATA    2F 00 CD D7 BC C9 C5 D5 F2
250 DATA    E5 F5 3E 09 CD 1E BB 28 EF
260 DATA    0A 3E 17 CD 1E BB 28 03 30
270 DATA    CD 59 00 F1 E1 D1 C1 C9 53
280 DATA    00 00 00 00 00 00 00 00 00
290 DATA    00 00 00 00 00 00 00 00 00
300 DATA    DD 21 52 00 DD 36 01 00 64
310 DATA    DD 36 02 00 DD 36 03 90 BB
320 DATA    DD 36 04 01 DD 36 05 00 30
330 DATA    DD 36 00 00 3E 1B CD CC 05
340 DATA    BB D5 E5 CD 2B BD 3E 41 A9
350 DATA    CD 2B BD 3E 06 CD 2B BD AE
360 DATA    CD 2E BD 38 FB 3E 1B CD 11
370 DATA    2B BD 3E 4B CD 2B BD 3E 64
380 DATA    40 CD 2B BD 3E 01 CD 2B 2C
390 DATA    BD DD 36 00 00 DD 36 06 E9
400 DATA    00 DD CB 00 26 DD 5E 01 0A
410 DATA    DD 56 02 DD 6E 03 DD 66 C6
420 DATA    04 CD F0 BB B7 28 04 DD 3C
430 DATA    CB 00 C6 DD 34 06 DD 7E 03
440 DATA    06 FE 07 28 12 DD 6E 03 93
450 DATA    DD 66 04 2B 2B DD 75 03 F2
460 DATA    DD 74 04 18 CC 18 A9 CD C7
470 DATA    2E BD 38 FB DD 7E 00 CD 46
480 DATA    2B BD DD 34 01 DD 34 01 0C
490 DATA    20 03 DD 34 02 DD 7E 01 92
500 DATA    D6 82 20 07 DD 7E 02 FE DA
510 DATA    02 28 12 DD 6E 03 DD 66 CD
520 DATA    04 11 0C 00 19 DD 75 03 8F
530 DATA    DD 74 04 18 8C DD 34 05 0F
540 DATA    3E 0A CD 2B BD 3E 0D CD 15
550 DATA    2B BD 3E 15 CD 1E BB 20 01
560 DATA    0F DD 36 01 00 DD 36 02 38
570 DATA    00 DD 7E 05 FE 22 20 A5 45
580 DATA    3E 1B CD 2B BD 3E 40 CD 59
590 DATA    2B BD E1 D1 CD C9 BB C9 B4
600 DATA    22 00 29 00 42 00 5B 00 E8
610 DATA    00 00 00 00 00 00 00 00 00
```

Program 3: 'Graphics screen dump'

This sub-routine allows you to 'dump' the contents of a graphics screen to the printer. The screen will be reproduced sideways on the paper to maintain the correct aspect ratio. (Text screens can be dumped using this routine; however, Program 2 is recommended for greater speed and clarity.)

Make sure that the variable '`mde`' in line `10030` is correctly set according to the screen mode that you wish to dump (you may only use either `0` or `1`). Note that the data in lines `10100` and `10110` determine the pen colour (shade) interpretation for each of the two modes, and each datum may be altered (in the range `0` to `15`).

Instructions:

* Before each dump, it is recommended that you switch the printer off, then on again.
* Make sure that printer is **ON LINE**.
* Load the dump sub-routine (on the next page) into the computer - do not run it yet.
* Now type in (or `MERGE`) the program which displays the screen that you wish to dump to the printer. You must ensure that the line numbering of the screen display program doesn't overwrite the dump sub-routine (at lines `10000` to `10380`) ahead. Make sure that the screen display program is terminated by an `END` command. After the screen is displayed, it should be sent to the printer by the command `GOSUB 10000` (which should be included just before the `END` statement). Example screen display program:

```
10 MODE 0
20 DEF FNw=1+RND*24
30 FOR n=0 TO 13
40   WINDOW FNw,FNw,FNw,FNw
50   PAPER n
60   CLS
70 NEXT
80 GOSUB 10000
90 END
```

* The above example will produce some coloured squares on the screen. When the program reaches line `80`, the dump sub-routine (ahead) will be called and the screen reproduced to the printer.
* If you wish to terminate the dump before the entire screen is reproduced, press **[ESC]** twice.

```
10000 ' GRAPHICS SCREEN DUMP by DAVID RADISIC
10010 ' copyright (c) AMSOFT 1985
10020 '
10030 mde=0: REM mde SHOULD BE EITHER 0 OR 1 DEPENDING
      ON MODE
10040 '*****************************************
10050 ' HUE DATA
10060 DATA 7,7,7,7,7,5,7,7,5,7,7,5,7,3,6,7
10070 DATA 3,6,3,7,3,6,3,6,3,2,5,6,5,2,5,2
10080 '
10090 ' HUE TABLE
10100 DATA 0,5,10,15
10110 DATA 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15
10120 '
10130 DIM hue(16,4)
10140 IF mde=1 THEN DIM nn(3): hcnt=3 ELSE DIM nn(15):
      hcnt=15
10150 RESTORE 10060
10160 FOR i=0 TO 7: ii=15-i
10170   FOR j=1 TO 4
10180     READ hue(i,j)
10190     hue(ii,j)=hue(i,j) XOR 7
10200   NEXT
10210 NEXT
10220 PRINT #8,CHR$(27);"@": WIDTH 255
10230 IF mde=1 THEN RESTORE 10100 ELSE RESTORE 10110
10240 FOR i=0 TO hcnt: READ nn(i): NEXT
10250 md=2: st=2: sp=7
10260 PRINT #8,CHR$(27);"3";CHR$(sp);
10270 FOR x=0 TO 638 STEP st
10280   PRINT #8,CHR$(27);"*";CHR$(md);CHR$(32);CHR$(3);
10290   FOR i=0 TO 199
10300     nn=nn(TEST(x,i*2))
10310     PRINT #8,CHR$(hue(nn,1));CHR$(hue(nn,2));
10320     PRINT #8,CHR$(hue(nn,3));CHR$(hue(nn,4));
10330   NEXT
10340   PRINT #8,CHR$(13);CHR$(10);
10350 NEXT
10360 PRINT #8,CHR$(27);"@"
10370 ERASE hue
10380 RETURN
```

Appendix 3
Notice for USA users

This equipment generates and uses radio frequency energy, and if not installed and used properly, that is, in strict accordance with the manufacturer's instructions, it may cause interference to radio and television reception. It has been type tested and found to comply with the limits for a Class B computing device in accordance with the specifications in Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such interference in a residential installation. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

....re-orient the receiving antenna

....re-locate the computer/printer with respect to the receiver

....move the computer/printer away from the receiver

....plug the computer/printer into a different outlet so that the computer/printer and receiver are on different branch circuits

If necessary, the user should consult the dealer or an experienced radio/television technician for additional suggestions. The user may find the following booklet prepared by the Federal Communications Commission helpful:

'How to Identify and Resolve Radio/TV Interference Problems'

This booklet is available from the United States Government Printing Office, Washington DC 20402 - Stock No 004-000-0035-4.

IMPORTANT - WHEN CONNECTING THE PRINTER TO YOUR COMPUTER, A CORRECT UL APPROVED SHIELDED CABLE MUST BE USED.

WARNING - This equipment has been certified to comply with the limits for a Class B computing device, pursuant to Subpart J of Part 15 of FCC Rules.

NOTE - The Amsoft PL-1 printer cable and various items of hardware, software and peripherals (referred to in Chapter 1 Pages 6 and 12) are NOT available in the USA.

Appendix 4
Index

(Note that all references given in this index correspond to chapter and page number, e.g. 7.2 refers to Chapter 7 page 2.)

A

Acknowledge signal 7.2
Alarm bleeper 1.8 1.11 6.3 7.5
AMSTRAD computers 1.6 1.7 1.12 2.1 2.2 2.6 5.1
Apple computer 7.5
ASCII 2.5 7.1

B

Backspace 4.2
BBC Microcomputer 1.6 1.12 2.2 4.1
Bit image graphics 5.1 5.3
Bleeper 1.8 1.11 6.3 7.5
Bold printing 3.8 3.13 7.5
Buffer 2.3 6.3 7.5
Busy signal 7.2

C

Carriage return 2.3 4.1 7.5
Catalog of disc to printer 2.3
Centronics interface 1.6 7.2
Changing typeface 2.7 3.1 7.5
Commodore computer 1.6 1.12 4.1
Condensed typeface 3.4 3.12 7.5
Connecting to computer 1.6
Connecting to mains supply 1.1
Control codes 2.8 4.1 5.1 6.2 6.3
Control layout 1.8
CP/M printing 2.3

D

Data in 7.2 7.4
Deleting characters 6.3
Denmark 2.6 2.7 6.4
Descender 6.5
DIP switches 2.5 6.4 7.5
DIR (CP/M) 2.3
Double density graphics 5.3
Double speed double density graphics 5.3
Double strike printing 3.7 3.13
Double width printing 3.10 3.12
Download characters 6.5
Dump (Hex) 6.8

E

Eight bit operation 5.4 6.2 6.5 7.2
Elite typeface 3.1 7.1
Escape code 2.8 3.11

F

Fault signal 7.3
FF button 1.12
Foreign characters 2.5 6.4
Form feed 1.12 4.2
France 2.6 2.7 6.4
Friction/tractor switch 1.9 1.10 1.14

G

Germany 2.6 2.7 6.4
Graphics 5.1

H

Half speed printing 6.4
Hex dump 6.8
Home head 6.3
Horizontal tab 4.5 4.6

I

Incremental printing 6.1
Ink ribbon 1.4
Input prime signal 7.3
Interface 1.6 7.2
International characters 2.5 6.4
Italics 3.8 3.11
Italy 2.6 2.7 6.4

J

Japan 6.4

L

Layout of controls 1.8
Left margin 4.3
Legs 1.14
LF button 1.10
Line feed 4.1 7.5
Listing a program 2.1
Loading paper 1.9

M

Mains on/off switch 1.8
Mains plug 1.1
Manual paper feed knob 1.10
Margins 4.3
Microsoft BASIC 1.12
Mini typeface 3.2 3.11 3.12
MSX computer 1.12

N

NLQ-standard typeface 3.4 3.13 7.5
NLQ-proportional typeface 3.5

O

Off line 1.11
On line 1.11
On/off switch 1.8
Oric computer 1.12

P

Page length 4.4 7.5
Paper feed knob 1.10
Paper feed rates 4.9
Paper (loading) 1.9
Paper out 1.8 6.3 7.3 7.5
Paper thickness 1.10
Parallel interface 1.6 7.2
Perforation skip 4.4 7.5
Pica typeface 3.1 7.1
POS (AMSTRAD BASIC) 2.3
Power lamp 1.8
Preparation 1.2
Print buffer 2.3 6.3 7.5
Printer cover 1.2
Printer cable 1.6
Printer legs 1.14
Print formatting 2.2 4.1
Print head stabiliser 1.2
Printing 2.1 6.2
PRINT SPC (AMSTRAD BASIC) 2.2
PRINT TAB (AMSTRAD BASIC) 2.2
PRINT USING (AMSTRAD BASIC) 2.2
Proportional typeface 3.3 3.13
Proportional attributes 6.5

Q

Quadruple density graphics 5.3

R

Reset printer 6.2
Reverse paper feed 4.11
Ribbon 1.4
Right margin 4.3

S

Select signal 7.3
Self test printing 1.11
Signal timing 7.4
Sinclair Spectrum 1.6 1.12 3.12
Single density graphics 5.3
Skip perforation 4.4 7.5
Slashed zero 7.5
SLCT IN signal 7.4 7.5
Spain 2.6 2.7 6.4
Specification 7.1
Standard typeface 3.2 7.5
Strobe signal 7.2
Subscript printing 3.6 3.12
Superscript printing 3.6 3.12
Sweden 2.6 2.7 6.4
Switching on 1.8

T

Tab channels 4.5 4.7 4.8
Tabulation 4.5
Technical specification 7.1
Thickness of paper 1.10
Timing 7.4
Tractor feed paper 1.13
Tractors 1.9 1.13
Typefaces 2.7 3.1 3.14 7.5

U

UK 2.6 2.7 6.4
Underlining 3.9 3.11
Uni-directional printing 6.4
Unpacking 1.1
USA 2.6 2.7 6.4
User defined characters 6.5

V

Vertical tab 4.6 4.7

W

WIDTH (AMSTRAD BASIC) 2.2 5.2

Z

Zero slashing 7.5
ZONE (AMSTRAD BASIC) 2.2

Téléchargé sur
Le Vieux Manuel

Http://www.abandonware-manuels.org