Chapter 2
Simple printing exercises....

Subjects covered in this chapter:

Printing and listing in BASIC
Notation used in this manual
Printing DOS files
Wildcards
Listing the disk directory to the printer
Echoing screen output to the printer
Printing a screen dump
Printing GEM files
Printing DOS Plus and CP/M files
The print buffer
Default character set
The DIP switches
How to print international characters
How to change to an alternative typeface
Control codes

Printing in BASIC

Having set up the DMP3000 and printed a word or so, you'll by now have gathered that to send text to the printer, you simply use the BASIC command `LPRINT` followed by the text you wish to print (inside quotes). This not only applies to the printing of 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).

Formatted printing may also be executed using the options: `LPRINT TAB`, `LPRINT USING` and `ZONE`.

Consult your BASIC manual about the use of these commands.

Listing a program in BASIC

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

```
LLIST
```

(Don't forget to always press the [↵] (**[Return]** or **[Enter]**) key on your computer after typing in an instruction.)

Alternatively, you may list a specific line (or range of lines).

Example commands (together with their meanings):

```
LLIST 20-50      (list from line 20 to line 50)
LLIST -200       (list from beginning of program to line 200)
LLIST 80-        (list from line 80 to end of program)
```

To list a program to the printer under Locomotive BASIC 2 (supplied with the AMSTRAD PC), use the mouse to pull down the PROGRAM menu, then select the LIST option.

Notation used in this manual

IMPORTANT: Before we go on to showing you some more example commands, you should note that from now on in this manual, an item shown in angled brackets, eg. ‹item› is NOT to be typed-in literally; the item merely represents the *sort* of information to be entered. Hence, where the manual gives an example command:

```
PRINT ‹filename›
```

....you would typically type in:

```
PRINT letter.jim
```

....where 'letter.jim' is the filename.

Furthermore, items shown in square brackets, eg. [‹item›] are optional and need not be entered unless required. Hence, where the manual gives an example command:

```
PRINT [‹drive›:]‹filename›
```

....you may typically type in:

```
PRINT letter.jim      ....or....      PRINT a:letter.jim
```

....where 'letter.jim' is the filename and 'a' is the (optional) drive. The : colon (which is NOT in angled brackets) *must* be typed-in literally if the optional part of the command is used.

NOTE: The next section in this manual deals with printing *non-BASIC* files under the operating systems: MS-DOS, PC-DOS, GEM, DOS Plus and CP/M. If you are only interested in printing from BASIC, then skip to the section ahead entitled 'Back to BASIC'.

Printing DOS files

Under MS-DOS, PC-DOS or DOS Plus, files may be printed-out using the following commands:

```
PRINT [‹drive›:]‹filename›
```

Example command:

```
PRINT a:autoexec.bat
```

....which will print-out the file 'autoexec.bat' (if it is present on the disk currently in Drive A:).

You may then see a prompt on the screen similar to:

```
Name of list device [PRN]:_
```

....whereupon you should press [↵] **([Return]** or **[Enter])**.

Further messages about the operation in progress will appear on the screen, and the specified file will be printed-out by the DMP3000.

Alternatively, you may use the `COPY` command to send the file to the printer, ie:

```
COPY [‹drive›:]‹filename› PRN:
```

Example command:

```
COPY a:autoexec.bat PRN:
```

After a file is printed-out, a message similar to:

```
1 File(s) copied.
```

....will appear on the screen.

Wildcards

In each of the above types of command, 'wildcards' may be used to specify a range of files to print-out, rather than just one.

There are 2 types of wildcard: ? and *. The ? question mark wildcard may be used to represent one character in a filename which has any value. For example, the filename B?D.B?G could specify any number of files, eg: BAD.BAG, BED.BUG, BID.BIG, BUD.BOG, BCD.B3G, B4D.BEG, etc.

The * asterisk wildcard may represent any group of characters up to the end of the filename field. For example, the filename B*.* could specify any number of files, eg: BIG.TOE, BINARY.DTA, BREATHE.IN, B.TRE, BLANK. etc. Notice how in the last example (BLANK.) the * wildcard specified represents no characters at all. This is a perfectly valid use of the wildcard. Finally, the filename *.* means *all files.*

With wildcards in mind, therefore, you may print-out, for example, all '.BAT' files, using either of the commands:

```
PRINT a:*.bat      ....or....   COPY a:*.bat PRN:
```

Listing the disk directory to the printer

The disk directory may be listed to the printer (under any of DOS systems provided with your PC) by typing the command:

```
DIR >PRN
```

Alternatively, you may switch 'printer-echo' on (described in the next section) and list the directory to the screen.

Echoing screen output to the printer

The **[Ctrl] P** function may be used to echo screen output to the printer, ie. whatever is written to the screen is also printed-out by the DMP3000.

To switch printer-echo on, hold down the **[Ctrl]** key on your computer and press the **P** key once; then release both keys.

To switch printer-echo off, simply press **[Ctrl] P** again.

Until you cancel the first **[Ctrl] P** by pressing **[Ctrl] P** a second time, *everything* which is written to the screen (including > prompts and error messages) will be printed-out by the printer. (Note that pressing **[Ctrl] P** doesn't itself produce a printed character on the screen.)

To experiment with printer-echo, press **[Ctrl] P** and type:

```
DIR
```

....then press **[Ctrl] P** again.

Printing a screen dump

(Providing the facility is installed within the operating system you are currently using on your PC), you may *dump* the entire contents of the screen (as you see it) to the printer, using the ⇧ (shift) **[Prt Sc]** function.

Simply hold down the ⇧ (shift) key and press the **[Prt Sc]** key once.

Printing GEM files

GEM files may be printed-out using the 'Print Spooler' option from the GEM Desktop menu. For further information, see your PC manual.

Printing DOS Plus and CP/M files

Under the DOS Plus and CP/M operating systems, files may be sent to the printer using the `PIP` command. The form of the command is:

```
PIP LST:=[‹drive›:]‹filename›
```

Example command:

```
PIP LST:=a:autoexec.bat
```

....which will print the file 'autoexec.bat' (if it is present on the disk currently in Drive A:).

Unlike MS-DOS and PC-DOS, you may NOT specify wildcards within the filename using the above command.

Back to BASIC

Welcome back to those of you who skipped here from earlier in the chapter. We are now going to explore the features of the DMP3000 and you should note that from here-on, all the example commands will be shown in BASIC.

So, if you haven't already done so, start up BASIC on your PC and be ready to type in some commands.

The print buffer

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

To explain the above, disregard the printer for a moment and concentrate on the screen.

If you compare the results of the following two programs:

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

RUN

123
456    - results on screen
789
```

....and....

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

RUN

123   456         789        - results on screen
```

....you can see that the semicolon and comma at the end of lines 10 and 20 have suppressed the carriage return and line feed on the screen.

Now modify the last program so that the three numbers are sent to the printer instead of the screen, ie:

```
10 LPRINT 123;
20 LPRINT 456,
30 LPRINT 789

RUN

123  456          789
```

- results on printer

Like the previous example, the results are printed-out on one line because the semicolon and comma have suppressed the carriage return/line feed to the printer. The data to be printed (in lines 10 and 20) is held inside the print buffer until the whole line is printed-out by an 'un-suppressed' `LPRINT` command (line 30).

REMEMBER: The `LPRINT` command (when it is not terminated by a semicolon or comma) automatically executes a carriage return and line feed.

Note that the buffer will always be flushed under the following conditions:

1. When the buffer is full.
2. When the printer is set off line.
3. When the printer receives a line feed.

Default character set

By default, the DMP3000 is supplied to you factory-set to reproduce IBM character set #2 (see appendix 2, table 3.2).

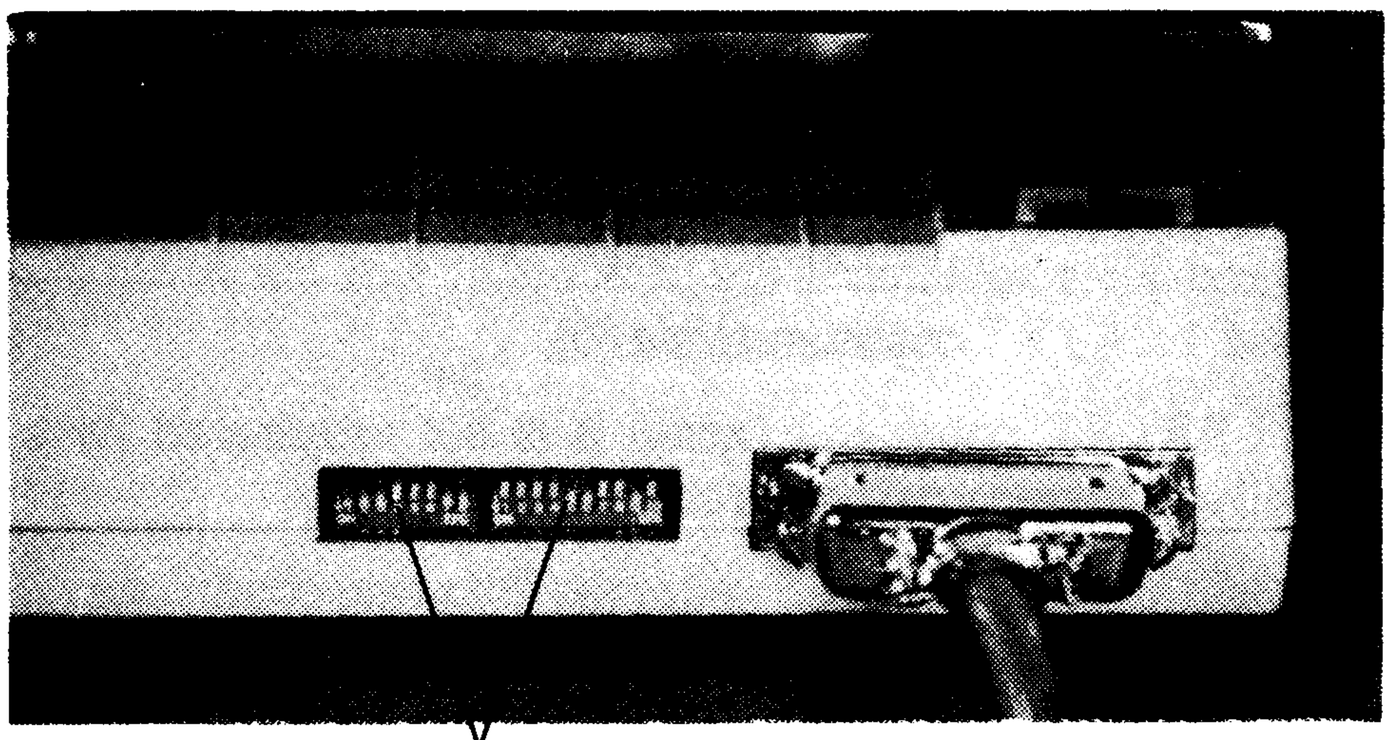
Type in the following test program to print-out part of the default character set:

```
10 FOR n=32 TO 126
20     LPRINT CHR$(n);
30 NEXT
40 :
50 FOR n=160 TO 254
60     LPRINT CHR$(n);
70 NEXT
80 LPRINT
RUN
```

The default character set may, however, be altered by using a series of miniature switches (called DIP switches) at the rear of the printer.

How to adjust the DIP switches

IMPORTANT: Always switch the printer *off* before adjusting the DIP switches.



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 individual switch is numbered, and on the corner of each bank, you will see the word 'ON' (showing you the direction to switch on).

The two switches that select the default character set are numbers 7 and 8 on the first bank (DS1). From now on, we will refer to these switches as DS1-7 and DS1-8.

The following table indicates the different settings of DS1-7 and DS1-8 required to select the appropriate default character set.

CHARACTER SET	DS1-7	DS1-8
Epson FX - standard	OFF	OFF
Epson FX - NLQ	ON	OFF
IBM #1	OFF	ON
IBM #2	ON	ON

We will now set both DS1-7 and DS1-8 to the *off* position. This cancels IBM character set #2 and selects the Epson FX - standard character set. If you have difficulty in adjusting the

small DIP switches by finger, you may find it easier to use the tip of a ball-point pen or similar object.

Now switch the printer *on* and RUN the test program again. Note the different default character set reproduced:

```
 !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmno
pqrstuvwxyz{|}~ /"#$%&'()*+,-./0123456789:;<=>?@ABCD
EFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
```

For the moment, leave the DIP switches set to reproduce the Epson FX - standard character set.

NOTE: The default character set may also be selected via software. This method is described in chapter 6 ahead, under the section entitled 'Character table selection'.

International characters

What you see on the keyboard and the screen is not always what you get on the printer! To illustrate this point, look at the £ sign shown above the 3 key on the top line of your PC keyboard.

Now type in the command:

```
LPRINT "£5"
```

The screen display corresponds to what you just typed. However, look at what's been printed-out by the DMP3000:

```
#5
```

The # hash sign has been printed-out instead of the £ pound sign because the DMP3000 is supplied to you factory-set to reproduce the *USA* ASCII character set by default. (ASCII stands for American Standard Code for Information Interchange.)

Fortunately, for those of you who wish to print £ pound instead of # hash, the DMP3000 may be adjusted to do so by use of DIP switches at the rear of the printer.

The three DIP switches that control the printing of international characters are numbers 1, 2 and 3 on the first bank (DS1). Again, these will be referred to as DS1-1, DS1-2 and DS1-3.

IMPORTANT: To print international characters, IBM character set #1 or #2 must NOT be selected - ie. DIP switch DS1-8 must be *off.*

The following table indicates the settings of DS1-1, DS1-2 and DS1-3 to select the required international characters:

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

Switch the printer off and adjust the DIP switches to the *UK* setting (DS1-1 *off*, DS1-2 *off*, DS1-3 *on*), then switch the printer on again.

Now type in the command:

```
LPRINT "£5"
```

....and you will see that the £ pound sign has been correctly reproduced.

Here is a table of the available international characters:

	CHARACTER CODE (HEX)											
	&23	&24	&40	&5B	&5C	&5D	&5E	&60	&7B	&7C	&7D	&7E
USA	#	$	@	[	\	]	^	`	{	¦	}	~
France	#	$	à	°	ç	§	^	`	é	ù	è	¨
Germany	#	$	§	Ä	Ö	Ü	^	`	ä	ö	ü	ß
UK	£	$	@	[	\	]	^	`	{	¦	}	~
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	}	~

NOTE: International character sets may also be selected via software. This method is described in chapter 6 ahead, under the section entitled 'International character set selection'.

You may now re-adjust the settings of DS1-7, DS1-8 (default character set) and DS1-1, DS1-2, DS1-3 (international characters) to suit your particular printing requirements.

(The functions of the remaining DIP switches (DS1-4 to DS1-6, and DS2-1 to DS2-10) are described in chapter 7 of this manual.)

How to change to an alternative typeface

The DMP3000 is capable of reproducing many combinations of different print style, or 'typeface'.

Here's one to try out. Type in:

```
LPRINT CHR$(27) + "x" + CHR$(1)
LPRINT "this is NLQ printing"
```

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:

```
LPRINT CHR$(27) + "x" + CHR$(Ø)
LPRINT "this is standard printing"
```

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

What is a control code?

A control code is used to call into action a function of the computer, while the code itself is not generally printed. `CHR$(7)` is a control code that makes the computer or the printer 'bleep'. Just try:

```
PRINT CHR$(7)      ....or....    LPRINT CHR$(7)
```

Now take a look at the commands we used to select and cancel NLQ printing. Notice that the first `LPRINT` statement had three parts (which were joined to each other by + plus signs). The three parts were:

1. CHR$(27)

2. "x"

3. CHR$(1) ....or.... CHR$(Ø)

Explanation

CHR$(27)

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

"x"

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.

CHR$(1)
....or....
CHR$(Ø)

The CHR$(1) or CHR$(Ø) part can be thought of as a switch to turn the particular function on or off. As you can see from the above examples, CHR$(1) switches the function *on*; CHR$(Ø) switches it *off*.

NOTE: In many of the examples that follow, the ASCII codes 'SOH' and 'NUL' are shown. These codes should be typed in as CHR$(1) and CHR$(Ø) respectively.

A 'shorthand' means of combining escape code parameters such as "x" + CHR$(1) (shown in the previous example) is to simply type in "x1", ie. to select NLQ, type:

```
LPRINT CHR$(27) + "x1"
```

....and to cancel NLQ, type:

```
LPRINT CHR$(27) + "xØ"
```

Chapter 3
Selecting print styles....

Subjects covered in this chapter:

Choice of styles
Cancelling your choice
Combined styles
Illegal combinations

Choice of style

The DMP3000 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 may 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 your *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 code required to select or cancel the typeface, together with an example command.

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

Example command:

```
LPRINT CHR$(27) + "M"
LPRINT "this is mini typeface"
```

TO CANCEL: ESC P

Example command:

```
LPRINT CHR$(27) + "P"
LPRINT "this is standard typeface again"
```

Proportional typeface

TO SELECT: ESC p SOH

Example command:

```
LPRINT CHR$(27) + "p" + CHR$(1)
LPRINT "this is proportional typeface"
```

TO CANCEL: ESC p NUL

Example command:

```
LPRINT CHR$(27) + "p" + CHR$(Ø)
LPRINT "this is standard typeface again"
```

Condensed typeface

TO SELECT: SI (or ESC SI)

Example command:

```
LPRINT CHR$(15)
LPRINT "this is condensed typeface"
```

TO CANCEL: DC2

Example command:

```
LPRINT CHR$(18)
LPRINT "this is standard typeface again"
```

NLQ-standard typeface

TO SELECT: ESC x SOH

Example command:

```
LPRINT CHR$(27) + "x" + CHR$(1)
LPRINT "this is NLQ typeface"
```

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

TO CANCEL: ESC x NUL

Example command:

```
LPRINT CHR$(27) + "x" + CHR$(Ø)
LPRINT "this is standard typeface again"
```

NLQ-proportional typeface

TO SELECT: ESC x SOH ESC p SOH

Example command:

```
LPRINT CHR$(27)+"x"+CHR$(1)+CHR$(27)+"p"+CHR$(1)
LPRINT "this is NLQ-proportional typeface"
```

TO CANCEL: ESC x NUL ESC p NUL

Example command:

```
LPRINT CHR$(27)+"x"+CHR$(Ø)+CHR$(27)+"p"+CHR$(Ø)
LPRINT "this is standard typeface again"
```

Selecting additional functions

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

Subscript option

TO SELECT: ESC S SOH

Example command:

```
LPRINT CHR$(27) + "S" + CHR$(1)
LPRINT "this is subscript option"
```

TO CANCEL: ESC T

Example command:

```
LPRINT CHR$(27) + "T"
LPRINT "option cancelled"
```

Superscript option

TO SELECT: ESC S NUL

Example command:

```
LPRINT CHR$(27) + "S" + CHR$(Ø)
LPRINT "this is superscript option"
```

TO CANCEL: ESC T

Example command:

```
LPRINT CHR$(27) + "T"
LPRINT "option cancelled"
```

Double strike option

TO SELECT: ESC G

Example command:

```
LPRINT CHR$(27) + "G"
LPRINT "this is double strike option"
```

TO CANCEL: ESC T

Example command:

```
LPRINT CHR$(27) + "H"
LPRINT "option cancelled"
```

Italics option

TO SELECT: ESC 4

Example command:

```
LPRINT CHR$(27) + "4"
LPRINT "this is italics option"
```

TO CANCEL: ESC 5

Example command:

```
LPRINT CHR$(27) + "5"
LPRINT "option cancelled"
```

Bold option

TO SELECT: ESC E

Example command:

```
LPRINT CHR$(27) + "E"
LPRINT "this is bold option"
```

TO CANCEL: ESC F

Example command:

```
LPRINT CHR$(27) + "F"
LPRINT "option cancelled"
```

Selecting underline or double-width printing

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

Underline option

TO SELECT: ESC - SOH

Example command:

```
LPRINT CHR$(27) + "-" + CHR$(1)
LPRINT "this is underline option"
```

TO CANCEL: ESC - NUL

Example command:

```
LPRINT CHR$(27) + "-" + CHR$(Ø)
LPRINT "option cancelled"
```

Double-width option

TO SELECT: SO (or ESC W SOH)

Example command:

```
LPRINT CHR$(14)
LPRINT "this is double width option"
```

TO CANCEL: DC4 (or ESC W NUL)

Example command:

```
LPRINT CHR$(20)
LPRINT "option cancelled"
```

Combining styles

Now that you know how to access each of the different typefaces, you might like to combine a few of them.

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 command (using mini typeface with italics and underline):

```
LPRINT 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$(0) + CHR$(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 the above 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 first assign the escape codes into string variables as follows:

```
ms$ = CHR$(27) + "M"                : REM (mini select)
mc$ = CHR$(27) + "P"                : REM (mini cancel)
is$ = CHR$(27) + "4"                : REM (italics select)
ic$ = CHR$(27) + "5"                : REM (italics cancel)
us$ = CHR$(27) + "-" + CHR$(1)      : REM (underline select)
uc$ = CHR$(27) + "-" + CHR$(0)      : REM (underline cancel)
```

....and thereafter use the string variables' names to select or cancel a particular typeface. Our above example then looks like this:

```
LPRINT ms$ + "You can emphasise a point using" + is$ +
" italics" + ic$ + " or" + us$ + " underline" + uc$ + mc$
```

You can see that the example command is now shorter and much clearer. In addition you could use the escape code 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; no doubt you'll incorporate something along these lines into your own text creation utility program.

So let's create the beginnings of such a program. We'll use the above escape code variables and add a few more typefaces and options as we go along. Furthermore, to save you repeatedly typing in CHR$(27) and CHR$(1) and CHR$(Ø) we have assigned these characters to the variables e$, s$ and n$ respectively. Line numbers have also been added in case you wish to run the program more than once or save it. (Note that you need not type in the REM statements.)

```
10  REM printer control codes
20  e$ = CHR$(27)          : REM escape (ESC character)
30  s$ = CHR$(1)           : REM on (SOH character)
40  n$ = CHR$(Ø)           : REM off (NUL character)
50  ms$ = e$ + "M"         : REM (mini select)
60  mc$ = e$ + "P"         : REM (mini cancel)'
70  is$ = e$ + "4"         : REM (italics select)
80  ic$ = e$ + "5"         : REM (italics cancel)
90  us$ = e$ + "-" + s$ : REM (underline select)
100 uc$ = e$ + "-" + n$ : REM (underline cancel)
110 ws$ = CHR$(14)         : REM (double width select)
120 wc$ = CHR$(2Ø)         : REM (double width cancel)
130 cs$ = CHR$(15)         : REM (condensed select)
140 cc$ = CHR$(18)         : REM (condensed cancel)
```

```
RUN
```

Now try this example:

```
LPRINT 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 escape code variables program:

```
150 nss$ = e$ + "x" + s$ : REM (NLQ select)
160 nsc$ = e$ + "x" + n$ : REM (NLQ cancel)
170 sbs$ = e$ + "S" + s$ : REM (subscript select)
180 sps$ = e$ + "S" + n$ : REM (superscript select)
190 ssc$ = e$ + "T"      : REM (subscript and superscript cancel)
```

```
RUN
```

(Note that the string variable s s c $ cancels BOTH the subscript and the superscript options.)

Example command (using NLQ-standard typeface with subscripts and superscripts):

```
LPRINT nss$ + "Subscripts include H" + sbs$ + "2" + ssc$ +
"O and Log" + sbs$ + "10" + ssc$ + ", while superscripts
include 10" + sps$ + "-3" + ssc$ + " and 100" + sps$ + "o" +
ssc$ + "C." + nsc$
```

A few more assorted typefaces for your program:

```
200 ps$ = e$ + "p" + s$ : REM (proportional select)
210 pc$ = e$ + "p" + n$ : REM (proportional cancel)
220 ds$ = e$ + "G"      : REM (double strike select)
230 dc$ = e$ + "H"      : REM (double strike cancel)
240 bs$ = e$ + "E"      : REM (bold select)
250 bc$ = e$ + "F"      : REM (bold cancel)

RUN
```

Example commands (using proportional, double strike, and bold typefaces):

```
LPRINT ps$ + "Liberal/SDP members will favour the
proportional representation of their print-out" + pc$

LPRINT "Trade unionists will vote for the " + ds$ + "double
strike " + dc$ + "option"

LPRINT "Your mission: " + bs$ + "to boldly print like no
other printer" + bc$
```

In addition to the individual commands shown in this chapter, you may also select from a limited range of typeface combinations using the single escape code ESC ! ‹n›. This code is described in chapter 6 under the section entitled 'Print mode selection'.

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 following table 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: If, in the examples ahead, you wish to follow a control code by your own further printer instructions, then you should terminate the code with a semicolon to suppress the carriage return/line feed which will otherwise be automatically sent after the code.

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

Example command:

```
LPRINT 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

Example command:

```
LPRINT CHR$(10)
```

Backspace

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

TO SELECT: BS

Example command:

```
LPRINT CHR$(8)
```

Note that backspace will not operate during proportional printing.

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

Example command:

```
LPRINT CHR$(12)
```

Margins

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

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›

Example command:

```
LPRINT CHR$(27) + "l" + CHR$(2Ø)
```

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›

Example command:

```
LPRINT CHR$(27) + "Q" + CHR$(5Ø)
```

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 setting (described ahead) will not alter the page length.

TO SELECT: ESC C ‹n›

Example command:

```
LPRINT 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 NUL ‹n›

Example command:

```
LPRINT 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›

Example command:

```
LPRINT CHR$(27) + "N" + CHR$(5)
```

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

Example command:

```
LPRINT 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 DMP3000 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 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› NUL

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 CHR$(Ø).

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

Example command:

```
LPRINT 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: HT

Example command:

```
LPRINT 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› NUL

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 `CHR$(0)`.

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

Example command:

```
LPRINT 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

Example command:

```
LPRINT 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.... NUL) 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› NUL

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 CHR$(Ø).

Example command:

```
LPRINT CHR$(27) + "b" + CHR$(7) + CHR$(1Ø) + CHR$(2Ø) + CHR$(Ø)
```

Channel selection

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

TO SELECT: ESC / ‹channel›

Example command:

```
LPRINT 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 (note that this is the *number* Ø)

Example command:

```
LPRINT CHR$(27) + "Ø"
```

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

TO SELECT: ESC 1

Example command:

```
LPRINT CHR$(27) + "1"
```

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

TO SELECT: ESC 2

Example command:

```
LPRINT 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›

Example command:

```
LPRINT 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›

Example command:

```
LPRINT 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›

Example command:

```
LPRINT 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›

Example command:

```
LPRINT CHR$(27) + "j" + CHR$(108)
```

WARNING: Do not attempt a reverse paper feed while printing within the top 30mm or the bottom 80mm of the paper, (or within 30mm of the perforations on tractor feed paper).

Chapter 5
Graphics printing....

Subjects covered in this chapter:

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

What is graphics printing?

In contrast to normal character printing mode, the DMP3000 offers what's known as 'graphics mode'. When set to this mode, the printer has a different way of interpreting the data it receives; instead of a character being printed *as is*, the data is used to directly control each of the tiny individual 'pins' inside the print head. For each item (byte) of data received, only one line of dots is drawn vertically. Whether or not a dot is drawn depends upon 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'.

In graphics mode, control codes are also 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. Moreover, the graphics mode does NOT itself execute a carriage return and line feed automatically when it reaches the end of a line.

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 dot graphics before reverting to normal character printing mode (in order to execute a carriage return, line feed, etc.). These parameters are known as ‹n1› and ‹n2›. The ‹n2› parameter represents the number of whole 256-dot portions to be printed, while the ‹n1› parameter represents the remainder of individual dots. Both of the parameters ‹n1› and ‹n2› must be in the range 0 to 255.

Hence, for example, if you were specifying a code to produce a graphics dump of a screen image 640 pixels wide, the parameter ‹n2› would be set to 2 (as there are 2 x 256 in 640), while ‹n1› would be set to 128 (the remainder). Such an example command may look like this:

```
LPRINT CHR$(27) + "L" + CHR$(128) + CHR$(2)
```

To calculate the values for ‹n1› and ‹n2›, you may use the following program:

```
10 INPUT "number of dots";d
20 PRINT "<n1> =";  d MOD 256
30 PRINT "<n2> =";  INT(d/256)

RUN
```

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.

Type in the following example program:

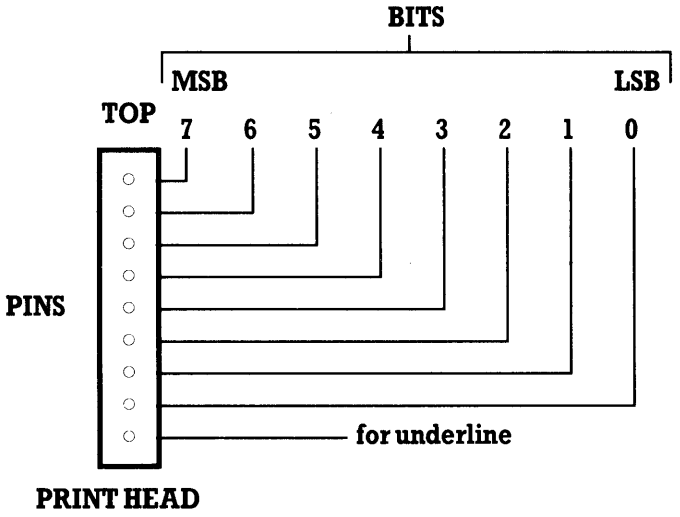
```
10 LPRINT CHR$(27) + "K" + CHR$(126) + CHR$(0);
20 FOR n=1 TO 126
30     LPRINT CHR$(87);
40 NEXT

RUN
```

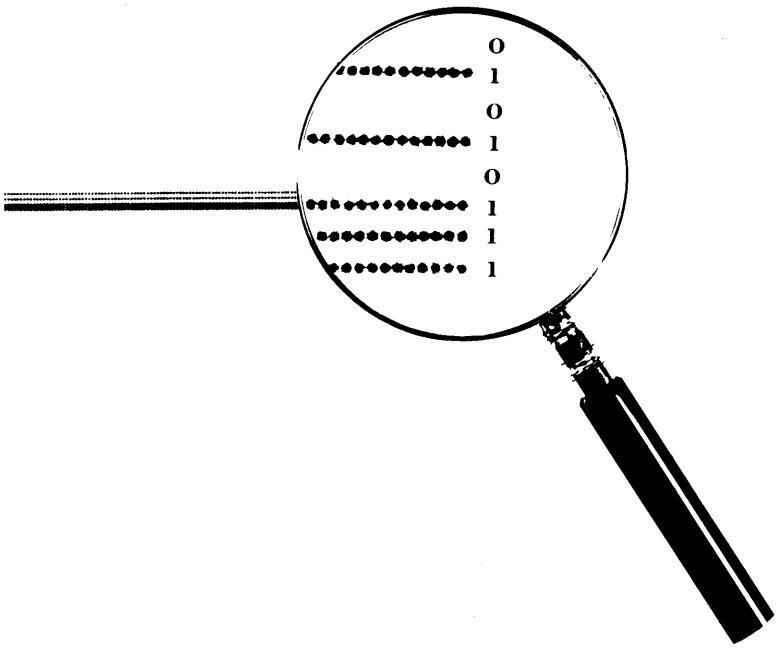
Note that the use of a semicolon to terminate the print statement (in line 10) which selects graphics mode is essential, as this suppresses the carriage return/line feed normally executed after a print statement, which would otherwise result in spurious dot patterns.

We have used 8 7 as our data in line 30 of the above example. This corresponds to the 8-bit binary number 01010111. Looking at the number in binary form you can see the effect of each 1 and 0 on the result.

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.



In the above example program, therefore, the graphics mode is selected in line 10, then the FOR....NEXT loop (lines 20 to 40) prints-out the binary data 01010111 as a column of dots downwards, 126 times along the paper. The result should look like this:



Try experimenting with different numbers in line 30 of the above example program to make sure that you understand the bit-to-dot correspondence.

Graphics modes

There is more than one graphics mode. Each of the modes (summarised below) offer differing combinations of density and speed.

REMEMBER: For each of the following commands, the parameters ‹n1› and ‹n2› are in the range 0 to 255.

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	Head speed (inch/sec) DMP3160	DMP3000
0	480 single density	480	16	10.5
1	960 double density	960	8	5.25
2	960 double speed/double density	480	16	10.5
3	1920 quadruple density	960	8	5.25
4	640 CRT graphic	640	8	5.25
5	576 plotter graphic	576	13	8.7
6	720 CRT graphic	720	8	5.25

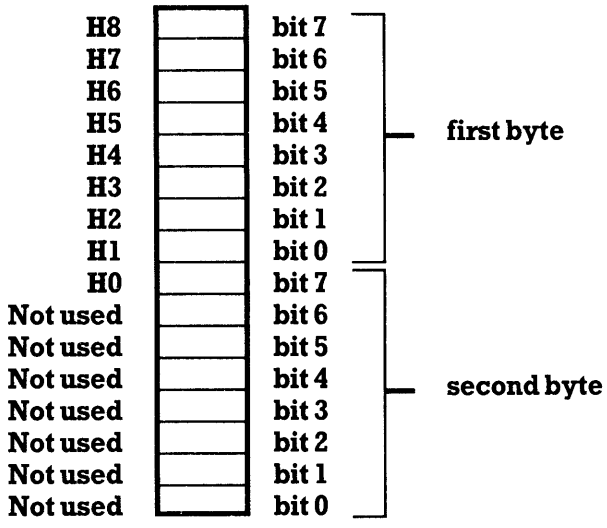
9-pin bit image mode

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 the data for the bottom pin:



Bit image mode selection/change

TO SELECT: ESC ? ‹code› ‹mode›

....where the ‹code› parameter is one of the escape 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).

Graphics character alignment

NOTE: When printing-out graphics characters or boxes whose vertical lines are to be continuous down the page (or when using ready-made graphics characters from IBM character sets #1 and #2), you should additionally select the code for uni-directional printing (ESC U SOH) described in chapter 6 ahead. This will afford more accurate alignment of the vertical lines.