Commodore
CBM Printer

User's Manual

Model 4022

(C= commodore

Commodore
CBM Printer

User's Manual

Model 4022

Part Number 4022025

⊏ commodore

Table Of Contents

Section 1

GENERAL INFORMATION

With the purchase of a Commodore CBM Printer, you have added a great deal of versatility and convenience to the use of your Commodore Computer. You have purchased the Tractor Feed Model 4022. While this manual contains all the information you need to check out, connect, and operate your printer, you should also refer to other Commodore manuals to get the most out of your computer system.

> NOTE TO BUSINESS KEYBOARD USERS ONLY: If you wish to duplicate the examples in this manual and you are using the CBM Business Keyboard computer, you should type the BASIC command POKE 59468,12 at the beginning of each session with your computer and printer.

DESCRIPTION

Your CBM printer is designed to operate through software control. It prints upper- and lower-case alphabetic characters, numeric characters, and all the graphic characters available on your Commodore computer, and even a custom user-defined character. Additionally, your printer has considerable formatting capability owing to its own internal microprocessor system.

Printing Characteristics

Your printer employs an Epson print head. The eight-wire heavy-duty jeweled head has a life expectancy of 50 million characters. If a print head malfunction should occur, obtain authorized technical assistance. Failure to do this may void your warranty.
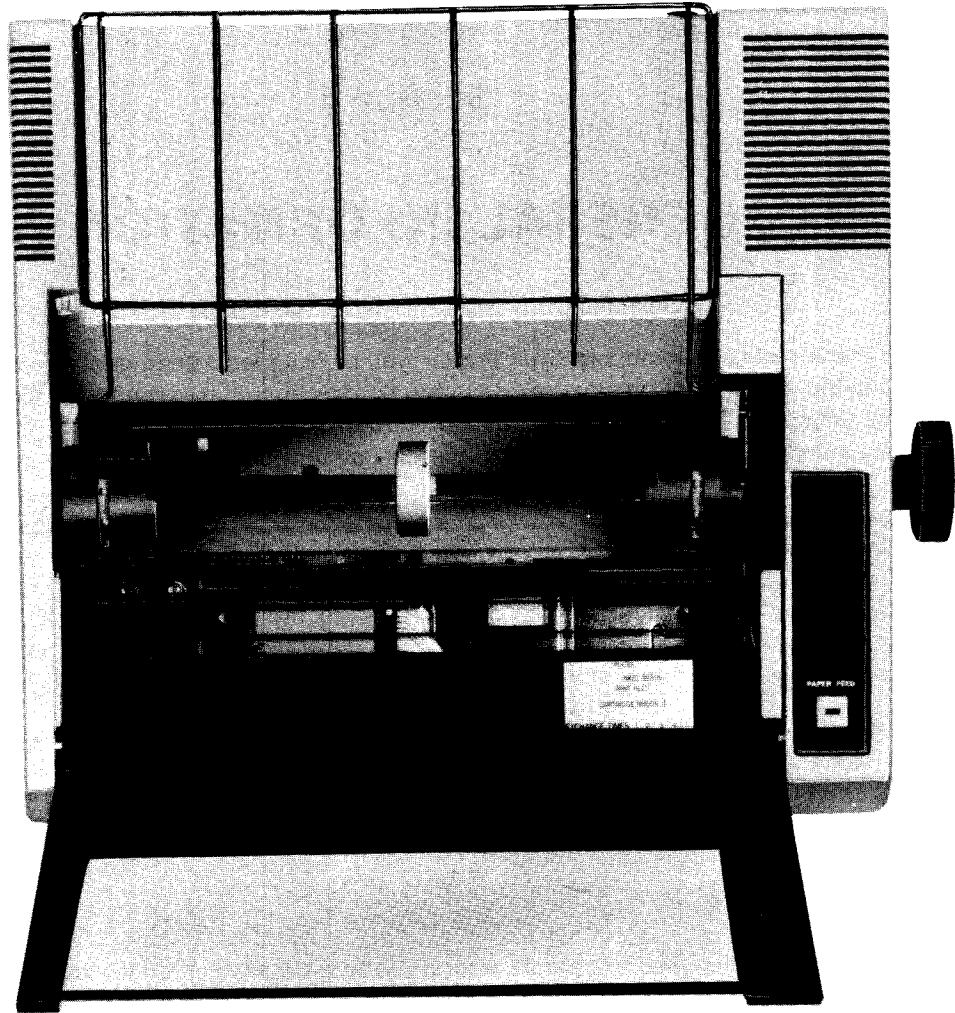
Figure 1. Feed Mechanism and Ribbon Cartridge

IEEE-488 Interface

Both printer models conform to IEEE-488 interface requirements and may be connected directly to your Commodore computer. The printers, designed to be used with the Commodore CBM Floppy Disk Drive, Model 2040, may be daisy-chained with other IEEE-488 devices.

Paper Feed Mechanisms

The Model 4022 Printer employs a tractor feed mechanism that involves the use of sprocket holes in the paper edges. This model is especially useful for the printing of business forms. See Figure 1.

Printer Ribbon

The printer ribbon (shown in Figure 2) is an easily-loaded cartridge.

Internal Microprocessor System

Your printer contains a microprocessor system that resets and executes a diagnostic and initialization sequence when power is applied. In addition, it contains a random access memory (RAM) in which you can store formatting data. Because your printer is an "intelligent" peripheral, it uses none of your computer's memory.

Specifications

Table 1 shows the specifications for the Model 4022 Printer.

TABLE 1

Model 4022 Printer Specifications	
Printing Method	Serial Impact Dot Matrix
Print Rate	30 lpm with 80 columns printed 50 lpm with 40 columns printed 80 lpm with 20 columns printed
Print Direction	Unidirectional
Column Capacity	80
Character Font	5 × 8
Column Spacing	1/10" — 10 characters per inch
Line Spacing	Programmable
Character Size	.11" high, .08" wide
Copies	3, including original
Ribbon Type	Cartridge
Ribbon Life	2×10^6 characters
Ribbon Cartridge	Commodore P/N 905242-01
Paper Width	10" computer folded paper
Forms	8.5 + .5 × 2 (sprocket margins) Pin-to-pin distance: .5" longitudinally 9.5" laterally 5/32" diameter

UNPACKING YOUR PRINTER

Before you unpack your printer, inspect the shipping carton for signs of damage. If it appears to be damaged, be especially careful when you inspect its contents. DON'T throw away any of the packaging material until you have located all the contents of the carton! The package should contain:

1. Commodore Printer, Model 4022

2. User Manual Number 4022025

3. Warranty card

4. Printer ribbon cartridge

5. Package of paper

6. Miscellaneous informational literature

If any of the first five is missing or damaged, notify your Commodore dealer immediately.

Additionally, you must obtain the appropriate cable from your Commodore dealer to connect your printer to your computer. See Section 2, page 8, for information about which IEEE cable to get.

Section 2

PREPARING TO USE YOUR PRINTER

Before starting to use your printer, you should make sure that it is in working condition. This procedure includes checking for obstructions in the path of the print head or paper feed and that the printer ribbon is properly in place. Follow this procedure:

1. Lift the plastic paper cover gently to expose print head and mechanism.

2. Carefully remove any foreign material that may have strayed into the mechanism or electronics.

3. Install the printer ribbon cartridge as shown in Figure 2.

4. Replace the plastic paper cover.

5. Insert the wire paper rest in the two holes in the rear of the cabinet. Do not attempt to load the paper at this time.
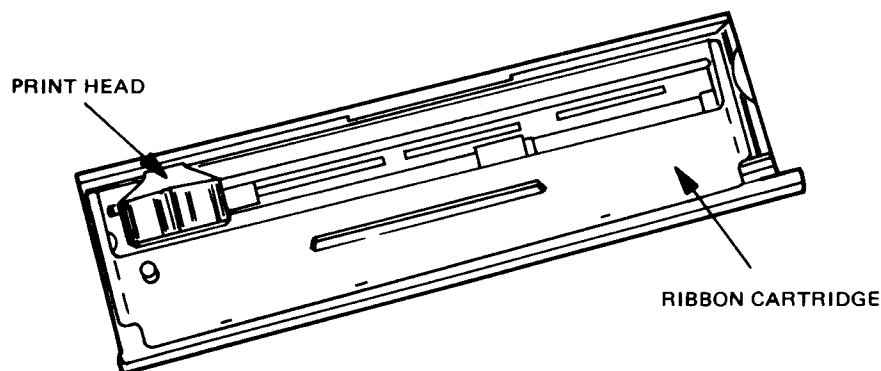
Figure 2. The Printer Cartridge

CONNECTING THE PRINTER TO YOUR COMPUTER

One of two connector cables is required to connect the printer to the computer:

1. PET-to-IEEE cable, P/N 320202
   Use this cable if the printer is to be the only (or first) IEEE device connected to your computer.

2. IEEE-to-IEEE cable, P/N 905080
   Use this cable if you will be using your printer in conjunction with the Dual Drive Floppy Disk Model 2040.

Follow these steps to connect your printer to your computer:

1. Turn off the AC power to your computer.

2. Place the printer in a convenient location close to the computer. DO NOT connect the printer to an AC outlet at this time.

3. If you are using the PET-to-IEEE cable, connect the cable between the IEEE-488 interface connector on your computer and the connector on the printer as shown in Figure 3. If you are using the IEEE-to-IEEE cable, connect the printer in a configuration similar to that shown in Figure 4.

4. Connect the printer's power cord to an AC outlet. DO NOT turn on the power at this time.

COMMODORE 2001 SERIES
COMPUTER

PET-TO-IEEE CABLE
(P/N 320101)

COMMODORE PRINTER
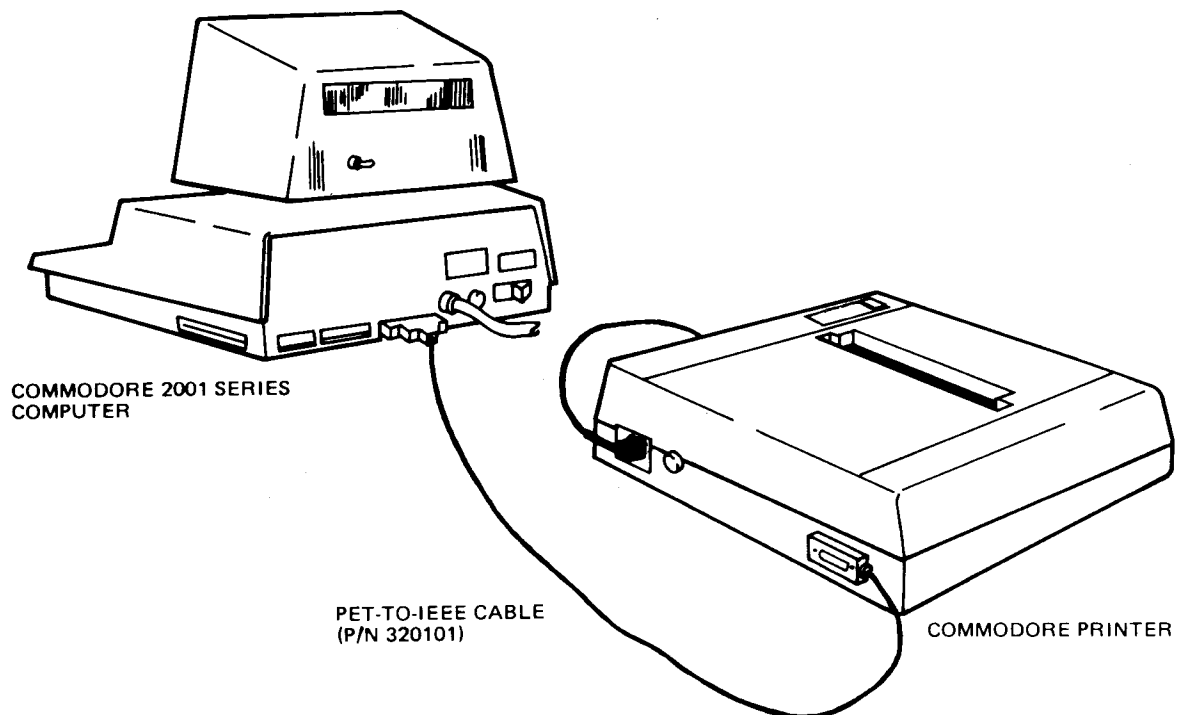
Figure 3. Printer to Computer Hookup

COMMODORE 2001 SERIES COMPUTER

PET-TO-IEEE CABLE
(P/N 320101)

DUAL DRIVE FLOPPY DISK,
MODEL 2040

IEEE-TO-IEEE CABLE
(P/N 905080)

COMMODORE CBM PRINTER,
MODEL 4022
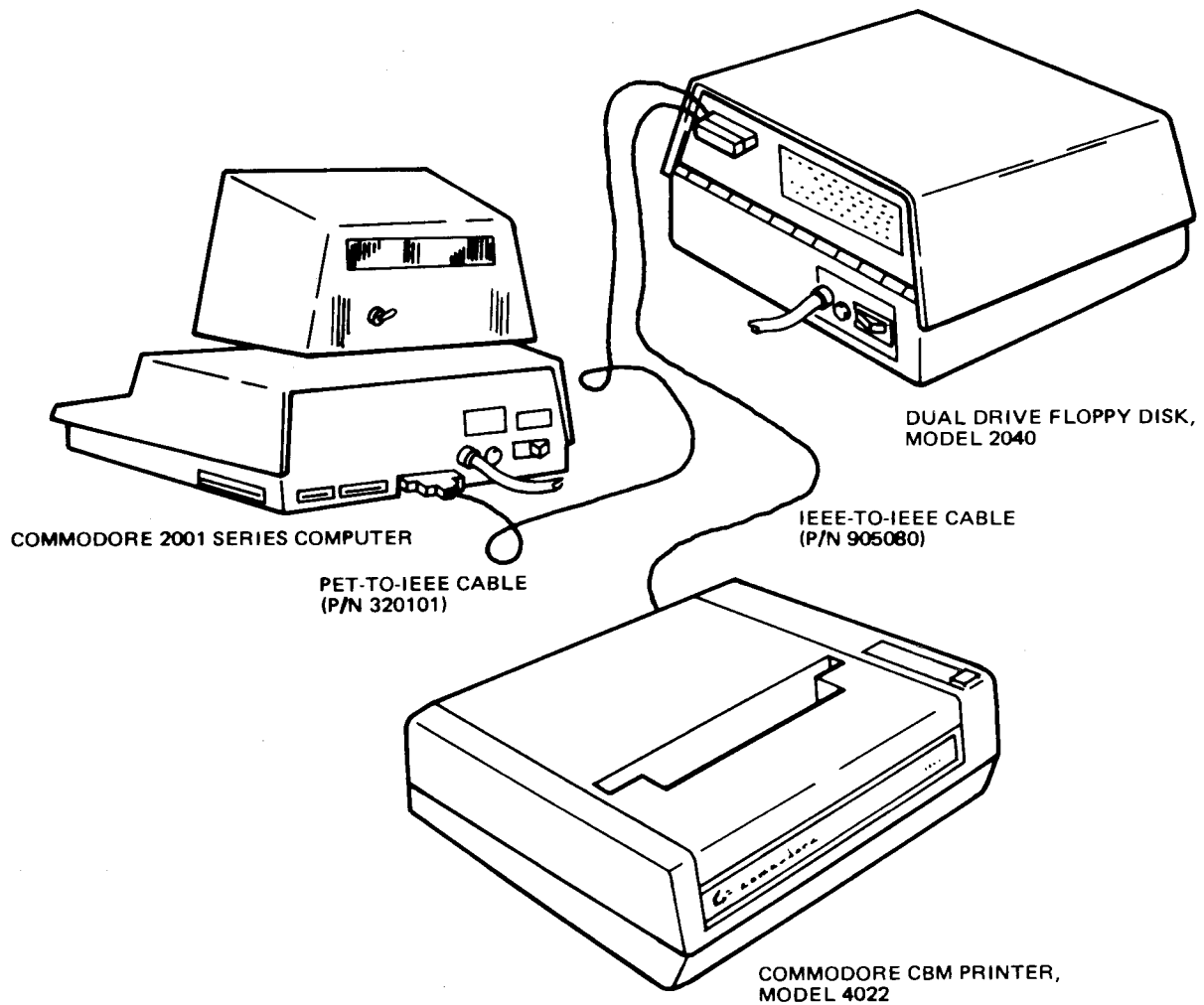
Figure 4. Multiple Hookup

PERFORMING THE POWER-ON TEST

You are now ready to proceed with the power-on part of the checkout:

1. Turn on the AC power to your computer and verify that it is working correctly.

2. To turn on the power to the printer, press the rocker switch at the rear of the printer so that the white dot is visible. In response to the application of AC power, the printer's microprocessor should move the print head all the way to the right, then to its home position at the far left. If this does not happen (and that's highly unlikely), turn off both machines, check all connections and try again. If you still get no response, contact your Commodore dealer.

INSTALLING THE PAPER

Now that you have gotten a positive response to all the previous steps, you can load the paper. See Figures 5 and 6.

PAPER ENTRY

TRACTOR FEED
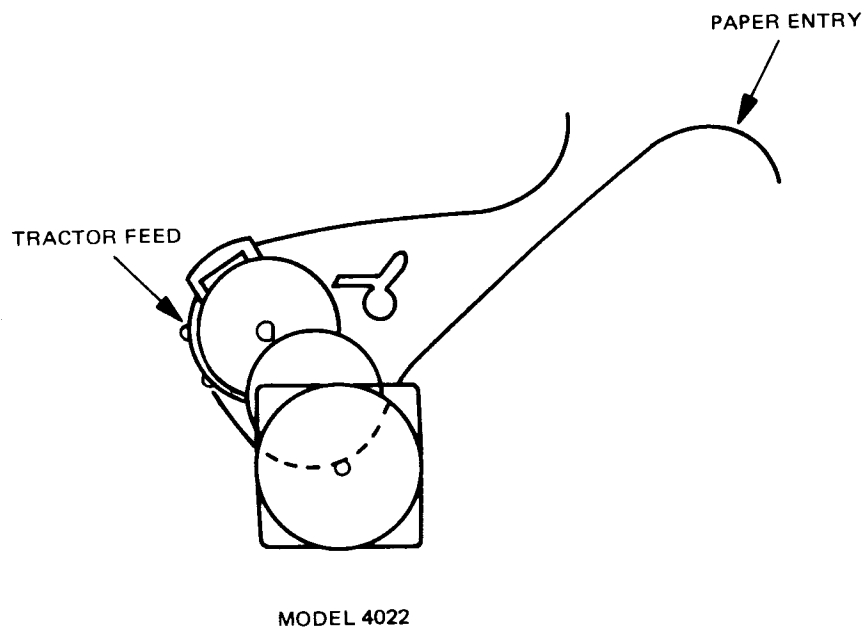
MODEL 4022

Figure 5. Paper Path

Loading Model 4022

The tractor feed printer accepts standard fan-folded pin feed paper. You can purchase this paper from your local Commodore dealer.

The paper drive in this printer may be adjusted to accept varying widths of paper. This is done by lifting the lever on each pin feed mechanism and sliding it to the desired position. Lock the levers in the rearward position when the adjustment is complete. Depress the paper feed button in the upper right-hand top panel to advance the paper after it is loaded. To remove the paper, open the feed mechanism retainers and pull the paper out.

Figure 6. Inserting Paper Into Model 4022

PERFORMING THE PRINT HEAD TEST

You can test the print head (and the ribbon cartridge as well) after you have inserted the paper. NEVER allow any printing to occur when there is no paper in the printer. To do so may result in damage to the print head. To perform this test, simply turn off the printer, then turn it back on while pressing the paper feed button. It is necessary to keep your finger on the feed button until the initial pass of the head is completed. This first pass of the head is used by the internal software to ensure that the head is able to move the entire cycle.

```
 !#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]↑←
!#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]↑←
#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]↑←
$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]↑←
%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]↑←
&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]↑←
'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]↑←
()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]↑←
)*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]↑←
*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]↑←
+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]↑←
,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]↑←
-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]↑←
```

You know how to connect your printer to your computer, how to check it out and how to insert the ribbon and paper. You are now ready for the next step: putting your Commodore printer to work.

Section 3

USING YOUR PRINTER

Your printer does much more than just give you clean, fast copy. Since it has its own internal microprocessor system, it is very versatile. In this section you will learn how to use your printer to print out listings, program results, and graphic displays. You will learn how to use it to format data and enhance character size. You will even be able to create your own custom character.

Before you attempt to use your printer, make sure you know how to do the following:

1. Operate your Commodore computer.

2. Do elementary programming in BASIC.

3. Write files to and from a peripheral device such as the tape cassette recorder or the Model 2040 Floppy Disk Drive.

4. Open and close files.

You should refer to your computer User Manual for this information.

This section uses certain conventions to indicate certain actions or requirements:

Example	Description
dn	Italicized lowercase letters in a syntax indicate that you should enter something (a variable) in place of the letters.
[ ]	Brackets indicates optional usage.

NOTE: The commands described in this manual apply specifically to use with the printer. Certain of the commands may follow a slightly different general form or produce different results from those described here when they are used for the computer or other peripheral devices. Consult the appropriate manual for the exact usage of these commands in other contexts.

SPECIAL PRINTER-ASSOCIATED COMMANDS

When you want to print something on your printer, essentially what you are doing is transferring the video screen function to the printer. A few special BASIC commands allow you to make this transfer. Most other BASIC considerations and rules remain the same. Always remember to press the [RETURN] key after you type each command.

The OPEN Command

The syntax of the OPEN command is:

OPEN *lfn, dn, [sa]*

This command sets a correspondence between a file number and a physical device. The *lfn* or *logical file number* may be any number you choose to assign to your file from 1 to 255. It doesn't matter which number you choose as long as you remain consistent throughout your set of commands. The *dn* or *device number* (also known as the primary address) refers to the device to which you wish to send the file. In the case of the printer, the number must be 4 since that is the number assigned to it at the factory.

> NOTE: If you should want to change a printer number, you can have it changed by an authorized Commodore technician. See your Commodore dealer about this. (You might want to have a printer's device number changed if you have two printers attached to your computer. You need to refer to each of them individually; hence, the need to change one of the device numbers.)

The *sa* or *secondary address* is somewhat of a unique concept. It alerts the printer's microprocessor system that formatting is to occur. Secondary addresses are thoroughly discussed starting on page 21, under FORMAT CONTROL.

The CMD Command

The syntax of this command is:

CMD *lfn*

CMD transfers control from the computer to the printer. The *lfn* must be the same as that in the OPEN statement with which it is associated. Unlike a PRINT command, the line or bus to the receiving device is left open. The line or bus to the receiving device (in this case, the printer) is said to be "listening." When you give the CMD command, the printer prints READY and the red LED in the paper feed button lights up to indicate the printer is waiting for further commands. If you follow a CMD command with a PRINT or a LIST command, the output is directed to the printer.

The PRINT# Command

The syntax of the PRINT# command is:

PRINT# *lfn, data*

PRINT# works just like PRINT except that it directs output to the printer instead of the video screen. The line to the printer is closed after printing the designated data. The line or bus to the printer is said to be "unlistened." Therefore, if you have used the CMD command, it is necessary to follow it with a PRINT# command in order to close down the connection between the printer and the computer.

NOTE: In standard Commodore BASIC, the PRINT command can be abbreviated as a question mark (?). You may not do this with PRINT#. It must always be typed out as PRINT#.

The CLOSE Command

The syntax of this command is:

CLOSE *lfn*

You should always close a file after printing from it. You may not exceed ten open files so it is well to make a habit of closing files when you are finished with them. This way you will always have the maximum number of files available for use. As you shall see later, one file may be open under several logical file numbers at one time.

Remember, since CMD does not close the line to the printer, you must always precede a CLOSE command with PRINT# in order to properly close the file.

Example:

Right		Wrong
OPEN 5, 4		
PRINT#5,"HELLO THERE"		
CLOSE 5		
OPEN 5, 4		OPEN 5,4
CMD 5,"HELLO THERE"	*not*	CMD 5, "HELLO THERE"
PRINT#5:CLOSE 5		CLOSE 5

```
OPEN 5,4                              OPEN 5,4
CMD 5,"HELLO THERE"        not     CMD 5,"HELLO THERE"
PRINT#5,"HELLO THERE"              PRINT#5,"HELLO THERE"
CLOSE 5                            PRINT#5:CLOSE 5


OPEN 5, 4                             OPEN 5, 4
PRINT#5,"HELLO THERE"      not     PRINT#5,"HELLO THERE"
CMD5,"HELLO THERE"                 CMD5,"HELLO THERE"
PRINT#5:CLOSE 5                    CLOSE 5
```

Armed with these thumbnail descriptions of what the printer-related commands do, you can now proceed to the next part of this section, which tells you how to use these commands to control your printer. You can find more detailed information about these commands in your computer User Manual.

PRINTING IN THE DIRECT MODE

The direct mode of communication allows you to enter printing commands at the keyboard.

The following example shows the entire direct mode listing process of a short BASIC program. A file containing a single BASIC statement is typed into the computer's memory. The file is opened, the output channel is opened, and the file is listed. After the file is listed, the output channel is closed and the file is closed. The printer is now out of use and the computer is ready to accept new commands.

Example:

You type:	The screen displays:	The printer prints:	Comments:
10?"THIS IS A TEST"	10?"THIS IS A TEST"		You place the data in the computer's memory
OPEN 3,4	OPEN 3,4 READY.		You open the file and give it a *lfn* of 3. The 4 makes the file available to the printer.
CMD 3	CMD 3	READY.	The printer is "listening."
LIST	LIST	10 PRINT"THIS IS A TEST" READY.	The program is printed on the printer. The printer is still "listening."
PRINT#3	PRINT#3 READY.		Use the PRINT# command to "unlisten" the printer.
CLOSE 3	CLOSE 3 READY.		You close the file so that *lfn* 3 can be used for something else.

PRINTING UNDER PROGRAM CONTROL

As we have seen, you can control the printer directly from the keyboard. You can also control the printer from within a BASIC program. In the example below, this short BASIC program is placed in the computer's memory. (It could have been placed there from the keyboard, a cassette tape, or a floppy disk):

```
10 OPEN 3, 4
20 CMD 3
30 PRINT"THIS PROGRAM ILLUSTRATES PROGRAM CONTROL"
40 LIST
```

> REMEMBER THIS: The LIST command within a program terminates program execution. Then, when you are finished running a program, you must type the PRINT# command to close the channel. Then type the CLOSE command to close the file.

The RUN command is given and the resulting printout is shown here:

```
THIS PROGRAM ILLUSTRATES PROGRAM CONTROL

10 OPEN 3,4
20 CMD 3
30 PRINT"THIS PROGRAM ILLUSTRATES PROGRAM CONTROL"
40 LIST
READY.
```

A more sophisticated example shows a BASIC program that takes advantage of some of your printer's special features. The program directs the printer to generate the following:

- An enhanced heading. You will find more about this feature on page 32.

- The entire printer character set.

- A listing of the program.

Two special character functions are used in the program: the OFF/RVS key and the CRSR Down (Cursor Down) key. You may find out more about these starting on page 33.

A printout of a program you place in your computer's memory looks like this:

```
10 OPEN 4,4
20 FOR I=32 TO 95   :A$=A$+CHR$(I):NEXT
30 FOR I=160 TO 223:B$=B$+CHR$(I):NEXT
40 C$=" "+A$
50 D$=" "+B$
60 E$=" "+A$
70 F$=" "+B$
80 G$=" "+C$
90 H$=" "+D$
100 PRINT#4,CHR$(1)"CBM 4022 PRINTER CHARACTER SET"
110 PRINT#4:PRINT#4:PRINT#4
120 PRINT#4,A$
130 PRINT#4,B$
140 PRINT#4,C$
150 PRINT#4,D$
160 PRINT#4,E$
170 PRINT#4,F$
180 PRINT#4,G$
190 PRINT#4,H$
200 CMD4
210 PRINT:PRINT:PRINT:PRINT:PRINT
220 LIST
```

Here is an explanation of each statement in the program.

Line No.	Explanation
10	Opens the printer as a file for printing
20	Sets A$ equal to a string containing all the unshifted characters from the keyboard
30	Sets B$ equal to a string containing all the shifted characters from the keyboard
40	Sets C$ equal to the set of all unshifted characters printed in reverse field
50	Sets D$ equal to the set of all shifted characters in reverse field
60-90	Each time a PRINT statement is encountered as in lines 120-190, a Carriage Return is executed (unless a semicolon is placed at the end of the preceding PRINT statement). Therefore, each string as in lines 160-190 must begin with a Cursor Down to reset the printer to lowercase.

Printing a Cursor Down has the same effect on the printer as a POKE 59468,14 has to the computer's screen in that it prints the characters following the Cursor Down in lowercase. However, after a Cursor Up or Carriage Return is encountered, the printer will resume printing in uppercase and graphics.

60	Sets E$ = to a Cursor Down plus A$
70	Sets F$ + to a Cursor Down plus B$
80	Sets G$ = to a Cursor Down plus C$
90	Sets H$ = to a Cursor Down plus D$
100-190	PRINT #4 indicates to the computer to print to device 4 (in this case, the printer)
100	CHR$(1) increases the width of the printed character to twice its original size. This line prints the title of the program in 'enhanced characters.'
110	Prints three Carriage Returns on the printer
120-190	Prints each of the strings constructed in lines 20 through 90 on the printer
200	CMD4 transfers all commands which would normally be displayed on the screen to the printer. However, if this mode is used, care must be taken to close the file to assure proper functioning of the computer after the printout is completed.
210	Prints 5 Carriage Returns on the printer.
220	Prints out a listing of the program.

To execute this program, simply type:

RUN

This is the result:

```
cBM 4022 PRINTER CHARACTER SET


 !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]↑←
```

(character set illustration)

```
10 OPEN 4,4
20 FOR I=32 TO 95   :A$=A$+CHR$(I):NEXT
30 FOR I=160 TO 223:B$=B$+CHR$(I):NEXT
40 C$="  "+A$
50 D$="  "+B$
60 E$="  "+A$
70 F$="  "+B$
80 G$="  "+C$
90 H$="  "+D$
100 PRINT#4,CHR$(1)"CBM 4022 PRINTER CHARACTER SET"
110 PRINT#4:PRINT#4:PRINT#4
120 PRINT#4,A$
130 PRINT#4,B$
140 PRINT#4,C$
150 PRINT#4,D$
160 PRINT#4,E$
170 PRINT#4,F$
180 PRINT#4,G$
190 PRINT#4,H$
200 CMD4
210 PRINT:PRINT:PRINT:PRINT:PRINT
220 LIST
READY.
```

Now, to close the channel and the file, type:

PRINT#4
CLOSE 4

FORMAT CONTROL

Through the printer's format control option, you can control the interpretation of data sent to the printer. The format control option allows you to print numbers in columns, set the number of lines per page, and perform other useful formatting tasks. To implement the format control option, you use the third parameter of the OPEN command.

OPEN *lfn, dn, sa*

As before, *lfn* is the *logical file number,* *dn* is the *device number* (also known as the primary address), and *sa* is the *secondary address.* It is in the secondary address position that you enter the format control parameter.

You can enter any of the seven secondary address in your OPEN command. These are:

0   Print data exactly as received

1   Print data according to a previously-defined format

2   Store the formatting data

3   Set the number of lines per page to be printed

4   Enable the printer format diagnostic messages

5   Define a programmable character

6   Set spacing between lines (for Model 2022 only)

After the appropriate OPEN statement has been transmitted, a PRINT# statement is required to transmit the secondary address information to the specified device (in this case, your printer).

Remember, that it is possible to have as many as ten files open simultaneously. This allows you to perform several formatting functions at one time on the data in the computer's memory. There are examples of the formatting capabilities described in this section in the Appendix.

Printing Data Exactly As Received: *sa*=0

This secondary address is the default value. Whether or not you include it in your OPEN statement, the printer prints data exactly as received. Up to 80 characters are printed on each line and if the next character is not a Carriage Return, a Carriage Return is performed automatically and the overflow characters are printed on the next line.

Example:

```
OPEN 5,4
PRINT#5,"THIS IS A TEST!"
```

Results in:

THIS IS A TEST!

Printing Data According to a Previously Defined Format: sa=1

A secondary address of 1 invokes the formatting features of your printer. The data to be printed is arrayed according to a previously specified format using **sa=2**. If you should transmit a string of data when **sa=1** is in effect and there is no formatting data in the printer's memory, then the data string is printed exactly as it is received.

When formatting string data from the computer, a skip, **CHR\$(29)**, must be sent to delimit the end of a string being edited to a field. Leading blanks are stripped off a string; therefore, to print a blank alpha field you must transmit a shifted blank, **CHR\$(160)**. The alpha field is then right padded with blanks as shown below.

Example:

```
PRINT#1,"AAA      AAA      AAA"
PRINT#2,"ABC"      CHR$(29)      CHR$(160) CHR$(29) "DEF"
```

Results in:

ABC   · DEF

Storing the Formatting Data: *sa*=2

Perhaps the most significant feature of your printer is its ability to format data. Picture formatting generates a simple one-to-one correspondence between the column position of the print line and the symbol that is to appear there. Formatting allows you to left- or right-justify columns of data, or to align numeric data on its decimal point position.

Example:

```
OPEN 2, 4, 2
OPEN 1, 4, 1
PRINT#2, "$$$.99"
PRINT#1, .05
```

Results in:

$.05

The commands in the example above transfer formatting data to the printer's internal memory for future print editing use. Once completed, **READY** and the blinking cursor appear on the screen.

The Formatting Characters

Specification of a format is accomplished by a set of formatting characters which, when joined, define the rules of formatting data for the printer format interpreter. The formatting characters may be grouped into three classes:

Numeric	9,Z,$,S, . , -
Alpha	A
Skip	(Blank)

Fields are specified by combining these formatting characters. Up to 80 characters are accepted in a format string. Excess (overflow) characters are printed onto the next print line. Figure 5 contains formatting examples.

Numeric:

9 — Specifies a digit position in a numeric field. If there is no digit to print in this position, then a blank is substituted.

Z — Also specifies a digit position in a numeric field. Unlike the 9, this character forces a 0 to be printed if there is no digit available for this position. This is desirable if leading zeroes are required in a numeric field.

$ — If one $ is specified, then the field is treated as a dollar amount with a fixed dollar sign position:

$123
$1234.00

And so on. If all digit positions to the left of a decimal point are $, then the number is printed as a floating dollar sign right justified before the most significant digit.

S — When preceding a numeric field, the sign of the number (+ or -) is to be printed in this fixed column position.

. — Defines the position of the decimal point and is printed literally in this position.

- — Specifies a trailing sign. If the number is +, a blank is printed. A numeric field cannot have both S and -. When this is the case, only the S will be honored.

Here are some examples of format fields:

a) Field width and decimal position specified —
   99   Two-digit integer with no sign
   99.999 Five-digit real type with 3 positions to right of decimal

b) Dollar sign insertion —
   $99.99 Fixed dollar sign
   $$$.99 Floating dollar sign

c) Leading zeros forced —
   ZZZZ   No decimal
   ZZ.99 Decimal

d) Trailing minus —
   999-   Blank is printed, if number is positive

e) Signed numbers —
   S99   + or - always printed

Figure 5 contains more examples of sample data, format fields, and the resultant editing.

FORMAT FIELD	DATA	EDITED RESULT
AAAAA	ABC	ABC
AAAAA	ABCDEFG	ABCDE
$$$$$	99	$99
$9999	99	$ 99
$99.99	77	$77.00
$99.99	-77	$77.00
$99.99-	-77	$77.00-
$99.99-	77	$77.00
S$99.99	77	+$77.00
ZZZZ	77	0077
ZZ.999	77	77.000
ZZZ.99	77	077.00
.999.99	77	77.00
.99	77	.**
.99	.001	.00
S.999	1.5E-02	+.015
Z.999-	1.5E-02	0.015
Z.999-	-1.5E-02	0.015-

Figure 8. Formatting Examples

The printer software handles up to ten significant figures, and an exponent range ± 99 on numbers passed as data to be formatted. Exponential numbers must be normalized such that $0 \leq |x| < 10$.

Alpha (String Data):

The letter A represents one position of an alpha field. Within the field, leading blanks are truncated, the field is left-justified, and it is padded to the right with blanks. Shifted blanks, **CHR$(160)**, are not deleted if in the leading positions.

Example:

```
PRINT#1,"A  AA  AAA"
PRINT#2, "CBM" CHR$(29)"CBM" CHR$(29)"CBM"
```

Results:

```
C  CB  CBM
```

Skip (Blank):

Simply use blanks where spaces are required:

Example:

```
PRINT#1,"AAA  AAA      AAA"
PRINT#2,"PET" CHR$(29) "PET"CHR$(29) "PET"
```

Results in:

```
PET PET     PET
```

To right justify integers, define an alpha string the length of the longest integer. Then blank fill the left of the integer data string and right justify the number within that string.

Literals in Format Strings

Literals are characters that are to be printed exactly as they exist in the printer's memory, rather than being used to format other data. Literals are flagged by preceding each literal in the format string with a reverse field **on** character.

One use of literals is to create a special form at the same time you are printing data. This is done by using the special graphics characters to form vertical, horizontal, or other lines between the fields. You can create a form while printing data, either by overprinting graphic characters or by inserting literal characters in a format string as shown below.

The only limitation on literal characters in format strings is that they cannot be printed reverse-field. They are also restricted to the same character set in which data are currently being printed.

Example:

    PRINT#2, "▓I AAA▓ "

Results in:

    I AAAI

Setting the Number of Lines Per Page: sa=3

This special secondary address allows you to vary the number of printed lines per page. In order for this paging option to take effect, you must turn paging on with the special paging character described on page 33. When paging is on, and the paging secondary address is not implemented, the default number of lines per page is 66 including three blank lines at the top of the page and three blank lines at the bottom of the page.

Example:

    OPEN 8,4,3
    PRINT#8,20

Enabling the Printer Diagnostic Messages: *sa*=4

When a secondary address of 4 is transmitted, the printer's diagnostic messages appear whenever an error occurs. When a formatting error occurs, a message is printed, the format is dumped, and a pointer is located at the offending field. If a secondary address of 4 has NOT been transmitted, then once an error occurs, data characters are dumped directly to the print line even if the last command was to format data.

To enable the printer diagnostic messages, execute the following program steps:

Example:

```
10 OPEN 4,4,4
20 PRINT# 4
30 CLOSE4
```

Whether or not diagnostic messages are enabled, overflow of numeric fields is always indicated by fields filled with asterisks (******). This is a nonfatal error; as in other cases, the secondary address option is reset to zero, and all data retrieved by the printer are printed exactly as they are received.

Diagnostic Messages

PE:L — Lines per page out of range. An attempt was made to set the lines per page to a value outside the range <13< linesperpage <128 via a print to SA 3. The command is ignored and the previous length remains in force.

PE:C — Bad command. You gave the printer an invalid SA outside the range. < SA <11. The command is ignored.

PE:M — Data-format mismatch. You gave the printer non-numeric data to print in a numeric field. The first character printed after the error message is the offending character.

PE:E — Exponent error. The numeric data given to the printer to print in a numeric field had an invalid exponent. The expected form of a number in exponential form is: n.nnnn+ee or n.nnnn−ee. The exponent must contain a plus or minus sign with a two digit exponent.

PE:F — Bad format. The data, that was sent to SA 2 to store, either contains illegal formatting characters, or was of an unrecognizable syntax.

PE:T — Terminator error. The SA changed before a terminator was detected. A terminator is defined as a carriage return (chr$(13)), a line feed (chr$(10)), a carriage stand (chr$(141)), or a carriage return line-feed sequence. Communication with the current SA must be terminated with a terminator before attempting to "talk" to another SA.

Defining a Programmable Character: *sa*=5

A secondary address of 5 allows you to create a custom character of your own. This programmable character is initialized with this secondary address.

Suppose you wanted to program the special symbol in Commodore Business Machines' logo, which is:

C⋶ commodore

Lay out a 7x6 matrix (the same matrix as is on the print head). To the left of the matrix, write the binary bit value of each line. Use dots, one per square to create your character. Then add up the binary bits indicated by your dots in each column. These totals are used in the DATA statement in your program.

Example:

64			●	●		
32		●			●	●
16	●				●	
8	●					
4	●				●	
2		●			●	●
1			●	●		
	28	34	65	65	54	34

◄─── TOTALS

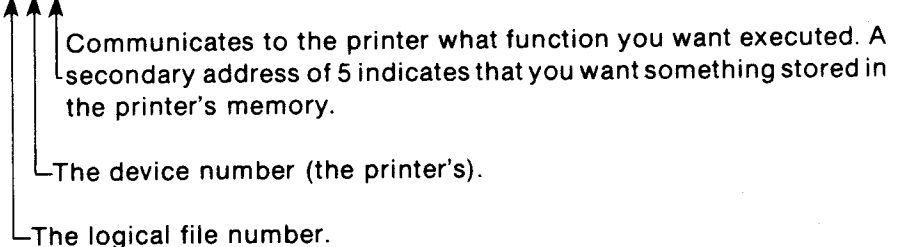The DATA statement in your program will read:

DATA 28,34,65,65,54,34

The program shown in the printout in the next example writes the Commodore logo ten times. It creates a string with the CHR$ value of the column totals and passes the string to the printer with *sa*=5. To achieve upper- and lowercase characters, use the CRSR Up (Cursor Up) for uppercase characters, and CRSR Down (Cursor Down) for lowercase characters.

Example:

```
10 DATA28,34,65,65,54,34
20 OPEN5,4,5
30 FORI=1TO6:READA:A$=A$+CHR$(A):NEXT
40 PRINT#5,A$
50 OPEN4,4
60 FORI=1TO10
70 PRINT#4,CHR$(1)CHR$(254)" COMMODORE BUSINESS MACHINES"
80 NEXT
90 CLOSE5
100 CLOSE4
```

The following is a description of the program shown above.

Line No.

20   Open 5,4,5

Communicates to the printer what function you want executed. A secondary address of 5 indicates that you want something stored in the printer's memory.

The device number (the printer's).

The logical file number.

30   The FOR NEXT loop contained in this line reads the data contained in line 10 and constructs a string A$. A$ contains the necessary information to print the Commodore logo.

40   Stores A$ in printer memory.

50   Opens the print file for printing.

60-80   Prints the Commodore logo and "Commodore Business Machines" ten times.

PRINT#4	Prints to the printer
CHR$(1)	Prints 'enhanced characters'
CHR$(254)	Prints the programmed character (the Commodore logo)
Cursor Up	Sets the printer to uppercase
Cursor Down	Sets the printer to lowercase

90-100   Closes the two files opened by the program.

After typing RUN, you get this result:

```
C Commodore Business Machines
C Commodore Business Machines
C Commodore Business Machines
C Commodore Business Machines
C Commodore Business Machines
C Commodore Business Machines
C Commodore Business Machines
C Commodore Business Machines
C Commodore Business Machines
C Commodore Business Machines
```

NOTE

Multiple programmable characters in the same line can only be made by overprinting. The programmable character cannot be changed when a line wraps around to the next line.

Setting Spacing Between Lines : sa=6

A secondary address assignment of 6 controls the number of steps between successive lines of print. There are 144 steps per inch, so a declared value of 18 produces eight lines per inch.

Example:

OPEN 6, 4, 6
PRINT#6, CHR$(18)

produces eight lines per inch, while

PRINT#6,CHR$(144)

produces lines spaced one inch apart.

The default value is 24, which produced the standard six lines per inch.

Selecting Upper or Lower Case: sa=7

To change the computer and the printer to all upper-case characters, you must change location 59468 to 12 and condition the printer as indicated by the following program:

Example:

10 POKE 59468,12
20 OPEN 7,4,7
30 PRINT#7
40 CLOSE7

Selecting ASCII/Graphics: sa=8

To change the computer and printer back to ASCII/graphics, you must restore location 59468 to 14 and restore the printer back to its original condition as indicated by the following program:

Example:

```
10  POKE 59468,14
20  OPEN 8,4,8
30  PRINT#8
40  CLOSE8
```

Suppress Diagnostic Message Printing: sa=9

To turn off diagnostic message printing, execute the following program:

Example:

```
10  OPEN 9,4,9
20  PRINT#9
30  CLOSE9
```

Resetting the Printer: sa=10

To reset the printer, execute the following program:

Example:

```
10  OPEN 10,4,10
20  PRINT#10
30  CLOSE10
```

SPECIAL CHARACTER FUNCTIONS

Special control characters can be used to change the mode of printing within a single line. Table 2 contains a summary of the special control characters. The table is followed by a brief description of each function.

Enhanced Characters

You can print any character double its width. While characters are normally printed in a seven-row by six-column matrix, enhanced characters appear in a seven-row by twelve-column matrix.

The example below shows how already enhanced characters can be enhanced.

Example:

```
10 OPEN4,4
20 PRINT#4,"H"CHR$(1)"E"CHR$(1)"L"CHR$(1)"L"CHR$(1)"0"
```

Results in:

HELLO

Paging

Paging must be turned on in your program by means of the code as keyboard entry shown in Table 2. Otherwise, printing is continuous. The paging function provides for 66 lines per page including three blank lines at the top of the page and three blank lines at the bottom. The number of lines per page may be altered by using the secondary address 3 option. When paging is in effect, the paging off character performs a top-of-form function.

Specifying Individual Upper- and Lowercase Characters

The printer default character set is that of the CBM 2001 when power is first applied to the printer. Though it is not possible to display both graphics and lowercase simultaneously on the video display screen, the printer does allow shifting between character sets on the same line. This function is analogous to the letters/figures shift on a Teletype™.

Example:

```
10 OPEN4,4
20 PRINT#4,"COMMODORE"
30 CMD 4
40 LIST
```
Results in:

Commodore

TABLE 2

Special Control Character Summary

Printer function	Code	ASCII	Keyboard
Enhance	CHR$(1)	SOH	NA
Unenhanced	CHR$(129)		NA
Paging on/reset	CHR$(147)		SHIFT & CLR HOME
Paging off	CHR$(19)	DC3	CLR HOME
RVS ON	CHR$(18)	DC2	OFF RVS
RVS OFF	CHR$(146)		SHIFT & OFF RVS
Carriage return	CHR$(13)	CR	RETURN
Carriage return with no line feed	CHR$(141)		NA
Line feed	CHR$(10)	LF	NA
Uppercase	CHR$(145)		CRSR Cursor Up
Lowercase	CHR$(17)		CRSR Cursor Down
Skip space	CHR$(29)		CRSR Cursor Right
Quote	CHR$(34)	"	" Quote

Reversing a Field

This function inverts the dot matrix to produce the effect of white on black. DO NOT use this mode of printing for more than five consecutive lines since extended printing in this mode will damage the print head.

Example:

```
10 OPEN5,4
20 PRINT#5,"     COMMODORE"
30 CMD 5
40 LIST
```
Results in:

The Carriage Return

If you attempt to print more than 80 characters on a line, a Carriage Return with Line Feed will be forced and the overflow characters will be printed on the next line. If you transmit a Shift Carriage Return ($8D), a Carriage Return without Line Feed is executed. This allows overprinting on a line. A Carriage Return turns off reverse field, character enhance and the quote mode.

Quotation Marks

If an odd number of quotation marks have been transmitted, control characters are made visible. This can be particularly useful when you are making a listing of a BASIC program containing cursor control characters in quotation marks.

APPENDIX

The programs in this appendix illustrate the many formatting capabilities of your printer. The first program is completely annotated. In the following programs, new concepts are commented upon.

SIMPLE STRING FORMATTING

This program opens four files to perform four different tasks. It prints three lines: the format string, the unformatted data string, and the formatted data string. It then lists the program.

Here is a printout from the program:

```
AA AA AA AA
ABC
AB AB AB AB
```

```
10 OPEN1,4
20 OPEN2,4,1
30 OPEN3,4,2
40 OPEN4,4,4
50 PRINT#4     :    REM ENABLE ERROR DIAGNOSTICS
60 A$="AA AA AA AA"   : REM FORMAT STRING
70 B$="ABC"           : REM STRING TO BE FORMATTED
80 PRINT#1,A$
90 PRINT#1,B$
100 PRINT#3,A$
110 C$=CHR$(29)
120 PRINT#2,C$B$C$B$C$B$C$B$
130 CMD1
140 PRINT:PRINT:PRINT:PRINT
150 LIST
READY.
```

An analysis of the program will help you understand how it works.

Line No.	Explanation

10-40   Four logical files are opened to the printer so that the data in the computer's memory can be used in four different ways:

Logical File No.	Secondary Address	Usage
1	0 (default)	Print data exactly as transmitted.
2	1	Format data before it is printed.
3	2	Transmit the format string.
4	4	Enable format error diagnostic messages to be printed (if any).

Line No.	Explanation
50	Prints error messages (if any)
60	Defines the format string
70	Defines the data string
80	Prints the format string
90	Prints the unformatted data string
100	Transmits the format string to the printer's memory
110	Skip a space
120	Prints the formatted data string
130	Transfers control from the computer to the printer
140	Prints four blank lines
150	Lists the program

LEADING ZEROES FORCED

This program prints ten rows of numbers, each row beginning with a number one higher than in the preceding row. In line 60, the formatting statement (which is prepared for in line 30, and transmitted in line 70) specifies leading zeroes in all fields and plus signs in the first two fields.

```
+0001.    +0002.    0003.    0004
+0002.    +0003.    0004.    0005
+0003.    +0004.    0005.    0006
+0004.    +0005.    0006.    0007
+0005.    +0006.    0007.    0008
+0006.    +0007.    0008.    0009
+0007.    +0008.    0009.    0010
+0008.    +0009.    0010.    0011
+0009.    +0010.    0011.    0012
+0010.    +0011.    0012.    0013
```

```
10 OPEN1,4
20 OPEN2,4,1
30 OPEN3,4,2
40 OPEN4,4,4
50 PRINT#4   :   REM ENABLE ERROR DIAGNOSTICS
60 F$="SZZZZ.  SZZZZ.  ZZZZ.  ZZZZ"
70 PRINT#3,F$
80 FORI=1TO10
90 PRINT#2,I,I+1,I+2,I+3
100 NEXT
110 CMD1
120 PRINT:PRINT:PRINT:PRINT
130 LIST
READY.
```

SIGNED NUMBERS

As shown below, a trailing sign, when specified in a format statement, either prints a blank for a positive number or a minus (-) for a negative number. This form is sometimes desired by accountants.

A leading sign always prints a plus (+) or a minus (-) for the sign of the number.

If no sign is specified in the format, then no sign is printed. This works essentially like an absolute value on the number.

```
1   -  1    1-
```

```
10 OPEN1,4
20 OPEN2,4,1
30 OPEN3,4,2
40 OPEN4,4,4
50 PRINT#4  :  REM ENABLE ERROR DIAGNOSTICS
60 F$="999  S999  999-"
70 PRINT#3,F$
80 PRINT#2,-1,-1,-1
90 CMD1
100 PRINT:PRINT:PRINT:PRINT
110 LIST
READY.
```

DECIMAL FRACTION SPECIFICATION

When the right hand side of the decimal point in a format statement has more digits of precision specified than contained in the formatted data, the remaining digit positions are filled with zeroes. If the converse is true, then the formatted number is truncated.

```
 5.00  -  1.25000    3
10.00  -  2.50000    6
15.00  -  3.75000    9
20.00  -  5.00000   12
25.00  -  6.25000   15
30.00  -  7.50000   18
35.00  -  8.75000   21
40.00  - 10.00000   24
45.00  - 11.25000   27
50.00  - 12.50000   30
```

```
10 OPEN1,4
20 OPEN2,4,1
30 OPEN3,4,2
40 OPEN4,4,4
50 PRINT#4  :   REM ENABLE ERROR DIAGNOSTICS
60 F$="999.99   S999.99999   999-"
70 PRINT#3,F$
80 FORI=1TO10
90 PRINT#2,-10*I/2,-5*I/4,3*I
100 NEXT
110 CMD1
120 PRINT:PRINT:PRINT:PRINT
130 LIST
READY.
```

FIXED AND FLOATING $ FIELDS

In fixed dollar fields the dollar sign ($) is printed in the leftmost position and leading unused digit positions are printed as blanks.

In floating dollar fields the $ is printed immediately to the left of the most significant figure or decimal point. Unused digit fields to the left of the $ are printed blank.

```
$64.08    $4.87    $ 7222    $.70
$24.02    $9.29    $ 2132    $.99
$17.72    $3.02    $ 7924    $.49
$30.78    $6.27    $ 3965    $.86
$14.68    $7.36    $ 5237    $.22
$68.74    $8.67    $ 5432    $.02
$ 8.67    $6.47    $ 6218    $.50
$33.82    $8.82    $ 8275    $.96
$63.43    $5.34    $ 9587    $.95
$29.13     $.67    $ 9359    $.51
```

```
10 OPEN1,4
20 OPEN2,4,1
30 OPEN3,4,2
40 OPEN4,4,4
50 PRINT#4  :  REM ENABLE ERROR DIAGNOSTICS
60 F$="$99.99  $$$.99  $99999  $.99"
70 PRINT#3,F$
80 FORI=1TO10
90 PRINT#2,RND(1)*100,RND(1)*10,RND(1)*10000,RND(1)
100 NEXT
110 CMD1
120 PRINT:PRINT:PRINT:PRINT
130 LIST
READY.
```

MIXED FORMAT FIELD TYPES

Shown below are integer with leading zeroes, alpha field, floating dollar and integer with zero suppressed.

In line 80, CHR$(29), (skip) is used as a terminator for the alpha field so that it is formatted correctly.


```
00002  PET COMPUTER             $795.00   50
```


```
10 OPEN1,4
20 OPEN2,4,1
30 OPEN3,4,2
40 OPEN4,4,4
50 PRINT#4   :   REM ENABLE ERROR DIAGNOSTICS
60 F$="ZZZZZ  AAAAAAAAAAAAAAAAAA  $$$$$.99   999"
70 PRINT#3,F$
80 A=2
90 B$="PET COMPUTER"
100 C=795
110 D=50
120 PRINT#2,A,B$,CHR$(29),C,D
130 CMD1
140 PRINT:PRINT:PRINT:PRINT
150 LIST
READY.
```

FORMATTING WITH LITERALS

In the example below, a literal appears at the end of the printout even though the format statement does not contain one in that position. Your printer automatically prints any literal that appears at the beginning of the format string at the beginning and end of the printout.

```
| 00 | $662 | 01.418 |
| 01 | $482 | 05.593 |
| 08 |  $47 | 02.790 |
| 09 | $657 | 04.847 |
| 05 | $781 | 02.316 |
| 09 | $425 | 06.210 |
| 01 |  $15 | 04.099 |
| 06 | $854 | 05.610 |
| 03 | $324 | 04.054 |
| 03 | $971 | 03.325 |
```

```
10 OPEN1,4
20 OPEN2,4,2
30 OPEN3,4,1
40 F$="#| ZZ #| $$$$ #| ZZ.999   "
50 PRINT#2,F$
60 FORI=1TO10:X=10*RND(1):Y=1000*RND(1):Z=8*RND(1)
70 PRINT#3,X;Y;Z:NEXT
80 CMD1:LIST
READY.
```

C= commodore