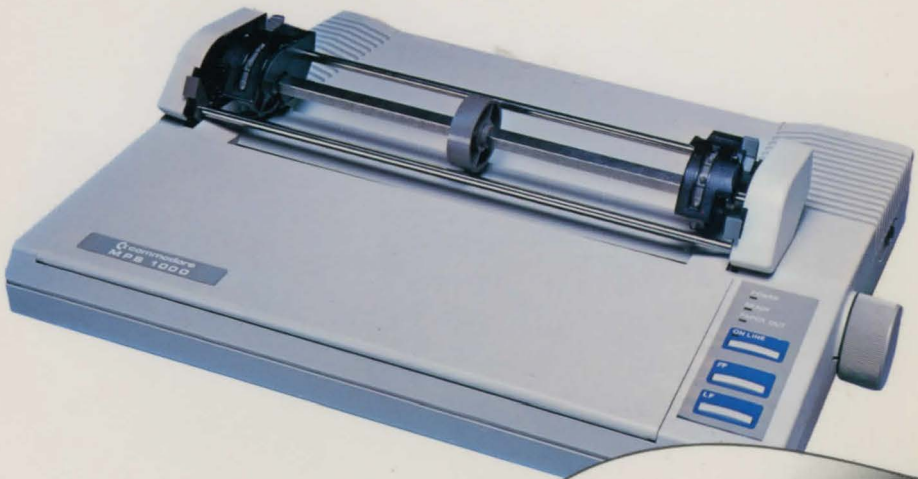


COMMODORE[®]

**mPS
1000**

DOT MATRIX PRINTER

USER'S MANUAL



mps 1000

DOT MATRIX PRINTER

USER'S MANUAL

INTRODUCTION 1

CHAPTER 1. Setting Up the Printer 3

 Unpacking your printer 4

 Getting your printer ready to use 4

 Using the printer 11

 The printer's controls 16

 Connecting the MPS 1000 to your computer 18

 The DIP switches 22

 Printer paper 24

CHAPTER 2. Printing 25

 The print head 25

 Bidirectional printing 26

 Changing pages 28

 New Letter Quality (NLQ) mode 28

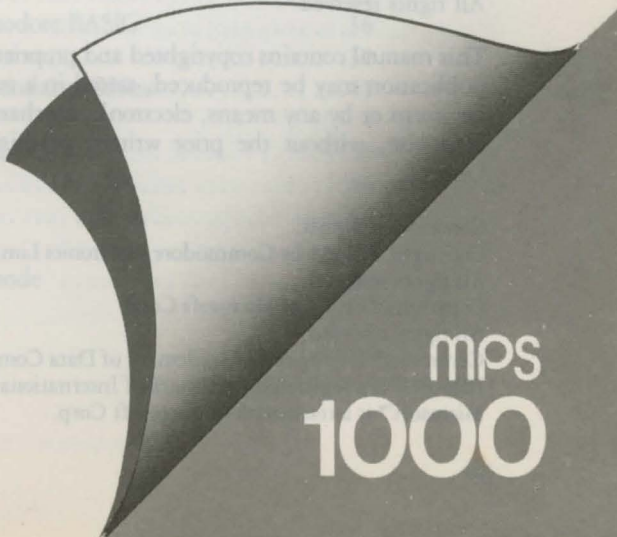
CHAPTER 3. Printer Control Codes 31

 ASCII and IBM Extension ASCII codes 31

 Escape codes and ESCape codes 34

 Programming languages 34

Copyright © 1987 by Commodore International, Inc.
Copyright © 1984 by Softek Epson Corporation. All rights reserved.
All other rights reserved.
This manual contains copyrighted and registered trademarks of the manufacturer of the printer. The printer is manufactured in Japan. The MPS 1000 is a registered trademark of Commodore International, Inc. All other trademarks are the property of their respective owners.
Commodore International, Inc.
15455 E. Alameda Avenue
Denver, Colorado 80231
Softek Epson Corporation
10000 W. Alameda Avenue
Denver, Colorado 80231



mps
1000

1000 MP DOT MATRIX PRINTER USER'S MANUAL

Copyright © 1985 by Commodore Electronics Limited
Copyright © 1984 by Seiko Epson Corporation, Nagano, Japan
All rights reserved

This manual contains copyrighted and proprietary information. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior written permission of Commodore Electronics Limited.

Commodore BASIC
Copyright © 1985 by Commodore Electronics Limited
All rights reserved
Copyright © 1977 by Microsoft Corp.
All rights reserved

Centronics® is a registered trademark of Data Computer Corporation.
IBM-PC® is a registered trademark of International Business Machines Corporation.
Microsoft™ is a trademark of Microsoft Corp.

TABLE OF CONTENTS

INTRODUCTION	1
CHAPTER 1, Setting Up Your MPS 1000 Printer	5
Unpacking your printer	5
Getting your printer ready to use	6
Using the tractor unit	11
The control panel	16
Connecting the MPS 1000 to your computer	18
The DIP switches	22
Printer maintenance	24
CHAPTER 2, Elements of Dot Matrix Printing	25
The print head	25
Bidirectional printing	26
Changing pitches	26
Near Letter Quality (NLQ) mode	28
CHAPTER 3, Printer Control Codes	31
PETASCII and IBM Extension ASCII codes	31
Secondary addresses and ESCape codes	33
Programming languages	34
CHAPTER 4, Commodore Print Mode	35
Demonstration programs	35
The MPS 1000 and Commodore BASIC	36
Control codes for printing text	37
Advanced control with secondary address	45
CHAPTER 5, IBM® Print Mode	59
Demonstration programs	59
Printing text	60
ESCape codes	63
Page formatting in IBM mode	68

CHAPTER 6, Dot Graphics in Commodore Mode	73
Introduction to dot graphics	73
Bit image coding	75
Dot address determination	79
Double density coding	80

CHAPTER 7, Dot Graphics in IBM Mode	81
The print head	82
IBM dot graphics mode	83
Design your own graphics	90
Graphics programming tips	93

CHAPTER 8, Troubleshooting	99
Problem/solution chart	99
Beeper error warnings	101
Specific computer problems	101

APPENDICES

Appendix A, MPS Character Fonts, PETASCII and IBM	
Extended ASCII Codes	A-1
Appendix B, Secondary Addresses and Control Codes	B-1
Appendix C, Formatting Examples	C-1
Appendix D, Serial and Parallel Interfaces	D-1
Appendix E, Technical Specifications	E-1
User's Manual Statement	Inside Back Cover

Introduction

The Commodore MPS 1000 is a full-function dot-matrix printer that allows you to create printed copies of documents, programs, or illustrations you design with your computer. The documents you print can appear in several character styles, and the ability to use several bold and enhanced features adds variety to the printed material.

The MPS 1000 can function with other Commodore equipment connected with the serial port, or with any other equipment that can use a Centronics® parallel port for connection, such as IBM®-compatible machines.

Operating Modes

The MPS 1000 has two basic modes of operation:

- Commodore mode
- IBM® 5152 Plus* mode

The printer is shipped from the factory ready to be used in Commodore mode, but either mode can be chosen before the printer is turned on by setting a DIP switch on the back of the printer. See Section 1, Setting Up Your MPS 1000, for details about changing the DIP switch setting if you want to use IBM mode.

Your MPS 1000 can be connected through the serial bus to use the Commodore mode or the IBM mode. The parallel port connection can be used for IBM mode only. This means that serial-connecting computers, like the Commodore 128, can take advantage of both print modes through the serial bus.

*This mode is IBM-compatible with enhancements.

Character Styles

Each character the MPS 1000 prints is made up of dots, arranged in a specific pattern stored in the printer's permanent memory. The MPS 1000 has four styles of characters, called fonts, which allow you flexibility in the appearance of the printout. The four fonts are:

- Draft Quality characters, Commodore mode
- Draft Quality characters, IBM mode
- Near Letter Quality (NLQ) characters, Commodore mode
- Near Letter Quality (NLQ) characters, IBM mode

Draft characters are legible, but are often considered inappropriate for final drafts because they are less dense and therefore more difficult to read than NLQ characters. NLQ characters are made up of many more dots than draft characters, so the characters appear more fully formed.

In Commodore mode, the Commodore character set, including graphics, is available. In IBM 5152 Plus mode (IBM mode), the IBM character sets are available. The full character sets are displayed in Appendix A.

Using the MPS 1000

Many applications programs, including word processors and spreadsheets, contain all the built-in printer instructions needed in order to print successfully. Certain programs, however, may not contain all of the print functions you require. For example, you may wish to use a raised numeral superscript to signify a footnote, but the program you have chosen doesn't support this function. Your MPS 1000, however, has the ability to print superscripts and many other enhancements, and a short BASIC program can tell the printer to use these special features.

About This Manual

You can read as much or as little of this manual as you need. If you have used printers before and have a specific program that you want to use with the MPS 1000, a quick reading of the first chapter may be all you need. If, on the other hand, you are new to computers and printers, you will find this manual easy to follow and the MPS 1000 easy to use. No matter what your background is, if you want to learn about all the advanced features of the MPS 1000, the information you need is here.

For a preview of what your MPS 1000 can do look at the following samples of a few of its typestyles and graphics.

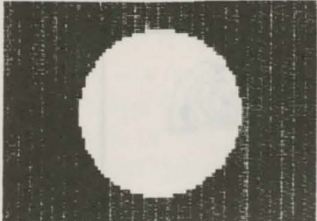
NEAR LETTER QUALITY

NLQ standard ABCDEFGHIJKLMnopqrstuvwxyz
NLQ emphasized ABCDEFGHIJKLMnopqrstuvwxyz

DRAFT MODE

Pica ABCDEFGHIJKLMnopqrstuvwxyz
Elite ABCDEFGHIJKLMnopqrstuvwxyz
Condensed ABCDEFGHIJKLMnopqrstuvwxyz
Underline ABCDEFGHIJKLMnopqrstuvwxyz
Emphasized enlarged

~~REVERSE PRINTING~~
REVERSE PRINTING OFF
~~REVERSE~~ AND STANDARD PRINTING



Chapter 1

Setting Up Your MPS 1000 Printer

Unpacking Your Printer

After you unpack your printer, carefully check to see that you have everything you should, and that nothing is damaged. Don't throw away any of the packaging material until you have located all the contents of the carton. In addition to this manual, the package should contain:

1. Commodore MPS 1000
2. Serial cable
3. Paper feed knob
4. Paper separator
5. Warranty card
6. Printer ribbon cassette
7. Snap-on pull tractor

All the parts of the MPS 1000 are identified below in Figure 1-1.

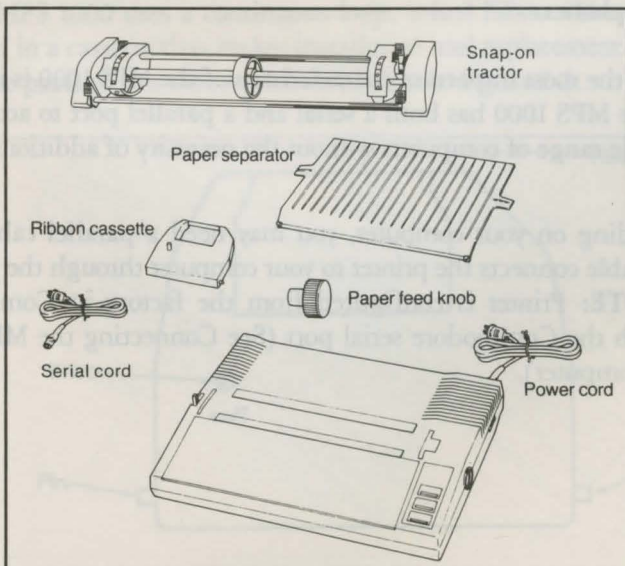


Figure 1-1. Printer parts

Getting Your MPS 1000 Printer Ready To Use

Setting up your MPS 1000 printer is a matter of attaching two parts, putting in the ribbon and paper, and connecting the printer to your computer.

This chapter will have you printing a test pattern within fifteen to twenty minutes and doing more complicated work not long after.

Now that your printer is unpacked, you need to find the best place to put it. Keep the following tips in mind:

- Place the printer close enough to the computer for the cables to reach.
- Place the printer on a stable surface. Avoid setting the MPS 1000 on carpeting or chairs.
- Use a grounded outlet and do not use an adapter plug.
- Avoid using outlets controlled by wall switches. A switch, turned off by mistake, could destroy valuable information in your computer's memory and stop your printing.
- Try not to use an outlet on the same circuit breaker with large electrical machines or appliances. These can cause disruptive power fluctuations.
- Protect your printer from direct sunlight, excessive heat, moisture, and dust.
- Keep your printer and your computer away from base units for cordless telephones.

One of the most important characteristics of the MPS 1000 is its flexibility. The MPS 1000 has both a serial and a parallel port to accommodate a wide range of computers without the necessity of additional interfaces.

Depending on your computer, you may need a parallel cable. The parallel cable connects the printer to your computer through the parallel port. **NOTE:** Printer is configured from the factory in Commodore mode with the Commodore serial port (See Connecting the MPS 1000 to Your Computer).

Installing The Paper Feed Knob

Before you can load paper into your MPS 1000, you need to install the paper feed knob. Simply push the knob onto the shaft on the right side of the printer. The shaft has one flat side that must be matched with the flat side of the hole in the knob. See Figure 1-2 below.

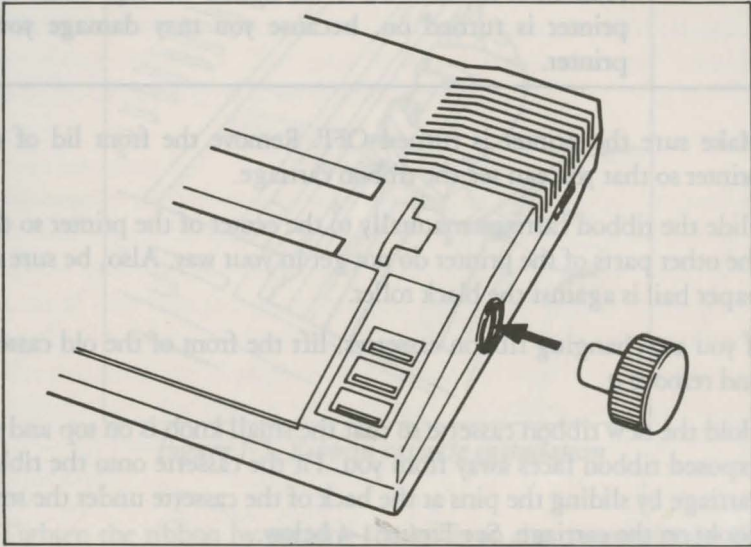


Figure 1-2. Paper feed knob installation

Installing Or Replacing The Ribbon

The MPS 1000 uses a continuous-loop, inked fabric ribbon which is enclosed in a cassette that makes installation and replacement clean and easy. The parts of this case are labeled below in Figure 1-3.

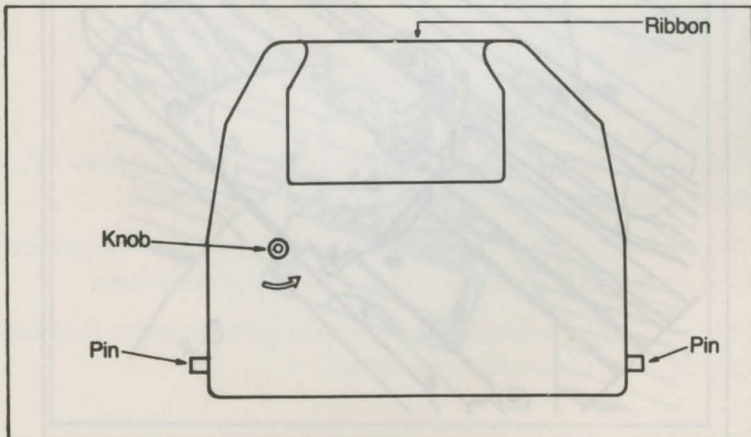


Figure 1-3. Ribbon cassette

To install or replace the ribbon, just follow these steps:

CAUTION: If you have been using your printer just before you change ribbon cassettes, be careful not to touch the print head because it becomes hot during use. Also remember never to move the ribbon carriage when the printer is turned on, because you may damage your printer.

1. Make sure the printer is turned OFF. Remove the front lid of the printer so that you can see the ribbon carriage.
2. Slide the ribbon carriage manually to the center of the printer so that the other parts of the printer do not get in your way. Also, be sure the paper bail is against the black roller.
3. If you are changing ribbon cassettes, lift the front of the old cassette and remove it.
4. Hold the new ribbon cassette so that the small knob is on top and the exposed ribbon faces away from you. Fit the cassette onto the ribbon carriage by sliding the pins at the back of the cassette under the small hooks on the carriage. See Figure 1-4 below.

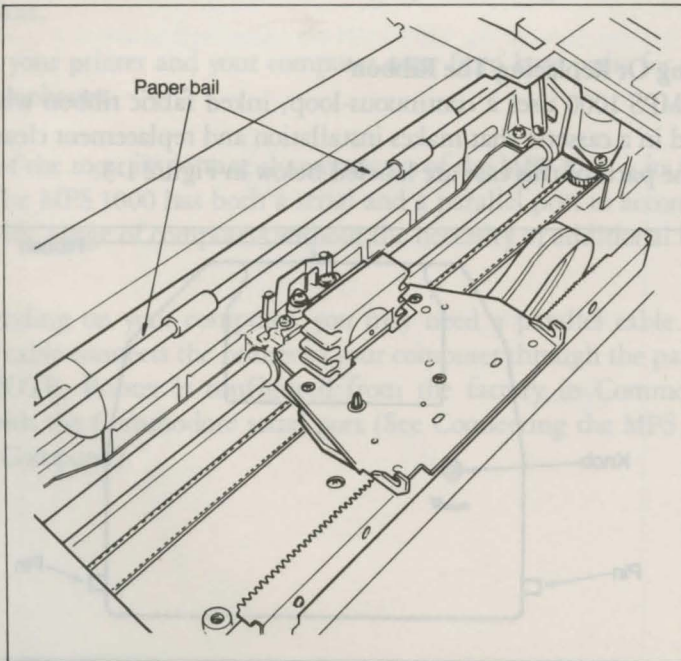


Figure 1-4. Print head assembly

5. Snap the cassette into place so that the exposed ribbon fits between the print head and the silver ribbon guide. See Figure 1-5. below.

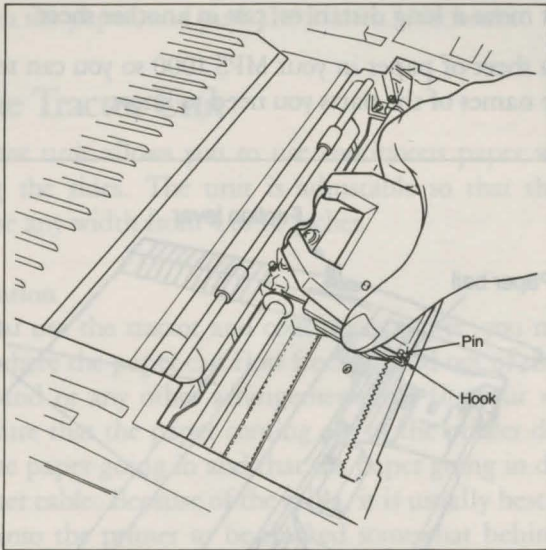


Figure 1-5. Ribbon cassette installation

6. Tighten the ribbon by turning the knob on the cassette in the direction of the arrow. As you turn the knob, make sure the ribbon slips down into its proper place between the print head and the silver ribbon guide. See Figure 1-6 below.

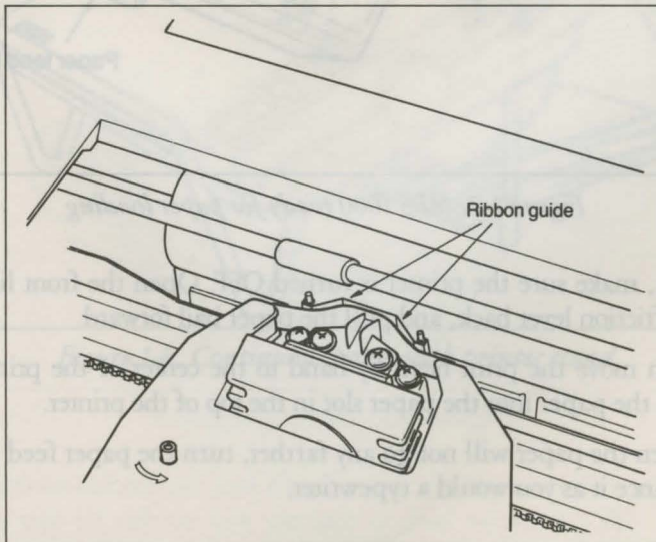


Figure 1-6. Ribbon placement

Paper Loading

A sheet of paper is already in the printer to protect the paper-out sensor during shipping. Remove it before loading any paper. If you ship the printer or move it long distances, put in another sheet.

Now put a sheet of paper in your MPS 1000 so you can test it. Figure 1-7 shows the names of the parts you need to know.

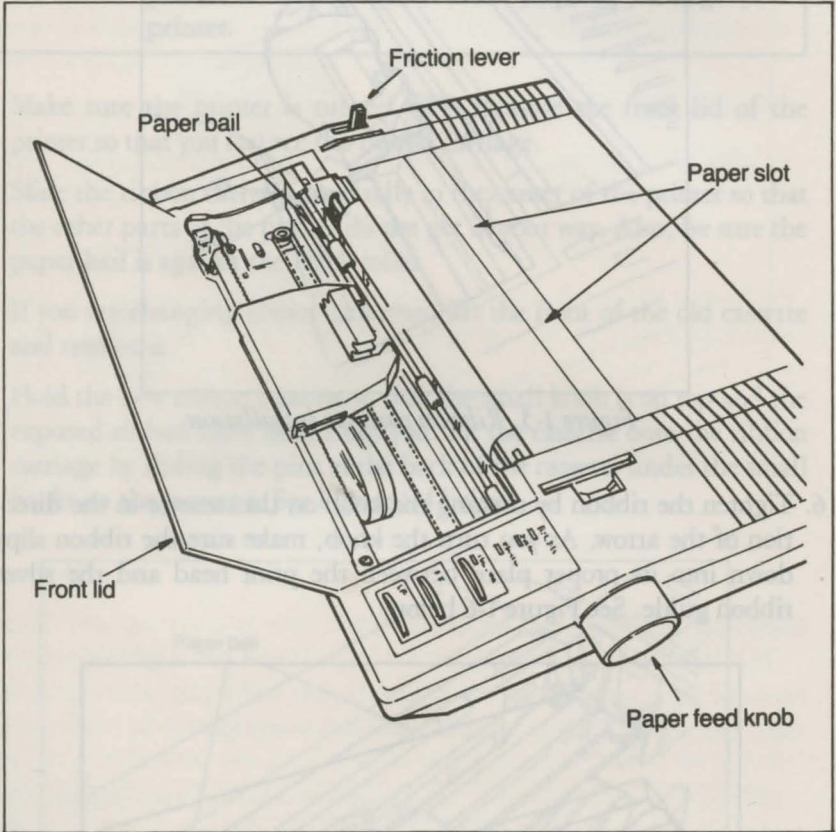


Figure 1-7. MPS 1000 ready for paper loading

1. First, make sure the printer is turned OFF. Open the front lid, push the friction lever back, and pull the paper bail forward.
2. Then move the print head by hand to the center of the printer and feed the paper into the paper slot in the top of the printer.
3. When the paper will not go any farther, turn the paper feed knob to advance it as you would a typewriter.

4. Turn the knob until the top of the paper is at least $3/4$ inch above the ribbon guide. Then push the paper bail against the paper. If the paper becomes crooked, pull the friction release lever forward, straighten the paper, and push the friction lever back.

Using the Tractor Unit

The tractor unit allows you to use continuous paper with pin feed holes along the sides. The unit is adjustable so that the continuous paper can be any width from 4 to 10 inches.

Printer Location

When you use the tractor and continuous paper, you must put your MPS 1000 where the paper can flow freely in and out of the printer. Use a printer stand or any other arrangement that fits your working area. Just make sure that the paper coming out of the printer does not interfere with the paper going in and that the paper going in does not catch on the printer cable. Because of the cable, it is usually best for the paper that feeds into the printer to be stacked somewhat behind the printer instead of directly beneath it. Two possible setups are shown in Figures 1-8 and 1-9.

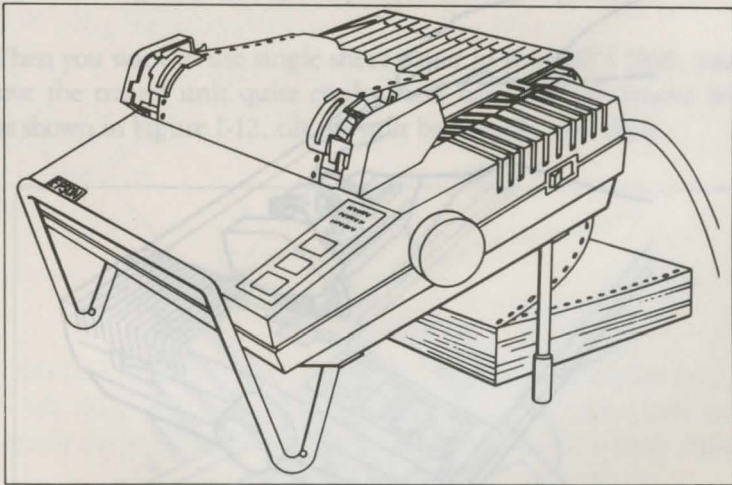


Figure 1-8. Continuous paper with printer stand

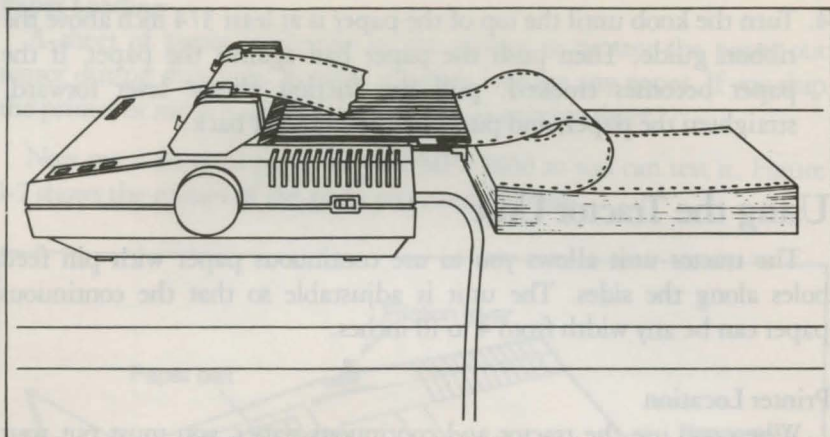


Figure 1-9. Continuous paper without stand

Tractor Unit Installation

To install the removable tractor unit, first pull the friction lever toward the front of the printer. Then hold the tractor with the gears to the right as shown in Figure 1-10.

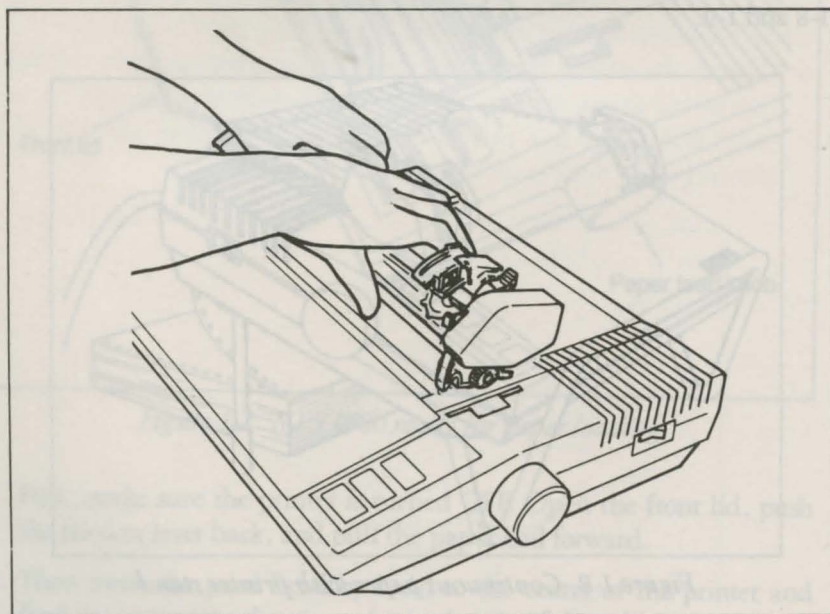


Figure 1-10. Tractor placement

In each tractor slot are two pegs that fit into the notches on the tractor fittings. Tilt the tractor back so that the rear notches fit over the rear pegs. Then tilt forward until it clicks into place. That's all there is to it.

Install the paper separator and pull out the paper guide as shown in Figure 1-11. Fit the notches in the bottom corners of the separator over the pins at the front of the paper slot. The separator keeps the paper that is coming out of the printer from being pulled back in. Pull out the paper guide at the back of the printer. This guide helps keep the incoming paper from catching on the printer cable.

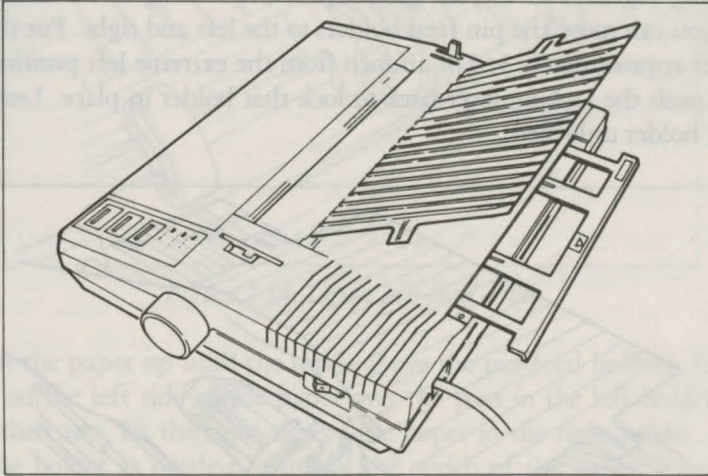


Figure 1-11. Paper separator and paper guide

When you want to use single sheet paper in your MPS 1000, you can remove the tractor unit quite easily. Push back the two tractor release levers shown in Figure 1-12, tilt the unit backwards, and lift it.

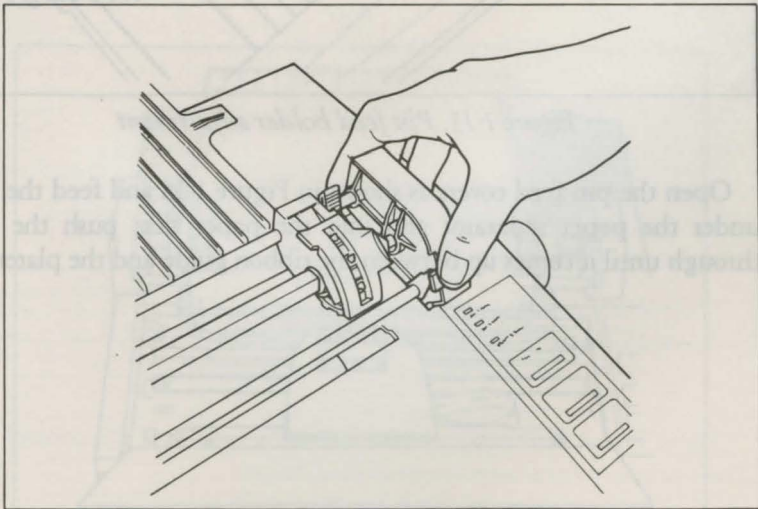


Figure 1-12. Tractor release levers

Loading Continuous Paper

Once your tractor is installed, you can load continuous paper. See that the printer is turned off and open the front lid. Then move the print head to the center of the printer. Pull the paper bail away from the platen (the black roller) just as you did for single sheet loading. (Look at Figure 1-7 if you need to check on the names of any of these parts.)

Using Figure 1-13 as your guide, pull the locking levers forward so that you can move the pin feed holders to the left and right. Put the left holder approximately $\frac{3}{4}$ of an inch from the extreme left position and then push the locking lever back to lock that holder in place. Leave the other holder unlocked.

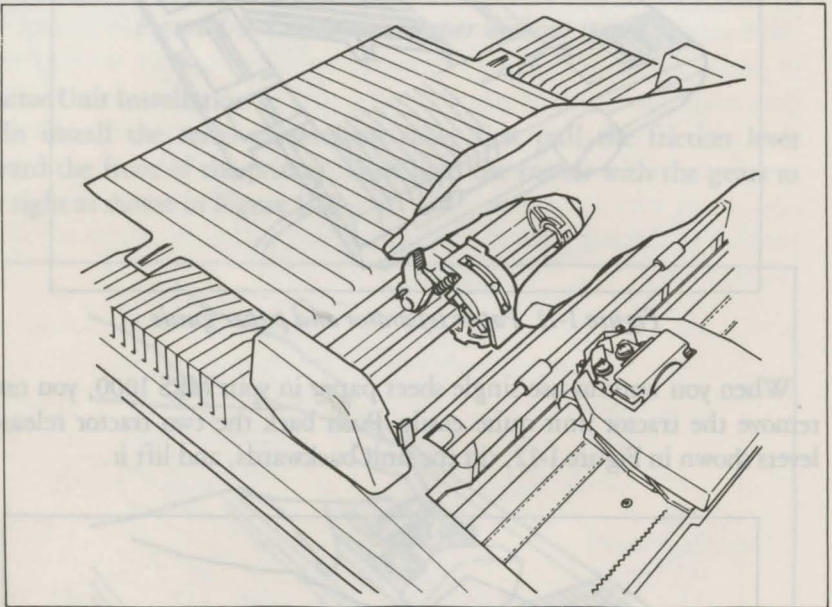


Figure 1-13. Pin feed holder adjustment

Open the pin feed covers as shown in Figure 1-14 and feed the paper under the paper separator and into the paper slot; push the paper through until it comes up between the ribbon guide and the platen.

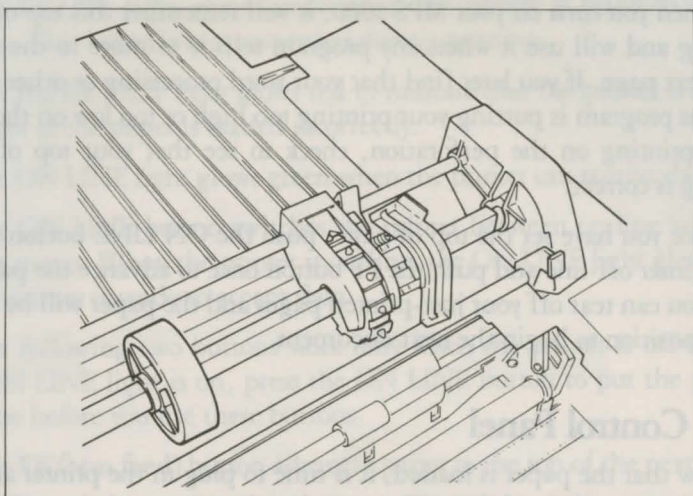


Figure 1-14. Open pin feed cover

Pull the paper up until the top is above the pin-feed holders. Fit the holes on the left side of the paper over the pins in the left holder and close the cover. Fit the right side of the paper in the right holder, moving the holder as needed to match the width of the paper. Close the second cover, make sure the paper has no dips or wrinkles, lock the right holder in place and push the paper bail against the paper.

Now you are ready to set the top of page position. Turn the paper-feed knob to advance the paper until a perforation between sheets is approximately $1/8$ of an inch below the top of the ribbon guide, as you see in Figure 1-15.

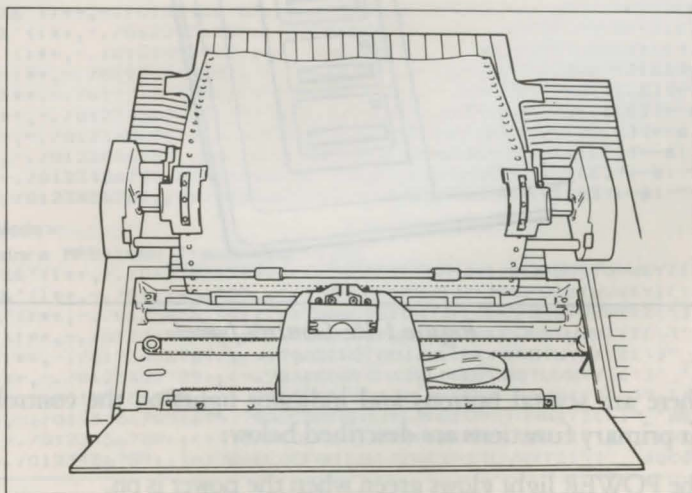


Figure 1-15. Top of page position

When you turn on your MPS 1000, it will remember this top of page setting and will use it when any program tells it to move to the top of the next page. If you later find that your word processing or other applications program is putting your printing too high or too low on the page or is printing on the perforation, check to see that your top of page setting is correct.

Once you have set the top of page, push the ON LINE button to put the printer off-line and push the FF button once to advance the paper so that you can tear off your just-printed pages and the paper will be in the right position to begin the next document.

The Control Panel

Now that the paper is loaded, it is time to plug in the printer and see what the control panel does. First, see that the power switch on the right side of the printer is turned off; then plug in the power cord. Now turn on the power switch and look at the control panel on the right side of the top of your MPS 1000. See Figure 1-16 below.

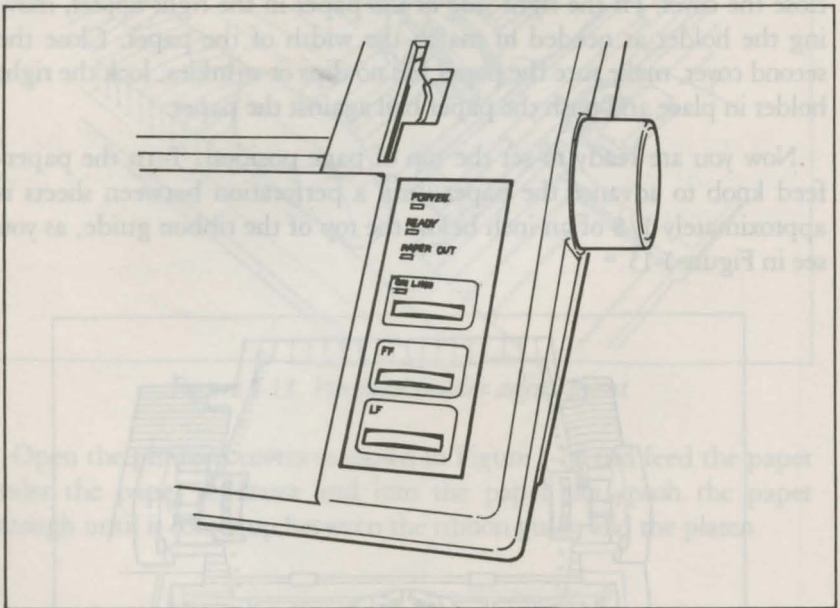


Figure 1-16. Control panel

There are several buttons and indicator lights on the control panel. Their primary functions are described below:

- The POWER light glows green when the power is on.

- The READY light glows green when the printer is ready to accept data. This light flickers somewhat during printing.
- The PAPER OUT light glows red to indicate that the printer is out of paper or the paper is loaded incorrectly.
- The ON LINE light glows green when the printer can receive data.
- The ON LINE button switches the printer between on-line and off-line status. When the printer is on-line, the ON LINE light glows and the printer is ready to accept data.

The following two buttons work only when the printer is off-line. If the ON LINE light is on, press the ON LINE button to put the printer off-line before you use these buttons.

- The FF(form feed) button advances paper to the top of the next page.
- The LF (line feed) button advances paper one line at a time.

Test Pattern

Now you'll see your MPS 1000 print something even though it's not connected to a computer yet. Make sure your printer has paper in it and the power switch is off. Now, hold down the LF button on the control panel while you turn the printer on with the power switch. The MPS 1000 will begin printing all the letters, numbers, and other characters in its ROM (Read Only Memory) for the draft mode. When the printing starts, you can release the LF button; the printing will continue until you turn the printer off or to the end of the page.

```

<Commodore Mode>
COMMODORE MPS-1000 C-MODE:VER-1.01
!"#%&'()*+,-./0123456789;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`
!"#%&'()*+,-./0123456789;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`a
!"#%&'()*+,-./0123456789;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`ab
!"#%&'()*+,-./0123456789;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abc
!"#%&'()*+,-./0123456789;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcd
!"#%&'()*+,-./0123456789;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcde
!"#%&'()*+,-./0123456789;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdef
!"#%&'()*+,-./0123456789;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefg
!"#%&'()*+,-./0123456789;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghi

```

Figure 1-17. Test Pattern

Connecting The MPS 1000 To Your Computer

Now that the test pattern has shown that your printer is working well, it's time to hook it up to your computer. Turn off the printer and the computer when you do this.

Your printer is designed to connect directly by cable into your computer through either the serial port (six pins) or the parallel port (double-row of 36 pins).

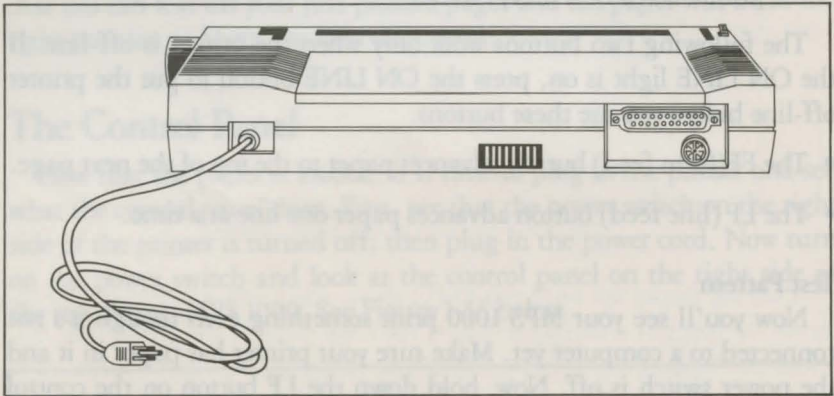


Figure 1-18. Printer Connections

- CAUTION:**
- When the MPS 1000's serial port is selected, any cable connected to its parallel port **MUST** be disconnected.
 - When the MPS 1000's parallel port is selected, any cable connected to its serial port **MUST** be disconnected.

An important feature of the MPS 1000 is its ability to print in two modes of operation: Commodore and IBM 5152 Plus. In its Commodore mode, the MPS 1000 can print Commodore business fonts and graphics. In its IBM mode, the MPS 1000 is able to produce characters in two different styles. Type selection is discussed fully in Chapter Four.

As delivered from the factory, the MPS 1000 set is in Commodore mode. To perform in its IBM-compatible mode, the DIP switch 1-1 located in the rear of the MPS 1000 must be changed from its OFF position to ON. DIP switch 1-3 selects the interface: ON for serial port and OFF for parallel. See Appendix C for its precise location.

If you are connecting the printer to your computer using the serial port:

1. Connect one end of the six-pin cable to the six-pin socket located in the back of your printer. Figure 1-19, below is a picture of the serial port. The cable is keyed so that you cannot plug it into the socket the wrong way. The pins are positioned so that a slight pressure will insert the pins properly. Do not force these pins; doing so could damage your cable.

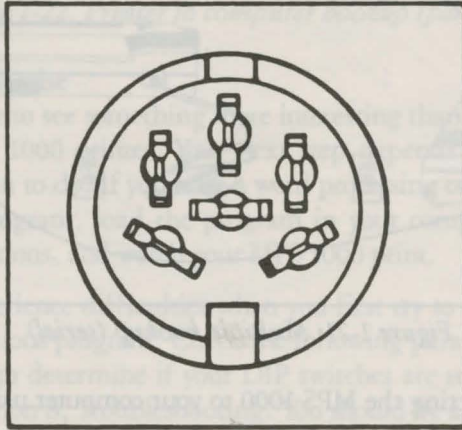


Figure 1-19. Serial port

2. Connect the other end of the cable to the serial port on the computer. Again, make sure you are keying the pins properly into the connector with six holes. Refer to figure 1-20 below.

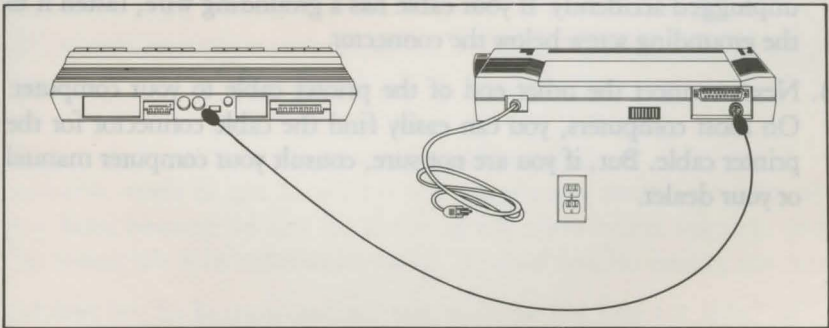


Figure 1-20. Printer to computer hookup (serial)

3. If you are using a disk drive, connect the printer to your drive unit using the serial port as in Step 2 above. Connect the drive to the computer. If any of the ports are not compatible with your cables, consult your dealer or your computer and disk drive manuals. Refer to Figure 1-21 below.

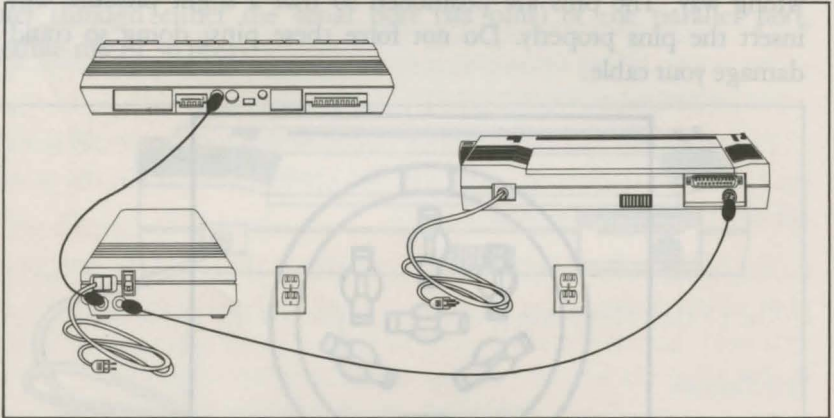


Figure 1-21: Multiple hookup (serial)

If you are connecting the MPS 1000 to your computer using the parallel port:

1. Refer to the Figure 1-22. Plug one end of your cable connector (36 pins) into the cable connector of the MPS printer. The plug is shaped so that there is only one way it will fit the connector.
2. Secure the plug to the printer with the wire clips on each side of the connector. These clips ensure that your cable will not be loosened or unplugged accidentally. If your cable has a grounding wire, fasten it to the grounding screw below the connector.
3. Next, connect the other end of the printer cable to your computer. On most computers, you can easily find the cable connector for the printer cable. But, if you are not sure, consult your computer manual or your dealer.

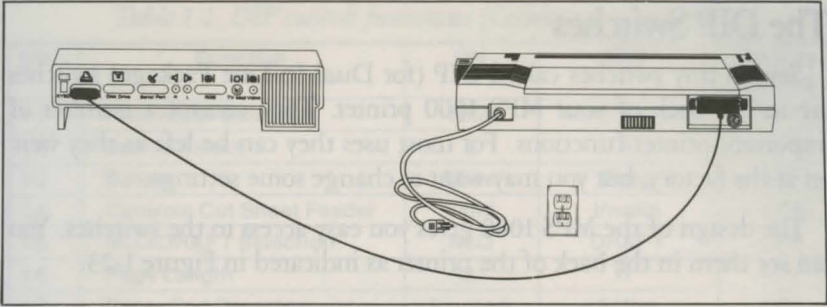


Figure 1-22. Printer to computer hookup (parallel)

First Printing Exercise

Now it is time to see something more interesting than the test pattern from your MPS 1000 printer. Your next step depends on the kind of printing you plan to do. If you have a word processing or other commercial software program, load the program in your computer, follow its printing instructions, and watch your MPS 1000 print.

You may experience difficulties when you first try to use your printer with an applications program. Check the following paragraphs on "The DIP Switches" to determine if your DIP switches are set appropriately, and consult Section 8, Troubleshooting. You should be able to solve your printing problems by following the instructions provided.

The DIP Switches

Several tiny switches called DIP (for Dual In-Line Package) switches are in the back of your MPS 1000 printer. They control a number of important printer functions. For most uses they can be left as they were set at the factory, but you may want to change some settings.

The design of the MPS 1000 gives you easy access to the switches. You can see them in the back of the printer as indicated in Figure 1-23.

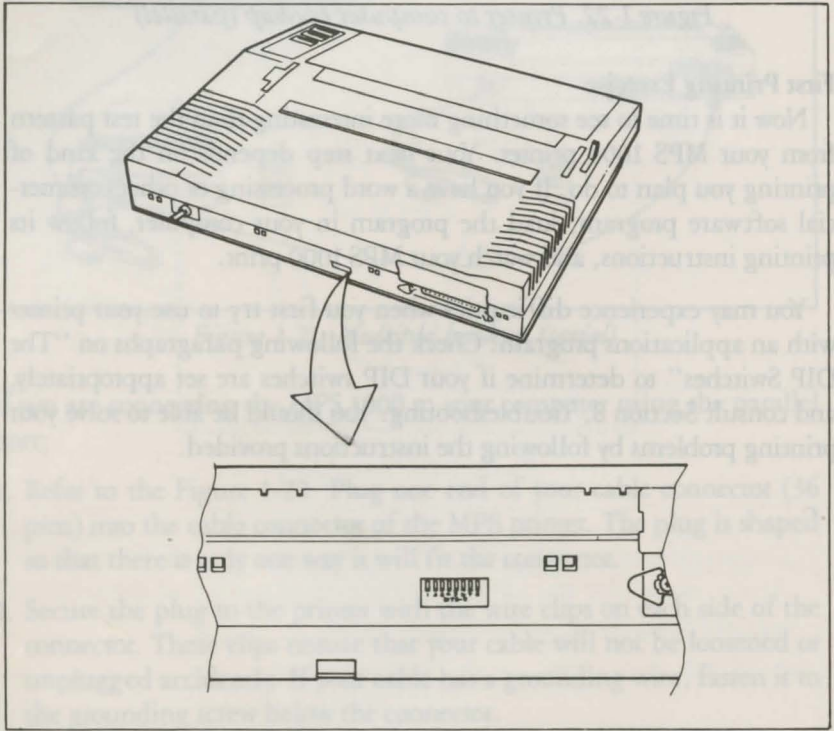


Figure 1-23. DIP switch location

Always turn the power OFF (with the switch on the right side of the printer) before you change the setting of any of these switches. Any changes made while the power is on will be ignored until you turn the printer off and back on. So set all switches with the power off.

In Table 1-1 and 1-2 we show you the functions of all the switches in each mode.

DIP switch 1-1 controls the mode of the MPS 1000. If switch 1-1 is OFF, that is, when the printer is in its Commodore mode, the following Table 1-1 applies:

Table 1-1. DIP switch functions (Commodore mode).

SW	Function	ON	OFF	Factory Setting
1-1	Print Mode Selection	IBM	Commodore	OFF
1-2	Device Select	5	4	OFF
1-3	Buzzer	Mute	Beep	OFF
1-4	Controls Cut Sheet Feeder	Valid	Invalid	OFF
1-5	NLQ/DRAFT Selection	NLQ	DRAFT	OFF
1-6	Page Length	12"	11"	OFF
1-7	Paper End Detector	Inactive	Active	OFF
1-8	Uni-Directional Printing	Valid	Invalid	OFF

NOTE: Printer is set for Commodore mode at the factory.
DIP switches are read at power-on reset.

If DIP switch 1-1 is ON, the MPS 1000 is in its IBM mode. Table 1-2 applies to IBM mode.

Table 1-2. DIP switch functions (IBM mode)

SW	Function	ON	OFF	Factory Setting
1-1	Print Mode Selection	5152 +	Commodore	OFF
1-2	Character Graphic Select	Table 2	Table 1	OFF
1-3	Interface	Serial	Parallel	OFF
1-4	Controls Cut Sheet Feeder	Valid	Invalid	OFF
1-5	NLQ/DRAFT Selection	NLQ	DRAFT	OFF
1-6	Page Length	12"	11"	OFF
1-7	Paper End Detector	Inactive	Active	OFF
1-8	Auto LF with CR	Valid	Invalid	OFF

*With DIP Switch 1-1 in the ON state (IBM mode) and DIP Switch 1-3 in the OFF state (Centronics Parallel interface), DIP Switch 1-2 is CG Select rather than Device Select. Note: Default is Table 1 if Commodore Serial Bus is selected with IBM 5152 + Mode).

DIP switches are read at power-on reset.



Printer Maintenance

Always

Always keep your MPS 1000 printer in a safe and clean location. Keep it away from:

Dust and grease

Heaters and furnaces. Safe temperature range is 41°F (5°C) to 95°F (35°C)

Now and Then

Clean particles and dust from the printer every so often with a soft cloth or brush. Use a mild cleanser for the outside framework. After removing the ribbon cassette, use denatured alcohol for the inside.

See your Commodore dealer for replacement ribbon cassettes.

Rarely

Once in a great while your printer should be lubricated. Commodore recommends two lubricants: O-2 (Epson Part No. B710200001) and G-2 (Epson Part No. B70020001). Every 6 months or 1 million lines, use O-2 lubricant on the shafts and platen bearings, wiping off excess with cloth. Use G-2 sparingly on ribbon feed, line feed and carriage return drive gears every 5 million lines.

Note: Using too much lubricant can cause problems with your printer. Apply the lubricant sparingly.

Chapter 2

Elements of Dot Matrix Printing

This chapter is for those who want to know something about how your printer works. It's a simple, non-technical explanation of the basics of dot matrix printing that will help you understand some of the later chapters.

The Print Head

The MPS 1000 uses a print head with nine pins or wires mounted vertically. Each time a pin is fired, it strikes the inked ribbon and presses it against the paper to produce a dot. This dot is about $1/72$ nd of an inch in diameter. The size varies slightly depending upon the age of the ribbon and the type of paper used. As the head moves horizontally across the page, these pins are fired time after time in different patterns to produce letters, numbers, symbols or graphics.

For example, to print a pica capital T, the head fires the top pin, moves $1/60$ th of an inch, fires the top pin again, moves $1/60$ th of an inch, fires the top seven pins, moves $1/60$ th of an inch, fires the top pin, moves another $1/60$ th of an inch, and fires the top pin once more to finish the letter. All this happens in only $1/100$ th of a second.

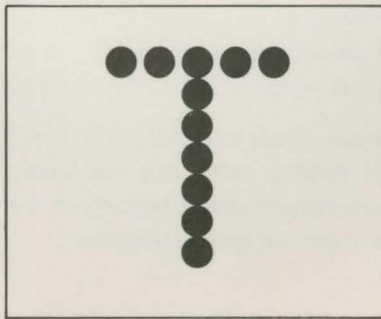


Figure 2-1. A capital T

Bidirectional Printing

In this manual, we describe the action of the print head as moving from left to right, as a typewriter does. During its normal operation while printing in the draft mode, however, the MPS 1000 prints bidirectionally. That is, the print head goes from left to right on every other line. On the other lines it reverses everything and prints right to left.

By reversing both dot patterns and printing direction, the MPS 1000 produces a line that is correct and looks no different from a line printed from left to right. It does this to save time. Otherwise, the time the print head takes to go from the right margin back to the left would be wasted.

The intelligence of the printer takes care of all the calculations for bidirectional printing, so you don't have to be concerned about it. Do your part of the work as if the printer will be printing from left to right on each line and let the MPS 1000 do all the necessary calculations.

Changing Pitches (IBM Mode Only)

In addition to pica, in which there are 10 characters per inch, the MPS 1000 can also print in other widths, or **pitches**. It does so by reducing the distance between pin firings. In **elite** mode it prints 12 characters per inch. In **condensed** mode it prints slightly more than 17 characters per inch. The pattern of the dots is not changed, but the horizontal space between them is reduced.

In Figure 2-2 are enlargements of four sample letters in each of the three pitches. These letters are chosen to show how the MPS 1000 prints letters that are uppercase and lowercase, wide and narrow, and with and without descenders such as the bottom part of the y.



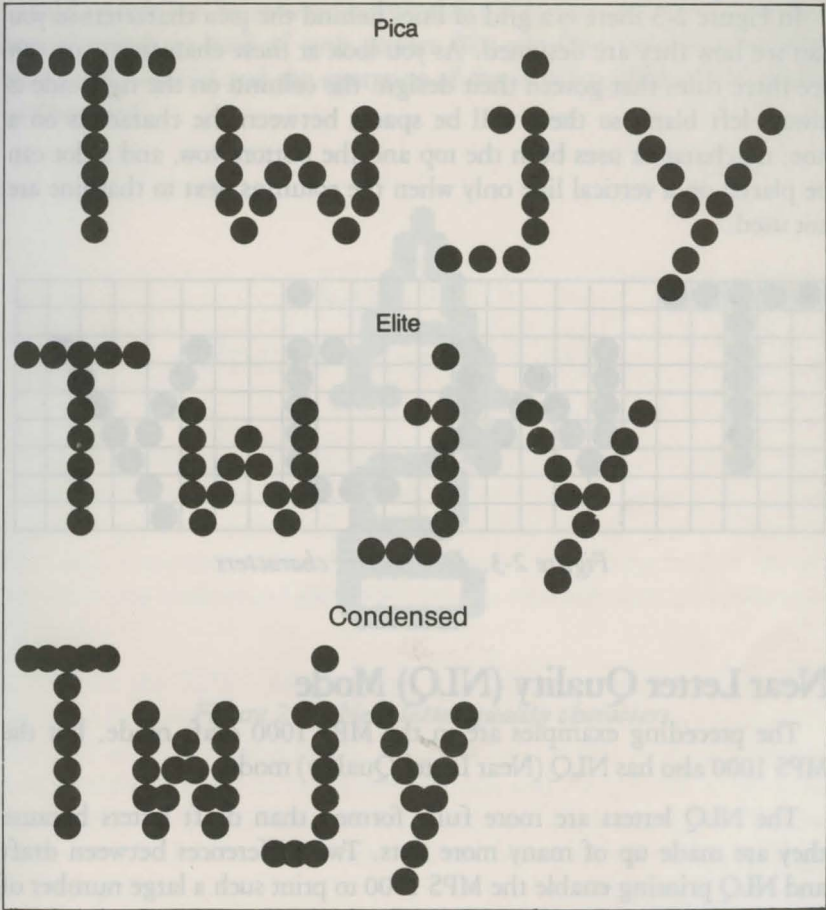


Figure 2-2. *The three pitches of the MPS 1000*

The dot pattern of each character is carefully designed so that in pica mode no dot overlaps another. The reason is that in normal high-speed printing of pica the pins cannot fire and retract and fire again quickly enough to print one dot overlapping another.

In Figure 2-3 there is a grid of lines behind the pica characters so you can see how they are designed. As you look at these characters, you can see three rules that govern their design: the column on the right side is always left blank so there will be spaces between the characters on a line; no character uses both the top and the bottom row, and a dot can be placed on a vertical line only when the columns next to that line are not used.

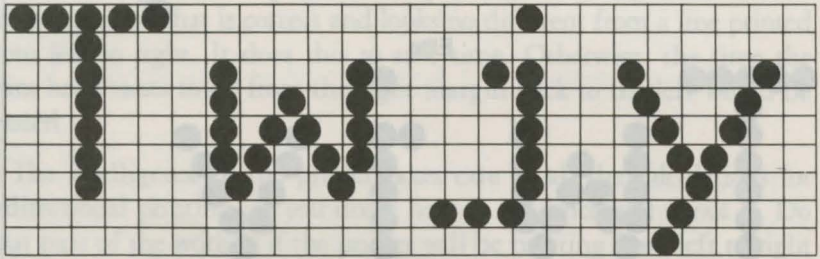


Figure 2-3. Dot matrix characters

Near Letter Quality (NLQ) Mode

The preceding examples are in the MPS 1000 draft mode, but the MPS 1000 also has NLQ (Near Letter Quality) mode.

The NLQ letters are more fully formed than draft letters because they are made up of many more dots. Two differences between draft and NLQ printing enable the MPS 1000 to print such a large number of dots for each character. In NLQ mode, the head moves more slowly, so dots can overlap horizontally. Each character is printed with two passes of the print head.

To further ensure the quality of NLQ characters, both passes of the print head are in the same direction so the alignment of the dots is exact.

Because NLQ mode uses two passes for each line and prints only in one direction, your printing does take longer in this mode. NLQ mode prints 20 characters per second (cps), rather than 100 cps in draft mode.

With the two modes, draft and NLQ, the MPS 1000 lets you choose high speed or high quality each time you print. You can print your ordinary work or preliminary drafts quickly in draft mode and use the NLQ mode for final copies or special purposes.

NLQ mode can be selected either with a software command or with a special switch in back of your printer. You can find the software command in Chapter 4 and the operation of the switch (called a DIP switch) in Chapter 1.

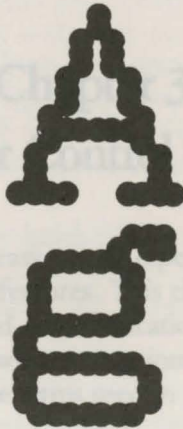


Figure 2-4. Near letter quality characters

If you are an advanced user or a programmer, you may want to refer to Appendix B, which has a full summary of all the MPS 1000 commands.

BITASCII and IBM Extension ASCII Codes

When you write a document with a word processing program, you press keys with letters on them. When you send the document to a printer, it prints the letters on paper. The computer and the printer, however, do not use or understand letters of the alphabet. They talk, and by manipulating numbers. Therefore, when you press the A key, for example, the computer sends a number to its memory. When the computer tells the printer to print that letter, it sends the number to the printer, which then converts the number to a picture of paper that will use to print the dots making up that letter.

Chapter 3

Printer Control Codes

The MPS 1000 printer is easy to use, especially with commercial software that has print control features. This chapter explains some of the basics of printer control and communications to help you understand how your computer works with your printer. This information should also help you understand the terms used in your software and computer manuals.

Because the MPS 1000 has both a Commodore mode and an IBM mode, it features two separate means of communicating with your computer. In its Commodore mode, the MPS 1000 is able to print Commodore letter and graphics characters in draft and NLQ fonts. In its IBM mode, your MPS 1000 has two character sets: IBM Character/Graphic (CG) set 1 and CG set 2. These will all be explained in this and the next chapter.

If you are an advanced user or a programmer, you may want to turn to Appendix B, which has a full summary of all the MPS 1000 commands.

PETASCII and IBM Extension ASCII Codes

When you write a document with a word processing program, you press keys with letters on them. When you send the document to a printer, it prints the letters on paper. The computer and the printer, however, do not use or understand letters of the alphabet. They function by manipulating numbers. Therefore, when you press the A key, for example, the computer sends a number to its memory. When the computer tells the printer to print that letter, it sends the number to the printer, which then converts the number to a pattern of pins that will fire to print the dots making up that letter.

The numbers that computers and printers use are in binary form, which means they use only digits 0 and 1. In this manual, however, we use decimal numbers in our explanations because most users are more familiar with these numbers and because most programming languages and applications programs can use decimal numbers. The computer system or the program takes care of changing the decimal numbers to binary form.

Computer and printer interaction would be confusing if different kinds of computers and printers used different numbers for the same letter of the alphabet. Therefore, most manufacturers of computers, printers, and software use the American Standard Code for Information Interchange usually referred to as ASCII. ASCII covers the decimal numbers from 0 through 127 and includes codes for characters (letters, punctuation, numerals and mathematical symbols) and a few control codes, such as codes for sounding the beeper and performing a carriage return.

Since ASCII codes run only from 1 to 127, there are extended versions of ASCII systems that feature additional characters and control codes. The two extended versions that you may make use of with this printer, PETASCII (the Commodore version of ASCII) and IBM Character Graphics Set ASCII, are different from standard ASCII.

In Commodore mode, a special extended version of the standard ASCII system is used. This code system is known as PETASCII (named for the first Commodore computer, the PET) or as Commodore ASCII. PETASCII is divided into two code tables, one for uppercase/graphic mode and one for lowercase/uppercase mode. Control code characters are from 0 to 31, and numbers and punctuation character codes (32–63) receive the same codes as in standard ASCII, but the remaining codes differ in PETASCII. One major difference in PETASCII is that lowercase characters occupy codes 64–95 instead of uppercase (as in standard ASCII). Uppercase characters in PETASCII are from 192–223, rather than from 96–127 of standard ASCII. The PETASCII codes that can be entered from the keyboard are 0–95 and 160–223. Codes 96–127 are the same as codes 192–223, but cannot be entered from the keyboard. Similarly, 224–254 are shadow codes of 160–191. Also, in PETASCII, the code value for 255 is the same as code 222. Something else to keep in mind is that while code values 128–159 are control codes like the values 0–31, in many cases they have opposite functions. For example, while code 18 turns reverse printing on, code 146 (18 + 128), which should perform the same function, actually turns reverse printing off.

The IBM mode character sets conform to a more standard version of ASCII. IBM divides into two ASCII systems, CG set 1 and CG set 2. These systems adopt the standard ASCII codes for values 0–127, with the extension of standard ASCII in 128–255. In IBM CG set 1, codes 128–159 are shadows of codes 0–31, and 160–255 are international characters, IBM graphic characters and scientific symbols. In CG set 2, codes 128–159 are assigned still more international characters, while codes 3–6 are assigned trump card symbols.

For a summary of PETASCII and IBM ASCII codes, refer to Appendix A of this manual.

Secondary Addresses and ESCape Codes

Although the original ASCII standard was designed to use the decimal numbers 1 through 127, computer and printer manufacturers soon extended this range (to 0 through 255) in order to make room for more features. On the MPS 1000 for example, the codes from 160 through 255 are used for graphic characters in Commodore uppercase/graphic mode, and graphic or international characters in IBM mode.

Because even this extended range is not enough for all the features used on modern printers, the range is extended in many Commodore systems through the use of what are called **secondary addresses**. The secondary address is used in Commodore BASIC to specify a printer function, such as setting the number of lines to be printed on a page or setting spaces between lines in the printer. This characteristic of Commodore BASIC will be discussed more in Chapter Four.

In many other systems, including those compatible with the IBM mode of the MPS 1000 printer, the range is extended through special ESCape codes (the code is often printed with the first three letters capitalized—as in ESCape—or abbreviated as ESC).

With the ESCape code, for which decimal 27 is used, printers and computers are not restricted to only 256 instructions. The ESCape code is a signal that the next code will be a printer control code instead of text to print. For example, if the printer receives the number 69, it prints a capital E because 69 is the ASCII code for that letter. If, however, the printer receives a 27 just before the 69, it turns on emphasized mode, because ESCape “E” is the code for emphasized.

You can see how important the ESCape code is by looking at Appendix B. There you will see that nearly every code the MPS 1000 uses in IBM mode is an ESCape code.

Programming Languages

You can write a program in BASIC or any other programming language to send control codes to your printer. In the upcoming chapters you will find examples of such programs. Just remember that your printer control code stays in effect for the whole document you print. This method is good for setting margins but does not work for italicizing a word.

Now you have some background on how printers work and how software can communicate with them. Turn to the next chapters to learn specific features of your MPS 1000 printer.

Chapter 4

Commodore Print Mode

Beginning with this chapter we describe many of the printing features of the MPS 1000. Although we include programs that demonstrate these features, you don't have to be a programmer to learn about the features from these chapters. How much of the rest of this manual you use depends upon your expertise, your interest, and your software.

If you have quite a bit of experience with computers, you may want to skip to Appendix B, which has a full description of the MPS 1000 commands. On the other hand, even if you are a computer novice, your software manual may contain all the printing instructions you need.

Demonstration Programs

Reminder: to use the MPS 1000 printer . . . the printer must be in Commodore mode. DIP switch 1-1 (accessible at the rear of the printer) must be in the OFF position. **IMPORTANT:** Never toggle the DIP switches while the printer is turned on. See Appendix D for more information.

Just about any program for a Commodore computer that is designed to produce single-color printed copy will work with the MPS 1000.

The demonstration programs included in these chapters are written in the BASIC programming language. Although we know that you will probably not use BASIC for most of your printing, we chose to include these programs because every Commodore computer has built-in BASIC.

You don't need to understand BASIC programming to run these programs. Although we supply a few notes about using Commodore BASIC with a printer in the next section, in most cases all you need to

do is turn on your computer and type in the programs exactly as they appear in this manual. If you have any questions, see your Commodore Computer User's Guide.

Although all the example programs are in BASIC, you can of course change them to any programming language that you prefer to use. Simply follow the instructions in the manual for your programming language.

The MPS 1000 and Commodore BASIC

In order to send any information to the printer, you must first open a file for it. A sample statement for opening a file is:

```
10 OPEN1,4
```

The 1 in the line above is the file number, and the 4 is the device number. You can use any number from 1 to 127 for the file number, for file numbers 128 to 255 the computer will insert a LF (linefeed) after each CR (carriage return). The device number should be 4 unless you have two printers. If you have two printers connected to your computer, you will sometimes use another device number. See Appendix D for further information.

After you have opened a file, you send information to the printer with a PRINT# statement, such as the sample below:

```
20 PRINT#1, "THE PRINTER PRINTS THIS"
```

The PRINT# statement is much like the PRINT statement in BASIC except that it must be followed by the same number of the file that has been OPENed and a comma. Anything inside quotation marks after that will be sent to the printer and printed.

As you will see in programs later in this chapter, you can also send instructions to the printer with a PRINT# statement.

After you have finished with a file, you should close it using the format below:

```
30 CLOSE1
```

All you need for this statement is the word CLOSE and the file number. If you wish, type in and run the three lines above, which form a program.

Each time you want to see the results of a program, type RUN and then press RETURN.

There is also a direct mode for sending commands or text to the printer. To use this mode, type the following without a line number before it:

```
OPEN1,4: CMD1
```

After you type this and press **RETURN**, everything that would normally be sent to the screen is sent to the printer instead.

To see how this mode works, type the following and press **RETURN**:

```
PRINT "DIRECT MODE"
```

The words **DIRECT MODE** will be printed on paper instead of appearing on the screen.

When you are through with the direct mode, cancel it with the following:

```
PRINT#1: CLOSE1
```

In the samples above, 1 is the file number and 4 is the device number. You can use any number between 1 and 255 as the file number, but be sure to use the same number in each statement.

One of the main uses for the direct mode is printing program listings. There are four steps to have your MPS 1000 print a listing of a program.

1. **LOAD** or enter the program.
2. Type **OPEN1,4: CMD1** and press **RETURN**.
3. Type **LIST** and press **RETURN**. The printer will now list your program.
4. Type **PRINT#1: CLOSE1** and press **RETURN**.

Step 4 cancels the direct mode so that you will not accidentally send information to the printer when you want it to appear on the screen.

Control Codes for Text

As you know, the Commodore computer can display graphic characters as well as letters of the alphabet. Given the right instructions, the MPS 1000 can print anything the computer can display on the screen, and it can also add several printing effects. These effects are described in the rest of this chapter.

Enlarged printing

The enlarged mode (sometimes called double width) of the MPS 1000 doubles the width of all the letters, symbols, and graphics characters. This mode produces striking headings and titles.

You can see printing in this mode by entering the following program.

```
10 OPEN1,4
20 PRINT#1, CHR$(14)"ENLARGED PRINTING"
30 CLOSE1
```

When you have entered the program, type RUN and press RETURN. You should see the results below.

```
ENLARGED PRINTING
```

Once you have turned on enlarged printing, it will stay on until you cancel it by using the canceling code or by turning off the printer. You can prove this with another short program.

First erase the other program from your computer's memory by typing NEW and pressing RETURN, but don't turn off your printer. Then type in and run this program. Remember that you run each program by typing RUN and pressing RETURN.

```
NEW
10 OPEN2,4
20 PRINT#2, "THIS IS STILL ENLARGED"
30 PRINT#2, CHR$(15)"NOW ENLARGED IS OFF"
40 CLOSE2
```

```
THIS IS STILL ENLARGED
NOW ENLARGED IS OFF
```

As you can see, the first line is still in enlarged printing, but the second one is not. The first is enlarged because the previous program turned on enlarged and it stays on until it is turned off. The second line is in standard printing because the code that turns off enlarged, CHR\$(15), comes before the words that are printed.

On the MPS 1000 printing modes are canceled in three different ways:

- With a specific canceling code, such as the CHR\$(15) that you used to cancel enlarged printing.
- By turning the printer off and back on.
- Automatically at the end of a line. Unlike enlarged, some printing effects stay on only for one line. Therefore, you don't have to turn them off if you use them for one whole line.

Near Letter Quality Mode (NLQ)

One feature of the MPS 1000 is the Near Letter Quality mode, which allows you to print documents in high-quality lettering. NLQ can be turned on by setting DIP switch 1-5 to ON; you can return your printer to draft quality mode by setting the same DIP switch back to OFF. It's also possible to use ASCII code to print in Near Letter Quality regardless of how the DIP switch is set. Issuing CHR\$(31) puts you in NLQ mode, while CHR\$(159) turns off NLQ to return to printing draft quality letters.

Example:

```
10 OPEN 4,4
20 PRINT #4, CHR$(31) "NEAR LETTER QUALITY ON:"
30 PRINT #4, CHR$(159) "NEAR LETTER QUALITY OFF"
40 CLOSE 4
```

Result:

```
NEAR LETTER QUALITY ON:
NEAR LETTER QUALITY OFF
```

Reverse printing

In reverse printing the letters and characters are white on a black background instead of the usual black on a white background. The code that turns this effect on is CHR\$(18), and the one that turns it off is CHR\$(146). Enter and run the following program to see this mode in action. Be sure to include the semicolon at the end of line 40. Again type NEW to erase the previous program before you enter this program.

```
NEW
10 OPEN 3,4
20 PRINT #3, CHR$(18) "REVERSE PRINTING"
30 PRINT #3, "REVERSE PRINTING OFF"
40 PRINT #3, CHR$(18) "REVERSE" CHR$(146);
50 PRINT #3, " AND STANDARD PRINTING"
60 CLOSE 3
```

```
REVERSE PRINTING
REVERSE PRINTING OFF
REVERSE AND STANDARD PRINTING
```

The printout above shows the two ways that reverse printing is turned off:

- By the ending of a print line
- By the CHR\$(146) code.

The words in line 20 are in reverse printing because that effect was turned on with `CHR$(18)`, but the next line is standard printing because reverse printing stays on for only one line unless it is turned on again. The first word of the last print line is in reverse printing, but the rest is in standard printing because reverse is turned off by `CHR$(146)`.

Uppercase/Graphic/Lowercase modes

As you know, the screen of your Commodore computer has two modes for displaying letters of the alphabet and graphics characters. They are the uppercase/graphic mode and the upper/lowercase mode.

In the uppercase/graphic mode (also called the cursor up mode), all letters are uppercase (capitalized) and many graphics characters are available. In the upper/lowercase mode (also called the cursor down mode), letters can be either upper or lowercase, but fewer graphics characters are available.

To change from uppercase/graphic to upper/lowercase on the screen, you press the `SHIFT` key and the `COMMODORE` key at the same time. This command affects only the screen.

If you do not give the printer any special instructions, it always prints in the uppercase/graphic mode. Therefore, all the examples so far in this chapter have been printed in the uppercase/graphic mode.

Before you type any programs that use the upper/lowercase mode, change the screen display to that mode by pressing the `COMMODORE` key and the `SHIFT` key at the same time. That way the letters will appear on your screen in the same mode that they are printed on paper.

No matter whether you print in the uppercase/graphic mode or the upper/lowercase mode, always have your screen display in the same mode that you plan to print to avoid confusion. If your screen is in the upper/lowercase mode and your printing is in the uppercase/graphic mode, you will get graphics characters when you want uppercase letters.

There are two ways to tell the MPS 1000 to use the upper/lowercase mode:

- Using an extra number (called a **secondary address**) when you open a file
- Using the code `CHR$(17)`.

For the first method, you use the number seven as the secondary address when you open the file for printing. This number goes after the device number, as shown in the example below.

```
new
10 open3,4,7
20 print#3, "This is upper/lowercase."
30 close3
```

When you run the program, you should get the following results:

```
This is upper/lowercase.
```

To test the CHR\$(17) code, type and run the following program. See that your screen is in the upper/lower case mode and be sure to include all the spaces and semicolons when you enter the program.

```
new
10 open3,4
20 print#3,chr$(17)"Upper/lowercase "
30 print#3, "now it is off"
40 close3
```

```
Upper/lowercase
NOW IT IS OFF
```

As you can see in the printout above, CHR\$(17) lasts for only one PRINT# statement without a semicolon after it.

Specifying Individual Upper- and Lowercase Characters

The printer default characters set is ASCII/Graphics when power is first turned on 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.

Example:

```
10 OPEN 4,4
20 PRINT #4, "<CRSR UP> C <CRSR DOWN> OMMODORE"
30 CLOSE 4
```

Result:

```
Commodore
```

Carriage Return Without Line Feed

You can instruct the printer to do a carriage return (CR) without moving down to the next line (a line feed, or LF), thereby repositioning at the first space on the current line, allowing you to overstrike the line two or more times. Enter the code for SHIFTeD RETURN, CHR\$(141), to issue a carriage return without a line feed. Enter the following program, which demonstrates how this can be used to boldface a line:

```
10 OPEN 4,4
20 PRINT#4, "CARRIAGE RETURN WITHOUT LINE
   FEED";CHR$(141);
30 PRINT#4, "CARRIAGE RETURN WITHOUT LINE FEED"
40 CLOSE4
```

Result:

```
CARRIAGE RETURN WITHOUT LINE FEED
```

You can also use this code to underline a sentence, by overprinting a second line of "underline" characters created by pressing the Commodore key with the @ key. The following program demonstrates underlining in this fashion:

```
10 OPEN 4,4
20 PRINT#4, "UNDERLINED CHARACTERS";CHR$(141);
30 PRINT #4, "_____ "
40 CLOSE 4
```

Result:

```
UNDERLINED CHARACTERS
```

Tabs

The MPS 1000 has two types of horizontal tabs. The first type divides the page into 80 columns, each one character wide. For this type, you send the code CHR\$(16) followed by the number of the column where you want the printing to begin. This number, which must be two digits, goes inside the quotation marks, just before the text. The number tells the printer where to start printing, but it is not printed. For example, the program below uses CHR\$(16) along with the number 15 inside the quotation marks. This makes the printing start in the fifteenth column. The numbers in line 20 give you a reference point so that you can see exactly where the next line starts. Notice that the first column is numbered zero, not one.

```

NEW
10 OPEN6,4
20 PRINT#6,"012345678901234567890"
30 PRINT#6,CHR$(16);
40 PRINT#6,"15THIS SKIPS 15 COLUMNS."
50 CLOSE6

```

```

012345678901234567890
                THIS SKIPS 15 COLUMNS.

```

The second type of horizontal tab gives more exact spacing by dividing the page into 480 columns, each 1/6th of a character wide. This type uses CHR\$(27) and CHR\$(16) followed by two other CHR\$ numbers. These other numbers specify the column where printing starts. The total number of columns is determined by adding the second number to 256 times the first number. Our samples in Table 4-1 illustrate the command.

Table 4-1. Dot spacing commands

Command	Number of columns
CHR\$(27)CHR\$(16)CHR\$(0)CHR\$(50)	50 (0 + 50 = 50)
CHR\$(27)CHR\$(16)CHR\$(0)CHR\$(186)	186 (0 + 186 = 186)
CHR\$(27)CHR\$(16)CHR\$(1)CHR\$(50)	306 (256 + 50 = 306)

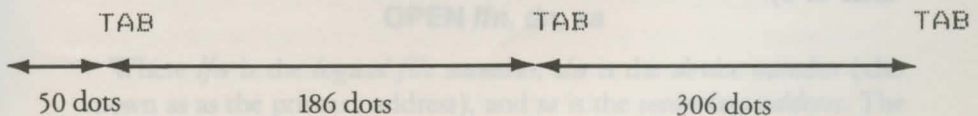
The following program shows how the three commands in Table 4-1 space the word "TAB" across the page.

```

NEW
10 OPEN3,4,0
20 FOR T = 1 TO 3
30 READ X
40 Y = INT(X/256): Z = X - (Y*256)
50 PRINT#3, CHR$(27)CHR$(16)CHR$(Y);
60 PRINT#3, CHR$(Z)"TAB";
70 NEXT T
80 PRINT#3
90 CLOSE3
100 DATA 50,186,306

```

Result:



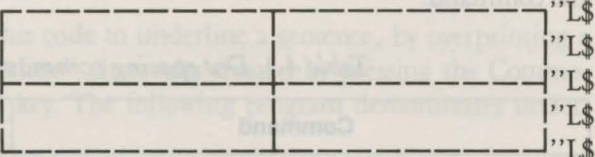
Line spacing

The CHR\$(8) code turns on the dot (or bit image) graphics mode, which is discussed later, but this code has another use. The CHR\$(8) code also reduces the space between print lines.

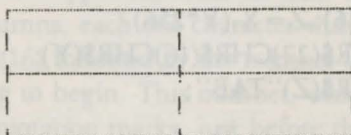
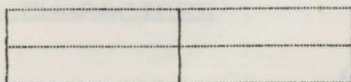
For printing text you want space between the lines, but occasionally with graphics characters you will want the space between lines eliminated so that the characters can combine into larger figures. The next program shows what we mean.

Lines 20-50 in this program use several graphics characters. To be sure that the lines meet each other, press SHIFT and * at the same time for the horizontal lines and SHIFT and - for the vertical lines.

```
NEW
10 OPEN2,4
20 S$ = CHR$(15): L$ = CHR$(8)
30 PRINT#2,S$""
40 PRINT#2,S$""
50 PRINT#2,S$""
60 PRINT#2,S$""
70 PRINT#2,S$""
80 CLOSE2
```



As you can see from the printout, the CHR\$(8) code allows you to combine some of the graphics characters to form boxes. Without the CHR\$(8) code, the vertical lines have gaps in them, as shown in the second printout.



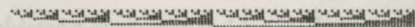
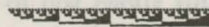
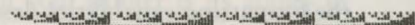
CHR\$(14) or CHR\$(15) returns the printer to the standard line spacing of $1/6$ " (when the line spacing is not defined by secondary address of 6).

Single and Double Density (Dot Graphics Mode)

Your printer usually prints on a single density line, which is 480 dots. However, you can make it print double density, or 960 dots per line. To turn on double density printing, issue the code `CHR$(2)`. To return to single density, the code is `CHR$(1)`. The following program shows how the two forms compare.

```
10 OPEN 4,4
20 A$ = " ":FOR I = 0 TO 127:A$ = A$ + CHR$(I + 128):NEXT I
30 :
40 PRINT#4, "DEFAULT AFTER RESET          :";
50 PRINT#4, CHR$(8);A$;CHR$(15)
60 PRINT#4
70 PRINT#4, "CHR$(2)...DOUBLE DENSITY :";
80 PRINT#4, CHR$(2);CHR$(8);A$;CHR$(15)
90 PRINT#4
100 PRINT#4, "CHR$(1)...SINGLE DENSITY  :";
110 PRINT#4, CHR$(1);CHR$(8);A$;CHR$(15)
120 CLOSE4
```

Result:

```
DEFAULT AFTER RESET          : 
CHR$(2)...DOUBLE DENSITY : 
CHR$(1)...SINGLE DENSITY  : 
```

Quotation Marks

If an odd number of quotation marks (`CHR$(34)`) 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.

Advanced Control with the Secondary Address

The secondary address is an additional parameter to use with the OPEN command. The secondary address allows you to perform some more advanced control functions, such as setting the number of lines per page or line spacing. The format of an OPEN command which includes the secondary address parameter is:

OPEN *lfn*, *dn*, *sa*

Where *lfn* is the *logical file number*, *dn* is the *device number* (also known as the primary address), and *sa* is the *secondary address*. The secondary address position determines the format.

You can enter any of twelve secondary addresses in your OPEN command. These are:

- 0 Print data exactly as received in Uppercase/Graphics mode
- 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
- 7 Print data exactly as received in Upper/lower case
- 9 Suppress diagnostic message printing
- 10 Reset printer

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 Appendix B.

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:

```
10 OPEN 5,4
20 PRINT #5, "THIS IS A TEST!"
30 CLOSE 5
```

Result:

```
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 arranged 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 the example shows.

Example:

```
10 OPEN 2,4,2
20 OPEN 1,4,1
30 PRINT #2,"AAA AAA AAA"
40 PRINT #1,"ABC"CHR$(29)CHR$(160)CHR$(29)"DEF"
50 CLOSE 2:CLOSE 1
```

Result:

```
ABC      DEF
```

Note: The data to be printed for any formatted line must be sent with only one "PRINT#" command.

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 lets you left- or right-justify columns of data, or align numeric data on its decimal point position.

Example:

```
10 OPEN 2,4,2
20 OPEN 1,4,1
30 PRINT #2,"$$$ .99"
40 PRINT #1,3.05
50 CLOSE 2:CLOSE 1
```

Result:

```
$3.05
```

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 79 characters are accepted in a format string. Excess (overflow) characters are printed onto the next print line. Table 4-2 contains formatting examples.

Numeric

9 – Specifies a digit position in a numeric field, for either number places to the left or right of a decimal point. Leaves blank spaces for empty positions according to the format. The result is a numeric digit with the leading zeros suppressed.

Z – Also specifies a digit position in a numeric field, and prints leading zeroes (only to the left of a decimal point) instead of blank spaces in a numeric field.

\$ – If one \$ is specified, then the field is treated as a dollar amount with a fixed-sign position as follows:

\$123
\$1234.00

If all digit positions to the left of the 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 the 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 positive, a blank is printed.

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
 - \$\$\$9.99 Floating dollar sign
- c) Leading zeroes forced –
 - ZZZZ No decimal
 - zz.99 Decimal
- d) Trailing minus –
 - 999 — Blank printed if number is positive
- e) Signed numbers –
 - s99 + or – always printed

Table 4-2 contains more examples of sample data, format fields, and the resultant editing.

Table 4-2. Formatting Examples

FORMAT FIELD	DATA	EDITED RESULT
AAAAA	ABC	ABC
AAAAA	ABCDEF	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-

The printer software handles up to ten significant figures, and an exponent range of ± 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:

```
10 OPEN 2,4,2
20 OPEN 1,4,1
30 PRINT #2, "A AA AAA"
40 PRINT #1, "CBM"CHR$(29)"CBM"CHR$(29)"CBM"
50 CLOSE 2:CLOSE 1
```

Result:

```
C CB CBM
```

Skip (Blank):

Simply use blanks where spaces are required:

Example:

```
10 OPEN 2,4,2
20 OPEN 1,4,1
30 PRINT #2, "AAA AAA AAA"
40 PRINT #1, "PET"CHR$(29)"PET"CHR$(29)"PET"
50 CLOSE 2:CLOSE 1
```

Result:

```
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 exit in the printer's memory rather than being used to format other data. Literals are flagged by preceding each literal in the formal string with a reverse-field ON character.

One use of literal is to create a special form at the same time you are printing data. This is done by using the special graphics character 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 formal strings is that they cannot be printed in the reverse field mode. They are also restricted to the same character set in which data is currently being printed.

Example:

```

10 OPEN 2,4,2
20 OPEN 1,4,1
30 PRINT #2, "<CTRL><RVS ON> [AAAAA<CTRL><RVS ON>] <CTRL><RVS ON> (AAAAA<CTRL><RVS ON>) <CTRL><RVS ON> ?AAAAA<CTRL><RVS ON>?"
32 A$ = ""
35 FOR I = 1 TO 10:A$ = A$ + CHR$(64 + I)
40 PRINT #1, A$CHR$(29)A$CHR$(29)A$CHR$(29)
50 NEXT
60 CLOSE 2:CLOSE 1

```

Result:

```

[A   ] (A   ) ?A   ?
[AB  ] (AB  ) ?AB  ?
[ABC ] (ABC ) ?ABC ?
[ABCD] (ABCD) ?ABCD ?
[ABCDE] (ABCDE) ?ABCDE?
[ABCDE] (ABCDE) ?ABCDE?
[ABCDE] (ABCDE) ?ABCDE?
[ABCDE] (ABCDE) ?ABCDE?
[ABCDE] (ABCDE) ?ABCDE?
[ABCDE] (ABCDE) ?ABCDE?

```

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, CHR\$(147), described later in the chapter. When paging is on and the paging secondary address is not implemented, six blank lines are left at the bottom of the page (since the default is 60 lines). Standard page length is 66 lines, so there is one inch of blank space left at the bottom of the page. This assumes the top of the page is set at line 1. If you prefer to split the blank space between the top and bottom of the page, set the top of the page approximately three lines (a half inch) down from the top of the page.

Example:

```
10 OPEN 4,4
20 OPEN 1,4,1
30 OPEN 2,4,2
40 OPEN 3,4,3
50 A$ = "999 9999 99.99999999 99.99999999"
55 PRINT #2,A$
60 PRINT #3, CHR$(60)
70 PRINT #4, CHR$(147)
80 FOR I = 1 TO 99
90 PRINT #1,I,I*I;SQR(I);I↑(1/3)
100 NEXT I
110 PRINT #4,CHR$(19)
120 CLOSE 4:CLOSE 3:CLOSE 2:CLOSE 1
```

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.

Example:

```
10 OPEN 4,4,4:PRINT #4:CLOSE 4
20 OPEN 25,4,25
30 PRINT #25
40 CLOSE 25
```

Result:

PE:C

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 is printed exactly as 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 $0 < \text{lines per page} < 128$ 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. The secondary address is reset to zero.
- *PE:M* – Data-format mismatch. You gave the printer non-numeric data to print in a numeric field.
- *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 an exponential number is: $n.nnnnE + ee$ or $n.nnnnE - ee$. The expression must contain a plus or minus sign with a two digit exponent.
- *PE:F* – Bad format. The data that was sent to SA = 2 either contains illegal formatting characters, or 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)), or a carriage return line-feed sequence. Communication with the current SA must be ended with a terminator before attempting to “talk” to another SA.

Note: suppressing diagnostic message printing is accomplished with a secondary address of 9.

Defining a Programmable Character: $sa = 5$

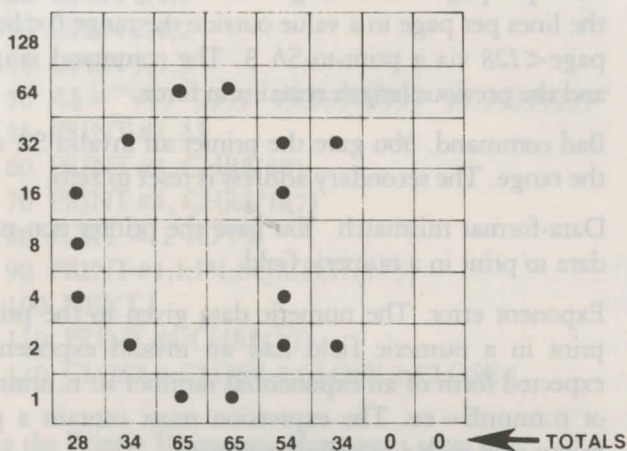
A secondary address of 5 allows you to create a custom character of your own. This programmable character is initialized with this secondary address.

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



Lay out an 8×8 matrix. 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:



The DATA statement in your program will read:

```
DATA 28,34,65,65,54,34,0,0
```

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 DATA 28,34,65,65,54,34,0,0
20 OPEN 5,4,5
30 FOR I = 1 TO 8:READ A:A$ = A$ + CHR$(A):NEXT I
40 PRINT#5,A$
50 OPEN 4,4
60 FOR I = 1 TO 10
70 PRINT#4,CHR$(14)CHR$(254)“<CURSOR UP>C<CURSOR
  DOWN>OMMODORE<CURSOR UP> B<CURSOR
  DOWN>USINESS <CURSOR UP>M<CURSOR DOWN>A-
  CHINES”
80 NEXT I
90 CLOSE 5
100 CLOSE 4
```

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 custom character buffer.

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.

Selecting Upper/Lower Case: $sa = 7$

Secondary address 7 is used to print data exactly as received. It is similar to SA0 except that all unshifted characters are printed in lower case and all shifted (or capital) characters are printed in upper case.

Example:

```
30 OPEN 4,4,7
40 PRINT #4, "MODEL MPS 1000 PRINTER"
50 CLOSE 4
```

Result:

```
model mps 1000 printer
```

Suppress Diagnostic Message Printing: $sa = 9$

To turn off the diagnostic message printing, send a secondary address of 9.

Example:

PE:M

COMMODORE

```
10 OPEN 4,4,4:PRINT #4:CLOSE 4
```

```
20 OPEN 2,4,2
```

```
30 OPEN 1,4,1
```

```
40 PRINT #2, "$$$$.9"
```

```
50 PRINT #1, "COMMODORE"
```

```
60 CLOSE 2:CLOSE 1
```

COMMODORE

```
10 OPEN 4,4,4:PRINT #4:CLOSE 4
```

```
15 OPEN 9,4,9:PRINT #9:CLOSE 9
```

```
20 OPEN 2,4,2
```

```
30 OPEN 1,4,1
```

```
40 PRINT #2, "$$$$.9"
```

```
50 PRINT #1, "COMMODORE"
```

```
60 CLOSE 2:CLOSE 1
```

Resetting the Printer: $sa = 10$

To reset the printer, send a secondary address of 10.

Example:

```
10 OPEN 10,4,10
```

```
20 PRINT #10
```

```
30 CLOSE 10
```

The commands for the modes used in this chapter are summarized in Appendix B.

Chapter 5

IBM Print Mode

Demonstration Programs

You can use IBM 5152 Mode, with IBM character/graphic sets (CG 1 and 2) and the special features available such as boldfacing and underlining, with a computer with a parallel port (such as the Commodore Amiga or an IBM computer) or a serial port computer (like the Commodore 128 or 64). To use IBM mode, you must reset DIP switch 1.1 to ON (see Chapter 1 on Setting the DIP Switches). You can also set the port you'll use to hook up the printer (parallel or serial bus) by setting DIP switch 1.3. This is also explained in Chapter 1.

To use the IBM mode with a Commodore serial bus, a standard setting for the DIP switches is to set switches 1-1, 1-3 and 1-8 to ON, setting the other five switches to OFF.

Where Commodore print mode uses secondary addresses in the BASIC language OPEN statement, IBM mode utilizes ESCape codes. These are issued by using the character codes for ESCape and the key or key sequence for what you want the printer to do. Since the MPS 1000 allows computers with both parallel and serial hook-ups to use IBM mode, this section includes two different BASIC language versions of each of the demonstration programs. One version is in Commodore BASIC, like the programs in Commodore mode. The second version of these programs is in Microsoft BASIC (or MBASIC), for other computers with that version of BASIC built in. Check your computer manual for the appropriate version.

You don't need to know anything about BASIC to type in and run these programs. Just check your BASIC manual to see how to load BASIC in your computer and how to run a program. As you run the programs (or even as you read the explanations and look at the printed examples), you learn how the MPS 1000 responds to the messages your

computer sends it by printing letters, numbers, symbols and graphics in various print modes.

Even if you never use BASIC again, you will know the capabilities of your printer.

If you don't want to do the exercises in BASIC, you don't have to. Many users are quite happy with their printers without ever learning any more about them than how to turn them on and off and how to load paper. Therefore, you shouldn't be intimidated by the information in this manual. In most cases the software that you use for word processing, business or graphics does the calculating and communicating with the printer for you.

If your computer system uses any other kind of BASIC, you may have to make a few changes; consult your computer manual to see if any modifications are necessary.

In Chapter 2 you saw the enlargements of the three MPS 1000 pitches. Now you'll learn how to produce them.

Printing Text

Pica Printing

The first exercise is a simple program to print a sample line of characters in pica, the standard pitch (or width). Type in this program for Commodore BASIC:

```
10 OPEN4,4
20 FOR X = 65 TO 105
30 PRINT#4,CHR$(X);
40 NEXT X:PRINT#4
50 CLOSE4
```

The MBASIC version of this same program:

```
40 FOR X = 65 TO 105
50 LPRINT CHR$(X);
60 NEXT X: LPRINT
```

Now run the program. You should get the results you see below, 10 pica characters per inch.

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ\|'`_ abcdefghi
```

Changing Pitches

Now you can try other pitches. As we explained in Chapter 3, the MPS 1000 uses the same pattern of dots for pica, elite and condensed characters, but it changes the horizontal spaces between the dots to produce the three different widths.

In elite mode there are 12 characters per inch. The MPS 1000 prints in elite when it receives the ESCape “:” command and prints in condensed when it receives the ASCII 15 command. Print a sample line of elite characters by adding this line to the previous MBASIC program:

```
20 LPRINT CHR$(27)“:”;
```

This line uses the command for elite, ESCape “:”, to turn on that mode. Your printout should look like the one below.

```
ABCDEFGHIJKLMNQPQRSTUVWXYZ[\]^_`abcdefghijkl
```

NOTE: No NLQ font is present for elite.

Canceling Codes

As you saw in the third version of the print pitch program, you must cancel a code when you do not want it any more. With very few exceptions, the MPS 1000 modes stay on until they are canceled. It is important to remember this because an MPS 1000 mode can stay on even if you change from BASIC to another type of software. For example, if you print a memo with a word-processing program after you run the program above, the printer will still be in condensed mode; therefore, the memo will be in condensed print. To cancel condensed, use ASCII 18.

To avoid having one program interfere with the printing modes of another, you can cancel a mode one of two ways:

- Each mode has a canceling code, which you can find in the discussion of the code and in Appendix B. Pica is an exception to this rule. To cancel pica, turn on elite or condensed.
- By resetting the printer, a method explained below.

Resetting the Printer

Resetting your MPS 1000 cancels all modes that are turned on. You can reset the printer with one of two methods:

- Sending the reset code (ESCape “@”)
- Turning the printer off, then on.

Either one of these methods returns the printer to what are called its defaults, which are the standard settings that are in effect every time you turn the printer on. The two effects of resetting the printer that you should be concerned with are: it returns printing to single-strike pica, thus canceling any other pitches or enhancements you may have turned on and the current position of the print head becomes the top-of-page setting.

Some of our demonstration programs end with a reset code so commands from one program will not interfere with commands in the next one. After you run a program with a reset code in it, remember to change the top-of-page setting before you begin printing full pages.

Pitch Comparison

Now that you have used three short programs to produce samples of the three main pitches, you can choose the pitch that you prefer or the one that best fits a particular printing job. Most people use either pica or elite for printing text and condensed for spreadsheets or other applications in which it is important to get the maximum number of characters on a line.

Near Letter Quality Mode

The examples so far in this chapter are in draft mode. You can see the NLQ mode with the following program for Commodore BASIC:

```
10 OPEN4,4
20 PRINT#4,CHR$(27)CHR$(120)CHR$(1);
30 FOR X = 65 TO 105
35 PRINT#4,CHR$(X);
40 NEXT X:PRINT#4
50 CLOSE4
```

The MBASIC version of this same program:

```
10 LPRINT CHR$(27)“x”CHR$(1);
20 FOR X = 65 TO 105
30 LPRINT CHR$(X);
40 NEXT X: LPRINT
```

Result:

ABCDEFGHIJKLMN O PQRSTU VWXYZ [\] ^ _ ‘ abcdefghi

Note you use a lowercase x, not a capital X, in line 10. Because of the high resolution of the NLQ mode, it prints only in pica, not in elite or condensed.

All the modes demonstrated in this chapter are compared in Table 5-1.

Print sample		CPI	Codes	
			On	Off
← 1 inch →				
Near Letter Quality		10.00	ESC "x" 1	ESC "x" 0
PICA PRINT		10.00		
Elite print		12.00	ESC ":",	
Compressed print		17.16	15	18

Table 5-1. Summary of MPS 1000 pitches (IBM mode)

Remember—you don't have to use BASIC to change modes; you can use any method that sends the printer the proper codes.

ESCAPE Codes

Now that you have seen how you can change the pitch of your MPS 1000 printing, we can show you many ways to vary or enhance your printing. So that you won't have to type dozens of programs to try all these features, we have provided a master program that can demonstrate any feature.

Bold Modes

Besides the pitches (pica, elite, and condensed) that we covered earlier, the MPS 1000 offers other typetypes, including two for bold printing—emphasized and double-strike.

Emphasized mode

In emphasized mode, the MPS 1000 prints each dot twice, with the second dot slightly to the right of the first. In order to do this, the print head must slow down so that it has time to fire, retract and fire the pins quickly enough to produce overlapping dots. As you can see in Figure 5-1, this method produces more fully-formed characters that are darker than single-strike ones.

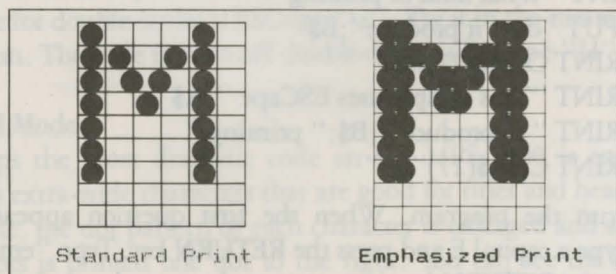


Figure 5-1. Single-strike and emphasized printing

Emphasized works only in draft pica and NLQ modes. In elite and condensed the dots are already so close together that even with the reduced print speed, the MPS 1000 cannot fire, retract and again fire the pins quickly enough to print overlapping dots.

You sacrifice some print speed and ribbon life with emphasized, because the print head slows and prints twice as many dots, but the increase in print quality is well worth it. You may want to use emphasized instead of NLQ mode for some purposes because emphasized printing is faster than NLQ printing.

Now that you have seen our example of emphasized printing, here's a master program that allows you to test almost any of the ESCape codes, including the ESCape code to turn on emphasized: ESCape "E".

Master program

First type in the program below. The program asks you what codes you want to test and then prints a sample of what the codes do. Be sure to type in the blank spaces in lines 70 and 80.

If you are using Commodore BASIC:

```
10 OPEN4,4
20 PRINT "WHICH ESCAPE CODE DO YOU"
30 INPUT "WANT TO TEST";A$
40 PRINT "WHAT KIND OF PRINTING"
50 INPUT "DOES IT PRODUCE";B$
60 PRINT#4, CHR$(27)A$
70 PRINT#4, "THIS SAMPLE USES ESCAPE ";A$
80 PRINT#4, " TO PRODUCE ";B$;" PRINTING."
90 PRINT#4, CHR$(27)"@"
100 CLOSE4
```

If you are using Microsoft BASIC:

```
20 PRINT "Which ESCape code do you"
30 INPUT "want to test";A$
40 PRINT "What kind of printing"
50 INPUT "does it produce";B$
60 LPRINT CHR$(27)A$
70 LPRINT "This sample uses ESCape ";A$
80 LPRINT " to produce ";B$;" printing."
90 LPRINT CHR$(27)"@"
```

Now run the program. When the first question appears on the screen, type a capital E and press the RETURN key. Type "emphasized" and press the RETURN key in answer to the second question. The pro-

gram is easy to use. Just remember to press the RETURN key after the answer to each question and to use a capital letter in the answer to the first question unless we tell you to use a lowercase letter for a specific code.

You should get the following printout when you run this program and type "E" and "emphasized" in answer to the questions.

This sample uses ESCape E to produce emphasized printing.

The code to turn off emphasized is ESCape "F".

Double-strike

The other bold mode on the MPS 1000 is double-strike. For this mode the printer prints each line, then moves the paper up slightly and prints the line again. Each dot is printed twice, with the second one slightly below the first as you can see in Figure 5-2.

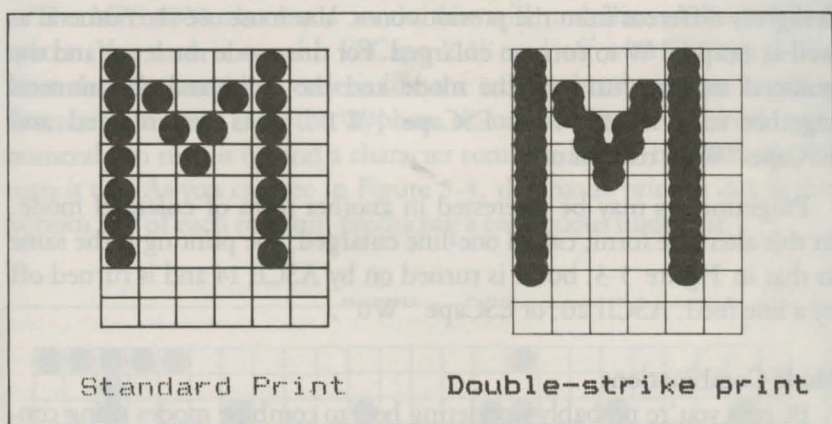


Figure 5-2 Single-strike and double-strike

Unlike emphasized, double-strike combines with any draft pitch (but not with NLQ) because it does not overlap dots horizontally. Since each line in this mode is printed twice, the speed of your printing is slowed. The code for double-strike is ESCape "G". Try it in the master program if you wish. The code to turn off double-strike is ESCape "H".

Enlarged Mode

Perhaps the most dramatic code on the MPS 1000 is enlarged. It produces extra-wide characters that are good for titles and headings. For this mode, the dot pattern of each character is enlarged and a duplicate set of dots is printed one dot to the right. You can see the difference between pica and enlarged pica in Figure 5-3

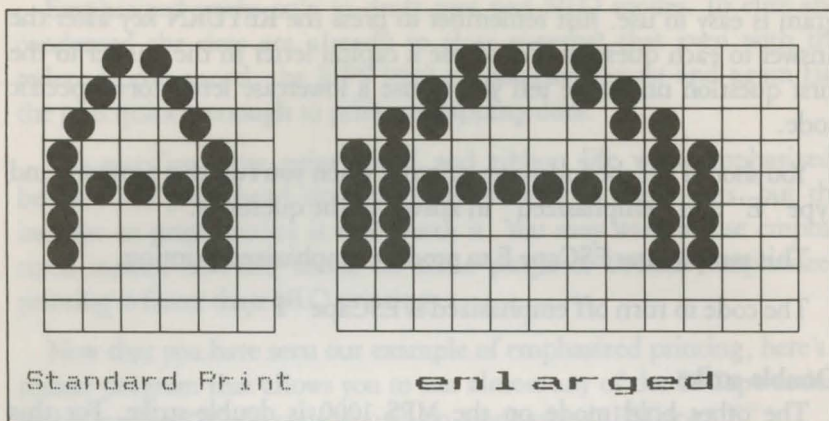


Figure 5-3 Standard and enlarged characters

You can try enlarged yourself by using the code "W1" in the master program. Notice that enlarged mode uses an ESCape code format that is slightly different from the previous ones. You must use the numeral as well as a capital W to turn on enlarged. For this mode the letter and the numeral together turn on the mode and the letter and the numeral together turn it off. Thus ESCape "W1" turns on enlarged and ESCape "W0" turns it off.

Programmers may be interested in another form of enlarged mode. In this alternate form, called one-line enlarged, the printing is the same as that in Figure 5-3, but it is turned on by ASCII 14 and is turned off by a line feed, ASCII 20, or ESCape "W0".

Mode Combinations

By now you're probably wondering how to combine modes using control codes. For example, can a title be made especially vivid by combining enlarged and emphasized? The answer is that you can combine nearly all of the print modes on the MPS 1000. Indeed, your MPS 1000 printer can print such complicated combinations as double-strike emphasized enlarged underlined subscript, although we're not sure you would ever want to use such a combination. The point is, however, the MPS 1000 has the ability to produce almost any combination you can think of; it's up to you to decide which ones you want to use.

To see a few combinations, remove line 90 from the master program. Now run the program once and enter "E" and "emphasized" in response to the questions on the screen. This will give you the same results as the first time you ran the program, but it will leave the printer

in emphasized mode so that you can add another mode. Then run the program again (without turning off the printer). The second time enter "W1" and "emphasized enlarged."

Your printout should be in the typestyle below, showing that the two modes combine with no trouble. You can experiment with other combinations if you wish or you can wait for the section later in this chapter that explains a special ESCape code, Master Select, which allows you to combine as many as seven features with one ESCape code.

emphasized enlarged

When you are through trying combinations, be sure to replace line 90 in the master program so that you can again try one feature at a time.

Underline Mode

The MPS 1000 also has a mode that will underline characters and spaces. You turn it on with ESCape "-1" and off with ESCape "-0". Note that the underline code is like the enlarged code in that it uses a character, in this case the hyphen or minus sign, combined with numeral 1 to turn it on and a character combined with the numeral 0 to turn it off. As you can see in Figure 5-4, this mode prints a dot in the bottom row of each column, producing a continuous underline.

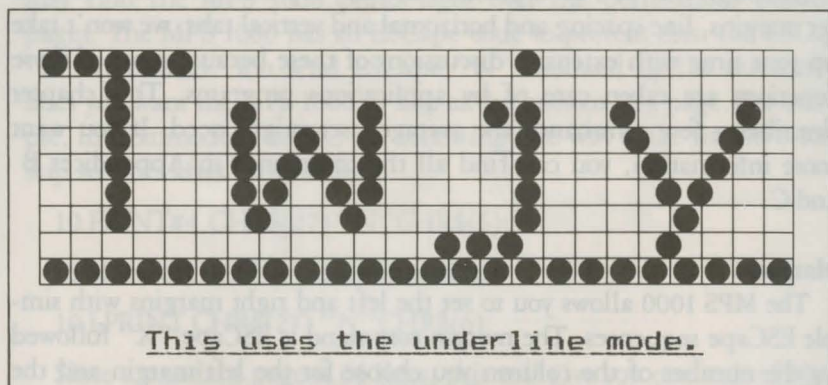


Figure 5-4. Underline mode

As shown in Figure 5-4, the underline mode is continuous, but some word processing and other applications programs produce an underline that leaves spaces between characters as demonstrated in the printout below.

This uses the underline character.

If your software prints this type of underline, it is using the MPS 1000 underline character (ASCII 95), not the underline mode. Because the underline character is only five dots wide, it does not fill the spaces between characters.

Superscript and Subscript

Your MPS 1000 can also print superscripts, which you can use for mathematical formulas, footnotes, and other items that require numbers or letters above or below the usual print line. ESCape "S0" turns on superscript and ESCape "S1" turns on subscript. ESCape "T" turns off either one. You can enter either ESCape "S0" or ESCape "S1" in the master program, but that will print the whole sentence in superscript or subscript. Some more realistic examples are below:

$E=MC^2$

H_2O

This fact is found in three sources.⁷

You can use superscript or subscript for an individual character. Find out how to send printer codes in your applications program; then send the proper codes to the printer.

Page Formatting in IBM Mode

Although the MPS 1000 printer has many sophisticated commands to set margins, line spacing and horizontal and vertical tabs, we won't take up your time with extensive discussions of these because most of these functions are taken care of by applications programs. This chapter describes a few commands the average user might need. If you want more information, you can find all the commands in Appendices B and C.

Margins

The MPS 1000 allows you to set the left and right margins with simple ESCape sequences. The margin command is ESCape "X" followed by the number of the column you choose for the left margin and the right margin.

If your word processing program does not allow you to change the margins, you can send margin commands to your MPS 1000 with BASIC or another programming language before you print your documents. For example, if you prefer wider margins than your word processing program gives you, run the following Commodore BASIC program

before printing. This program gives you a left margin of 10 and a right margin of 60, but you can use any numbers you prefer for the margin commands.

```
5 OPEN4,4
10 PRINT#4, CHR$(27)“X”CHR$(10);CHR$(60)
20 CLOSE4
```

In Microsoft BASIC, the program is:

```
5 LPRINT CHR$(27);“X”;CHR$(10);CHR$(60)
```

Such a program will also allow you to choose the margins you prefer for program listings. Just remember that once you run a program that sets margins, those margins are in effect until you change them with new margin commands or turn off or reset the printer.

You should be aware that a few applications programs reset the printer before each document or file they print. These programs will, of course cancel your new margin settings.

The maximum right margins on the MPS 1000 printer are 80 in pica, 96 in elite, and 137 in condensed. For further information on this command see Appendix B.

Skip Over Perforation

If you are using continuous pin-feed paper for printing program listings or other material not controlled by an applications program, you may find the MPS 1000 prints right over the perforations between pages. The MPS 1000 has an ESCape code to prevent this: the ESCape “N” command. You send ESCape “N” followed by the number of lines you want the MPS 1000 to skip at the bottom of a page. For example, in Commodore BASIC the following line will make the MPS 1000 skip six lines after each 60 lines:

```
10 PRINT#4,CHR$(27)“N”CHR$(6);
```

In Microsoft BASIC, the command is:

```
10 LPRINT CHR$(27)“N”CHR$(6);
```

Since a standard page is 66 lines, this will give you one inch of blank space at the bottom of each page. If you prefer to have half of the blank space at the top of the page and half at the bottom, simply set the top of page approximately three lines (1/2 inch) below the perforation.

Line Spacing

Ordinarily you don't have to worry about how the printer moves the paper so that it doesn't print lines of text on top of each other; the MPS

1000 takes care of this without special instructions. If, however, you want to understand how line spacing works or you need to change the line spacing for a special application such as graphics, you'll want to read this brief explanation.

The movement of the paper between lines is called a line feed. The distance the paper moves is called a line space. In ordinary printing the line spacing is 1/6-inch, which produces six lines of print per inch. Figure 5-5 will help you visualize this spacing. As you can see in the figure, the standard (default) spacing is the same as 12 rows of dots. Since the MPS 1000 characters use nine rows of dots, the 12 dot line spacing leaves three blank rows between the lines of text.

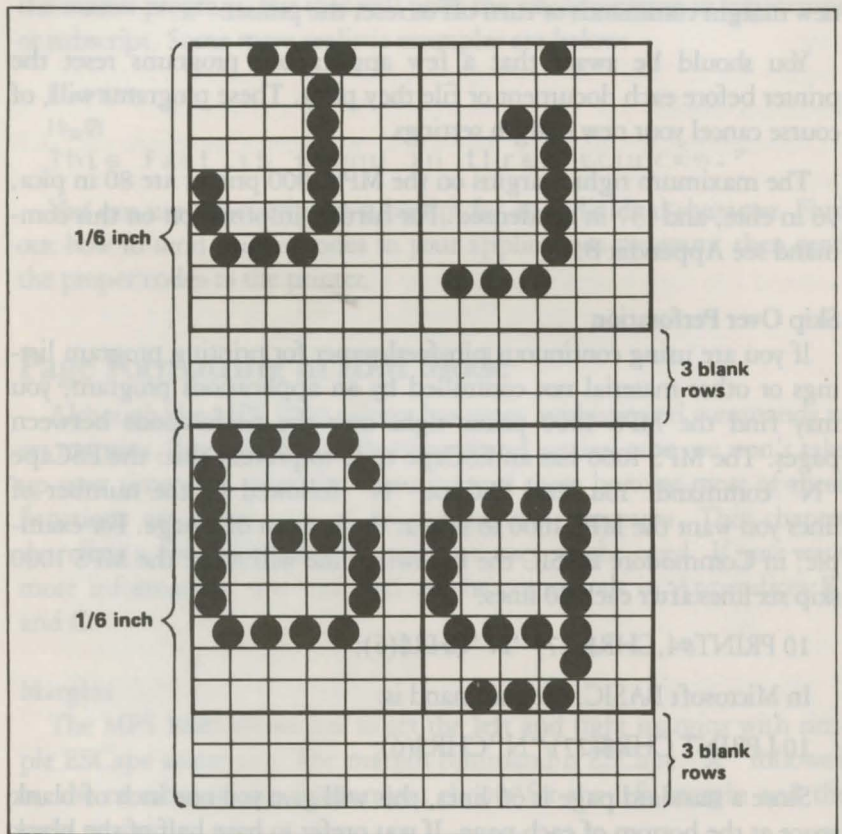


Figure 5-5. Standard line spacing

The default line spacing illustrated in Figure 5-5 is the only one you need for almost all printing of text, but in some cases you may want to increase or decrease the space between lines. The MPS 1000 has several

commands to do this. ESCape "0" decreases the line spacing to 9-dot (9/72-inch), ESCape "1" decreases it to 7-dot, and ESCape "2" sets line spacing to a value defined by ESC "A".

For example, in Commodore BASIC the following lines will set the MPS 1000 line spacing to 13 dot:

```
5 OPEN4,4
10 PRINT#4, CHR$(27);"A";CHR$(13)
20 PRINT#4,CHR$(27);"2"
30 CLOSE4
```

In Microsoft BASIC; the commands are:

```
5 LPRINT CHR$(27); "A"; CHR$(13)
10 LPRINT CHR$(27); "2"
```

In addition, there are commands to specify the line spacing in 72nds of an inch and 216ths of an inch. If you need to make such fine adjustments in the line spacing, see Appendix B for the proper commands. In the chapter on dot graphics you will see how useful ESCape "1" can be.

Paper-out Sensor

Under the platen (the black roller) of your MPS 1000 printer is a small switch that senses whether or not paper is in the printer. When the end of the paper passes this switch, it sends a signal that sounds the beeper and stops your printing. This saves wear on your print head, ribbon and platen, but because of the distance between the switch and the print head, it stops the printing about 2 inches from the end of the page. Therefore, if you use single sheet paper in your MPS 1000 you can't print on the last 2 inches of each page without an adjustment. See Appendix D for deactivating paper-out sensor.

Chapter 6

Dot Graphics in Commodore Mode

Introduction to Dot Graphics

Your MPS 1000 handles dot graphics differently in Commodore and IBM modes. This chapter explains dot graphics in Commodore mode, including creating your own graphic characters and combining lines to form more substantial graphic patterns in bit image graphics. The following chapter reviews comparable graphic topics for IBM mode.

The graphics modes on your printer are different than the text modes. Instead of sending codes for letters and printing functions, you send codes for dot patterns, one number for each column in a line. So rather than utilize the standard letters, numerals and graphic symbols stored in the MPS 1000 ROM (Read Only Memory), the MPS 1000 prints out patterns of dots, column-by-column and line-by-line, in graphics mode. You can plan where you want the dots to appear, and then use a program to have your MPS 1000 print them how and where you planned. This is basically how dot graphics allow your printer to produce pictures, graphs, charts and almost any other illustrations or designs you create.

Because many programs use dot graphics, you can print out many illustrations, charts and designs with your MPS 1000 without any programming, just by following the software instructions. The picture below was created by a graphic software program and printed out with an MPS 1000.

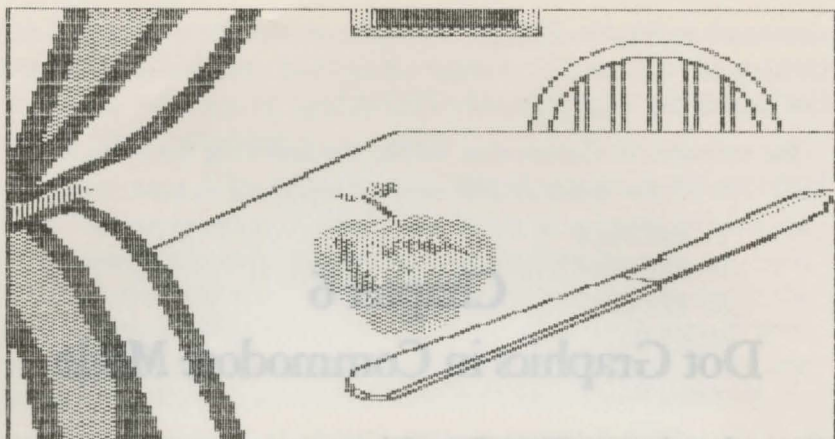


Figure 6-1. Software-generated graphic picture

Some commercial programs perform what is called a screen dump, where whatever is on the screen is sent to the printer and reproduced on paper. If you have a program for a screen dump or software that performs this function, all you really need to know about dot graphic mode is how to use the software. If you want to learn more about programming graphics in graphic mode or understand how the MPS 1000 handles graphics, please read on.

Dot Patterns

Your MPS 1000 forms pictures the same way that pictures in newspapers are printed. If you look closely at pictures in newspapers, you can see they are made up of tiny dots. The MPS 1000 also forms images with patterns of dots, as many as 120 dot positions per inch horizontally and 72 per inch vertically. This is even greater resolution than that of standard newspaper photographs. Careful planning and placement of dots and creation of a program that gives the proper instructions to the printer will enable your MPS 1000 to produce just about any shape or image.

The Print Head

For each column on a print line, the print head produces the pattern of dots you have specified. In Commodore mode, the top seven pins on the print head are used for graphic printing. This pin configuration might make it a little difficult for you to design your own screen dump, but other Commodore printers, such as the VIC 1515/1525 and MPS 801 and MPS 803, have this same seven vertical-dot image so many commercial software programs exist featuring high-resolution graphic screen dumps. If you want to design your own screen dump programs,

there is plenty of information available in books and magazines on the seven-pin vertical print head that applies to the Commodore graphic mode on your MPS 1000 printer.

Bit Image Coding

A bit image is a character programmed onto the 7 X 16 grid that corresponds to the print head of your printer. When the dots are programmed (placed where you want them on the grid), it is printed out by the print head when the pins at the locations you have programmed strike to create dots to form an image.

By using CHR\$(8), you enter the Bit Image graphic printing mode. This allows you to design and print bit image graphics by inputting bit image data. Each DATA statement is made up of numbers that represent a row of dots which, when READ together, make up your bit graphic image. To design a bit image graphic, follow the steps outlined in the example below.

- 1) Use a separate piece of paper to design your bit image graphic.
- 2) Make a dot grid like this, with 7 vertical rows and 16 horizontal columns. The numbering of the vertical rows is extremely important.

	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
1	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
2	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
4	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
8	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
16	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
32	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
64	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○

- 3) Now fill in the dots you want to print to form a bit image, as in the following.

	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
1	○	○	○	○	○	○	●	●	○	○	○	○	○	○	○	○
2	○	○	●	●	●	●	●	●	●	●	●	○	○	○	○	○
4	○	○	●	●	○	○	●	●	○	○	●	○	○	○	○	○
8	●	●	●	●	○	○	●	●	○	○	●	●	●	●	○	○
16	●	●	●	●	●	●	●	●	●	●	●	●	●	●	○	○
32	○	○	●	●	○	○	●	●	○	○	●	●	○	○	○	○
64	○	○	●	●	○	○	●	●	○	○	●	●	○	○	○	○

- 4) Now add together numbers for each column for the rows in which a dot appears. For instance, there are two dots in the first column, one

in the row with a value of 8 and one in the row with 16. The value for that column is then 24. You'll get sixteen column values to use later for your DATA statements for this bit image.

- 5) Now add 128 to each total for each column from step 4. The first column value would now be 152 (24 + 128). These values are now what you will use for DATA statements to produce the bit image.

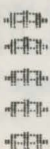
The DATA statements in a program for this bit image:

```
DATA 152,152,254,254,146,146,255,255
DATA 146,146,254,254,152,152,128,128
```

Once you figure out the DATA values for your bit image, you can write a program to print the image. In this example, the image is a 'flying object', which the following program prints five times.

```
10 OPEN 4,4
20 A$ = ""
30 FOR I = 1 TO 16
40 READ A:A$ = A$ + CHR$(A)
50 NEXT I
60 FOR J = 1 TO 5
70 PRINT#4,CHR$(8);A$;CHR$(15)
80 NEXT J
90 FOR K = 1 TO 8:PRINT#4:NEXT K
100 CLOSE 4
110 END
200 DATA 152,152,254,254,146,146,255,255
210 DATA 146,146,254,254,152,152,128,128
```

This control code defines 1/6" line feed for a separate printed line.

Result: 

Combining Bit Images to Form a Larger Image

You can create images larger than the seven row grid allows by combining grids. The following program provides an example of how you can design a larger character by using DATA statements to place three programmed bit images together to form a larger character.


```
10 OPEN 4,4
20 FOR R = 1 TO 3
30 FOR C = 1 TO 16
40 READ A:A$(R) = A$(R) + CHR$(A)
50 NEXT C
```



```

60 NEXT R
70 PRINT#4, CHR$(8);
80 FOR R = 1 TO 3
90 PRINT#4, A$(R)
100 NEXT R
110 PRINT#4, CHR$(15)
120 FOR L = 1 TO 8:PRINT#4:NEXT L
130 CLOSE 4
140 END
200 DATA 129,129,250,250,158,158,255,255
210 DATA 158,158,250,250,129,129,128,128
220 DATA 128,128,159,159,243,243,179,179
230 DATA 243,243,159,159,128,128,128,128
240 DATA 128,128,130,130,131,131,128,128
250 DATA 131,131,130,130,128,128,128,128

```

Result: 

All Bit Pattern Printing

All bit pattern printing works along the same principles as bit image printing. An all bit pattern is a pattern created by placing each dot individually, with the dots forming a pattern when a program designed to display the pattern is RUN. The following program demonstrates bit pattern printing in the Commodore dot graphics mode.

```

10 OPEN 4,4
20 A$ = ""
30 FOR I = 0 TO 127
40 A$ = A$ + CHR$(I + 127)
50 NEXT I
60 B$ = ""
70 FOR J = 127 TO 0 STEP -1
80 B$ = B$ + CHR$(J + 127)
90 NEXT J
100 FOR K = 1 TO 3
110 PRINT#4, CHR$(8); A$; CHR$(15); " ALL BIT PATTERN 1"
120 NEXT K
130 PRINT#4
140 FOR K = 1 TO 3
150 PRINT#4, CHR$(8); B$; CHR$(15); " ALL BIT PATTERN 2"
160 NEXT K
170 FOR M = 1 TO 8:PRINT#4:NEXT M
180 CLOSE 4

```

Result:

```
ALL BIT PATTERN 1
ALL BIT PATTERN 1
ALL BIT PATTERN 1

ALL BIT PATTERN 2
ALL BIT PATTERN 2
ALL BIT PATTERN 2
```

Repeat Bit Image Printing

Often, you'll need to reproduce the same line of bit image DATA for parts of images. The CHR\$(26) sequence within CHR\$(8) mode enables you to repeat printing a bit image quite easily. The format is as follows:

```
CHR$(8). . . . .CHR$(26);CHR$(n);CHR$(bit image data)
```

"n" is a binary number (from 0 to 255) which specifies the number of repetitions to be printed, and the "bit image data" is the DATA statement for the bit image that is being repeated. If you supply a value of zero for n, it is READ as 256, meaning the bit image will be reprinted 256 times. In order to repeat more than 256 times, you must use this sequence more than once.

The following program demonstrates how this sequence works:

```
10 OPEN 4,4
20 FOR I = 1 TO 6
30 READ N
40 PRINT#4,CHR$(8);CHR$(26);CHR$(N);CHR$(243);CHR$(15)
50 NEXT I
60 FOR J = 1 TO 8:PRINT#4:NEXT J
70 CLOSE 4
100 DATA 34,57,75,89,123,186
```

Result:

```
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

Dot Address Determination

You won't always want to place characters at the left margin of the paper. The dot address determination sequence lets you specify where on a line you want a bit image character (or anything else) to be printed. The code sequence for this is:

```
CHR$(27);CHR$(16);CHR$(nH);CHR$(nL)
```

where "nH" and "nL" are dot positions on the line where printing is to start. These values are numbers ranging from 0 to 480. When a number greater than 480 is specified, printing starts at the beginning of the next line. This program positions the "alien" bit image character 300 dots into the page. Note that the only difference between this and the program earlier in this chapter is line 90.

```
10 OPEN 4,4
20 FOR R = 1 TO 3
30 FOR C = 1 TO 16
40 READ A:A$(R) = A$(R) + CHR$(A)
50 NEXT C
60 NEXT R
70 PRINT#4, CHR$(8);
80 FOR R = 1 TO 3
90 PRINT#4,CHR$(27);CHR$(16);CHR$(1);CHR$(44);A$(R)
100 NEXT R
110 PRINT#4,CHR$(15)
120 FOR L = 1 TO 8:PRINT#4:NEXT L
130 CLOSE 4
140 END
200 DATA 129,129,250,250,158,158,255,255
210 DATA 158,158,250,250,129,129,128,128
220 DATA 128,128,159,159,243,243,179,179
230 DATA 243,243,159,159,128,128,128,128
240 DATA 128,128,130,130,131,131,128,128
250 DATA 131,131,130,130,128,128,128,128
```

Result:



← 300 dots →

Double Density Mode

All of the examples so far in this chapter have been in the single density bit image mode (480 dots per line). However, you can use double density (960 dots per line) to print graphics in Commodore mode. You can activate double density with the code `CHR$(2)`. To cancel double density, use the code `CHR$(1)`, which returns you to single density.

Adding this line to any of the programs in this chapter will allow you to see double density bit image graphics:

```
15 PRINT#4,CHR$(2)
```

Again, to return to single density, use `CHR$(1)`.

Chapter 7

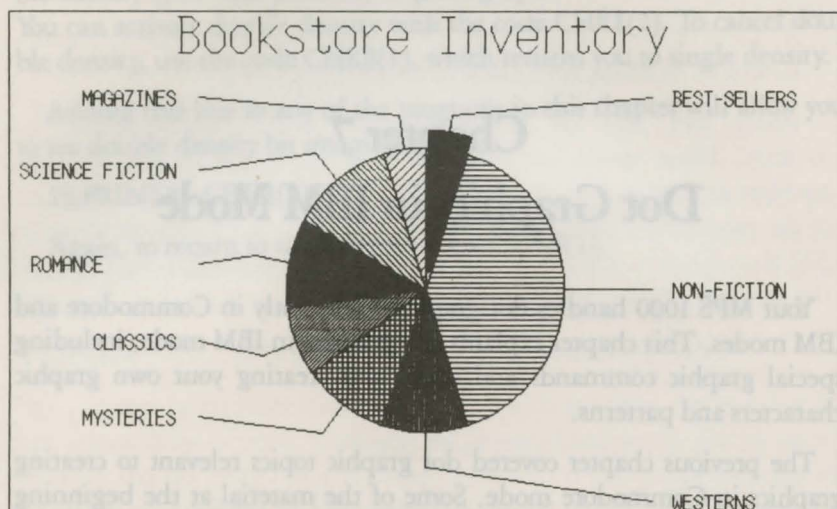
Dot Graphics in IBM Mode

Your MPS 1000 handles dot graphics differently in Commodore and IBM modes. This chapter explains dot graphics in IBM mode, including special graphic commands and codes and creating your own graphic characters and patterns.

The previous chapter covered dot graphic topics relevant to creating graphics in Commodore mode. Some of the material at the beginning of this chapter is repeated for those users who did not need to read about graphics in Commodore mode. Nearly all of the material provided in this chapter pertains only to dot graphics in IBM mode, beginning with "The Print Head."



Because many software programs can use dot graphics, you may be able to print graphics like the ones on this page and the next by simply giving your software a few instructions. The chart, for example, was created and printed in about 10 minutes.



Some commercial software programs perform what is called a screen dump, in which whatever is on the computer screen is sent to, and printed by the printer. This function is sometimes accomplished by a print screen command. If you have a program for screen dumps or any other software that produces graphics, all you need to know about dot graphics is how to use the software. If, on the other hand, you wish to do your own programming or to understand how the MPS 1000 prints graphics, read on.

The Print Head

The IBM graphics mode on the MPS 1000 is different from text modes. Instead of sending codes for letters and printing functions, you send codes for dot patterns, one number for each column in a line. Since none of the predefined characters or symbols in the printer's memory is used, your program controls where each dot is printed.

For each column on a print line, the print head prints the pattern of dots you have specified. In standard graphics mode it uses only the top eight pins on the print head because the computer uses eight data lines to communicate with the printer. Therefore, each of the top eight pins of the print head corresponds to one of the data lines.

To print figures taller than eight dots, the print head makes more than one pass. It prints one line, then advances the paper and prints another, just as it does with text. To keep the print head from leaving gaps between graphics lines as it does between text lines, the line spacing must be adjusted to eliminate space between lines. When the line spacing is properly adjusted, the MPS 1000 prints finely detailed graphics images that give no indication that they are made up of separate lines, each no more than 1/8 of an inch wide.

To insure the proper alignment of dots in figures that use more than one pass of the print head, the MPS 1000 abandons the bidirectional printing it uses for draft text. Instead it prints graphics from left to right only.

Each pass of the print head contains one piece of the total pattern, which can be as tall or short and as wide or narrow as you desire. You don't have to fill the whole page or even an entire line with your graphics figures. In fact, you can use as little or as much space as you like for a figure and put it anywhere on the page.

IBM Dot Graphics Mode

The graphics mode command is different from the other commands covered so far in this manual. For most of the other MPS 1000 modes, such as condensed and emphasized, one ESCape code turns the mode on and another turns it off. For graphics, the command is more complicated because the code that turns on a graphics mode also specifies how many columns it will use.

The MPS 1000 has several different graphics densities, but we'll keep things simple by using only one in the first exercises. The code for entering single-density graphics mode is ESCape "K" n1 n2. In Commodore BASIC the command is given in this format:

```
PRINT#4, CHR$(27) "K" CHR$(N1) CHR$(N2);
```

In Microsoft BASIC, the command is given in this format:

```
LPRINT CHR$(27) "K" CHR$(N1) CHR$(N2);
```

ESCape "K" specifies single-density graphics, and the next two numbers (n1 and n2) specify the number of columns reserved for graphics.

The graphics command requires more than one number to specify how many columns to reserve because as many as 1,920 columns are possible in graphics printing. Since the MPS 1000 uses eight-digit

binary numbers, it can't accept decimal numbers larger than 255. Therefore, the graphics mode command uses two numbers for reserving columns.

The number of columns reserved is the first number plus 256 times the second number. Since the command is set up for two numbers, you must supply two even if you only need one. When you need less than 256 columns, just make n1 the number of columns you are reserving and make n2 = 0.

Pin Labels

Once you put the printer into graphics mode and reserve the number of columns you want, your next step is to tell the print head which pins to fire in each column. There are 256 possible combinations of eight pins, and you send only the one number for each column. The numbering system that allows you to use a single number to specify which of the 256 possible patterns you want is shown in Figure 7-1.

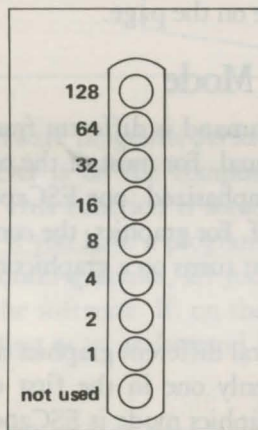


Figure 7-1. Pin labels

To fire any one pin, you send its number. To fire more than one pin at the same time, add up the numbers of the pins and send the sum to the printer. With these labels for the pins, you fire the top pin by sending 128. To fire the bottom graphics pin, you send 1. If you want to fire the top and bottom pins, simply add 128 and 1, then send 129.

By adding the appropriate label numbers together, you can fire any combination of pins you want. Figure 7-2 show you three examples of how to calculate the number that will fire a particular pattern of pins.

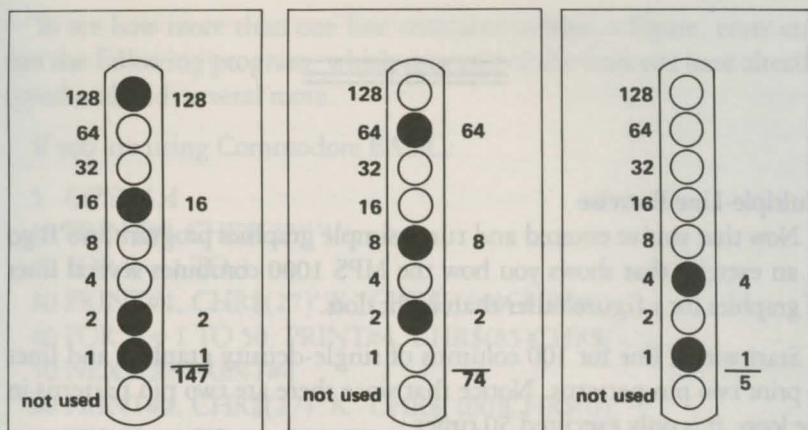


Figure 7-2. Calculating numbers for pin patterns.

With this numbering system any combination of the eight pins adds up to a unique decimal number between 0 and 255. Although you use a decimal number, it is, of course converted to a binary number before it goes to the printer.

Now that you know the principles of MPS 1000's graphics, we'll give you two simple exercises, more densities, and then something more complex as a basis for writing your own programs.

First Graphics Program

We could give you a program that prints a single column of dots. But it's difficult to see the pattern of a single column of dots, so your first graphics program prints the same pattern 40 times.

The first line is the code for 40 columns of single-density graphics. Our example is in Commodore BASIC, but you can adapt it to the programming language you prefer. After the OPEN file statement (10 OPEN 4,4), the next line is the code for 40 columns of single-density graphics.

```
20 PRINT#4, CHR$(27)“K”CHR$(40)CHR$(0);
```

The following line is the data printed as pin patterns. Be sure you type in the semi-colons in both lines:

```
30 FOR X = 1 TO 40: PRINT#4, CHR$(74);:NEXT X
```

Then close the file with 40 CLOSE 4.

That's it. Run the program to see the result. This allows you to see exactly how the mode works.

Multiple-Line Exercise

Now that you've entered and run a simple graphics program, we'll go to an exercise that shows you how the MPS 1000 combines several lines of graphics for a figure taller than eight dots.

Start with a line for 100 columns of single-density graphics and lines to print two pin patterns. Notice that since there are two pin patterns in the loop, it is only executed 50 times.

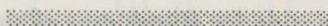
If you are using Commodore BASIC:

```
10 OPEN 4,4
20 PRINT#4, CHR$(27) "K" CHR$(100)CHR$(0)
30 FOR X = 1 TO 50: PRINT#4, CHR$(85)CHR$(42);
40 NEXT X
50 CLOSE 4
```

If you are using Microsoft BASIC:

```
30 LPRINT CHR$(27)"K"CHR$(100)CHR$(0)
40 FOR X = 1 TO 50: LPRINT CHR$(85)CHR$(42);
50 NEXT X
```

If you run the program now, you'll see how one line of the pattern looks:



To see how more than one line combines to form a figure, enter and run the following program, which uses two of the lines you have already typed and adds several more.

If you are using Commodore BASIC:

```
5 OPEN4,4
10 PRINT#4,CHR$(27)"1";
20 FOR R = 1 TO 3
30 PRINT#4, CHR$(27)"K"CHR$(100)CHR$(0);
40 FOR X = 1 TO 50: PRINT#4, CHR$(85)CHR$(42);
50 NEXT X: PRINT#4
60 PRINT#4, CHR$(27)"K"CHR$(100)CHR$(0);
70 FOR X = 1 TO 50: PRINT#4, CHR$(42)CHR$(85);
80 NEXT X: PRINT#4: NEXT R
90 PRINT#4, CHR$(27)"@"
100 CLOSE4
```

If you are using Microsoft BASIC:

```
10 LPRINT CHR$(27)"1";
20 FOR R = 1 TO 3
30 LPRINT CHR$(27)"K"CHR$(100)CHR$(0);
40 FOR X = 1 TO 50: LPRINT CHR$(85)CHR$(42);
50 NEXT X: LPRINT
60 LPRINT CHR$(27)"K"CHR$(100)CHR$(0);
70 FOR X = 1 TO 50: LPRINT CHR$(42)CHR$(85);
80 NEXT X: LPRINT: NEXT R
90 LPRINT CHR$(27)"@"
```

Now run the program to see the six print lines combine into a pattern:



This short and simple program that produced the pattern demonstrates many elements of graphics programming. Therefore, we'll explain each line. Line 10 changes the line spacing to 7-dot, which is the height of the dot patterns used in the program. Therefore, there is no space between the print lines.

Line 20 begins a loop to produce multiple print ines. Lines 30 and 40 were covered above, but an PRINT#4, or LPRINT is added to line 50 to produce a line feed after line 40. Lines 60 and 70 are like lines 30 and 40 except that line 70 uses a reversal of the patterns in line 40. As the loop is executed, the program prints lines 40 and 70 alternately so that the patterns of the print lines will fit together well.

Notice that the graphics command can be in effect for only one print line. The command is in lines 30 and 60 so that it is issued each time a new print line is begun.

Line 90 is the reset code to return the printer to its defaults.

Density Varieties

Although all the examples so far in this chapter have been in the single-density graphics mode, the MPS 1000 offers three other density modes. All the eight-pin densities and their commands are described in Table 7-1.

Table 7-1. Graphics modes

Density	ESCAPE Sequence	Description
Single	ESC"K"CHR\$(n')CHR\$(n')	60 dots/inch; 480 dots/8" line
Double at half speed	ESC"L"CHR\$(n')CHR\$(n')	120 dots/inch; 960 dots/8" line
Double at normal speed	ESC"Y"CHR\$(n')CHR\$(n')	120 dot positions/inch. Faster than Mode 1; does not print consecutive dots in a row.
Quadruple	ESC"Z"CHR\$(n')CHR\$(n')	240 dot positions/inch; 1920 dot positions/8" line. Does not print consecutive dots in a row.

Column Reservation Numbers

Now that we've introduced you to the rest of the eight-pin graphics densities and the reassigning code, we'll explain in more detail the part of the graphics command that reserves the number of columns for graphics (the numbers n1 and n2 in our examples).

If you need fewer than 255 columns of graphics, n1 is the number of columns you want and n2 is 0. As you can see in Table 7-1, however, a single line will hold as many as 1,920 columns in quadruple-density. Specifying more than 255 is where the second number slot (n2) fits in. The first number that you send (n1), indicates a number of columns, but the second does not represent a number of columns; it is multiplied by 256 and added to n1. The command for the maximum number of dots you can reserve on the MPS 1000 is:

```
CHR$(27)“Z”CHR$(128)CHR$(7);
```

Once you have chosen the number of columns you want to use, you can have your program do the calculations for you with the following format:

```
CHR$(27)“L”CHR$(N MOD 256)CHR$(INT(N/256));
```

N is the total number of columns you want to specify. The MOD (modulus) function calculates the value for n1, and the INT (integer) function calculates the value for n2. For programming languages other than BASIC, consult your manual for the proper form for these functions.

This format can be used with any graphics density and with any value of N up to the maximum number of columns per line for that density.

WIDTH Statements

Many computer systems automatically insert the control codes for a carriage return and a line feed after every 80 characters. This insertion is usually no problem with text, but it can spoil your graphics. In the graphics mode they may insert the control codes after 120 columns, which could be in the middle of a line.

You can usually prevent these unwanted control codes with a WIDTH statement such as the one below:

```
WIDTH “LPT1:”, 255
```

The format for your system may be different. Consult your computer or computer language manual to find the correct format for your system. Then put a WIDTH statement in one of the first lines of all your graphics programs. It is easier to put a WIDTH statement in all but the simplest of your programs than to examine each one to see whether or not such a statement is necessary.

Design Your Own Graphics

In this section we take you through the development of a graphics program. The example is not especially complicated, but it includes the same steps you would use for a more complex figure so that you have the basis for designing graphics on your MPS 1000.

You will plan your figure with dots on graph paper, but before beginning to place the dots, you should decide which graphics density you want. Figure 7-3 shows the difference between the three most-used graphics modes so you can choose the one you want.

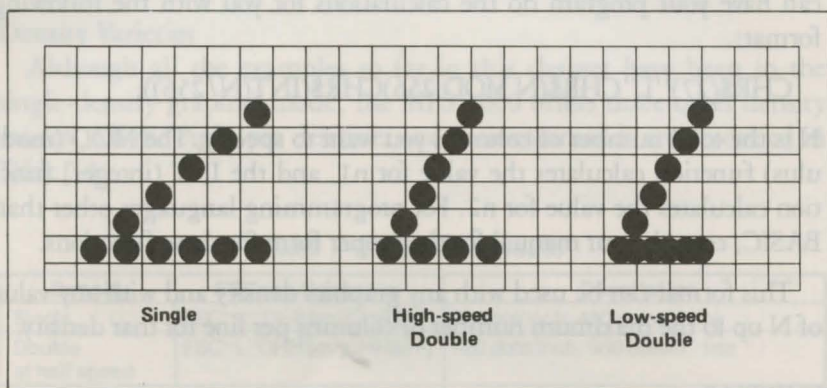


Figure 7-3. Designing in different densities

In this figure you can see the main rules for graphics design in the three densities. In single density, no dots can be placed on vertical lines. In high-speed double density, dots can be placed on vertical lines, but no dots can overlap. In low-speed double density, dots can be placed on vertical lines and they can overlap.

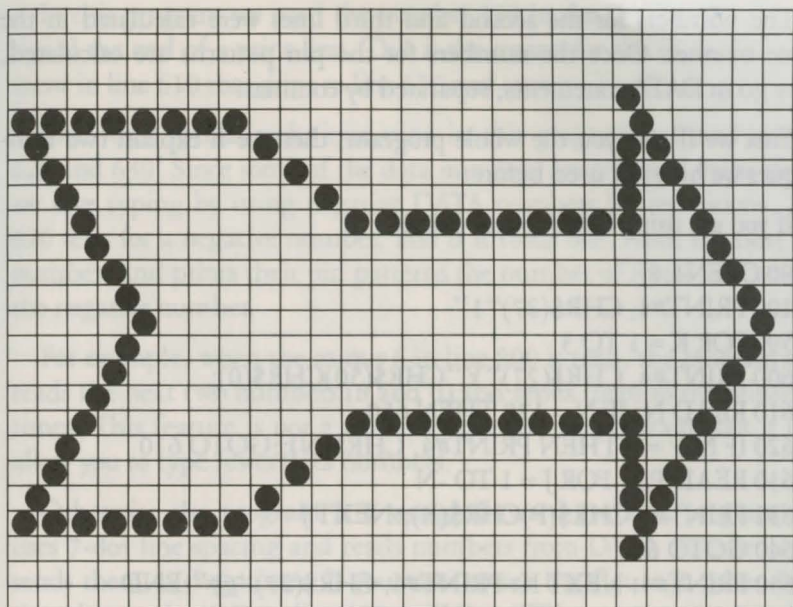


Figure 7-4. Arrow design

After plotting all the dots as in Figure 7-4, you calculate the numbers for each pin pattern by dividing the design grid into separate print lines. For the arrow design the grid was divided into three lines, each seven dots high. Then each column was examined and the sums of the pin values determined. This process for the first line is shown in Figure 7-5. The pin values are on the left side and the sums are at the bottom of each column.

Those of you who have read the previous chapter will see that designing graphics is much like designing user-defined characters.

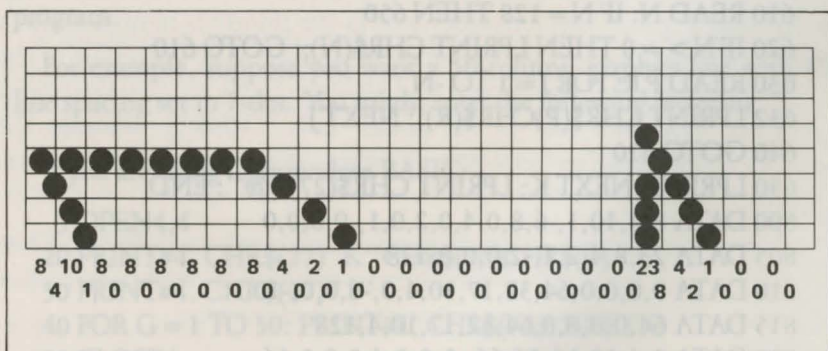


Figure 7-5. First line of arrow figure

The numbers for the second and third lines were calculated in the same manner. Once the numbers for the pin patterns are calculated, they go in DATA statements, separated by commas.

First we'll give you the whole program; then we'll explain two techniques we have not used before:

If you are using Commodore BASIC:

```
80 OPEN4,4
100 PRINT#4, CHR$(27)"1"
590 FOR K = 1 TO 3
600 PRINT#4, CHR$(27)"Y"CHR$(50)CHR$(0);
610 READ N: IF N = 128 THEN 650
620 IF N >= 0 THEN PRINT#4, CHR$(N);:GOTO 610
630 READ P,R: FOR J = 1 TO -N
632 PRINT#4,CHR$(P)CHR$(R);:NEXT J
640 GOTO 610
650 PRINT#4: NEXT K: PRINT#4, CHR$(27)"@" :END
800 DATA 8,4,10,1,-6,8,0,4,0,2,0,1,-9,0,0,0
805 DATA 23,8,4,2,1,-2,0,0,0,128
810 DATA 0,0,0,0,64,32,17,10,4,0,-6,0,0,-10
815 DATA 64,0,0,0,0,64,32,17,10,4,128
820 DATA 2,4,10,16,34,64,-5,2,0,4,0,8,0,16
825 DATA 0,-9,32,0,61,2,4,8,16,32,64,0,0,0,128
850 CLOSE4
```

If you are using Microsoft BASIC:

```
90 WIDTH LPRINT 255
100 LPRINT CHR$(27)"1"
590 FOR K = 1 TO 3
600 LPRINT CHR$(27)"Y"CHR$(50)CHR$(0);
610 READ N: IF N = 128 THEN 650
620 IF N >= 0 THEN LPRINT CHR$(N);: GOTO 610
630 READ P,R: FOR J = 1 TO -N
632 LPRINT CHR$(P)CHR$(R);: NEXT J
640 GOTO 610
650 LPRINT: NEXT K: LPRINT CHR$(27)"@" :END
800 DATA 8,4,10,1,-6,8,0,4,0,2,0,1,-9,0,0,0
805 DATA 23,8,4,2,1,-2,0,0,0,128
810 DATA 0,0,0,0,64,32,17,10,4,0,-6,0,0,-10
815 DATA 64,0,0,0,0,64,32,17,10,4,128
820 DATA 2,4,10,16,34,64,-5,2,0,4,0,8,0,16
825 DATA 0,-9,32,0,61,2,4,8,16,32,64,0,0,0,128
```



In this program we used the number 128 in the DATA statements to signal the end of a print line. This is the reason for the IF-THEN statement in line 610 that skips to line 650 and causes a line feed.

The other special technique used in this program is found in lines 620 and 630. Since some of the data numbers are repeated many times, we save typing by using negative DATA numbers for repetitions. Line 620 tests for a negative number, and if it finds one, reads the next two numbers and prints their pin patterns the number of times indicated by the negative number.

For example, when the minus 6 in line 800 is read, the program then reads the next two numbers (8 and 0) and sends them to the printer six times. This feature is not a necessary part of the program, but it does allow you to type fewer data numbers.

Otherwise the program is a straightforward graphics program that uses 7-dot line spacing and reads numbers from DATA statements and sends them to the printer. If you want to see the figure in other densities, change the "Y" in line 600 to "L" or "Z".

Graphics Programming Tips

Now that we've shown you how to design your own graphics, we'll review and emphasize a few elements of graphics programming. As usual, we use BASIC in the examples, but the principles apply to any programming language.

Semicolons and command placement

After the graphics command is issued, every number sent to the MPS 1000 is interpreted as a pin pattern and printed on paper. Therefore, you must be careful where you put graphics commands in your program.

For example, suppose you want a 50-column graphics line with the line spacing set to 7-dot. You might enter the following program:

If you are using Commodore BASIC:

```
5 OPEN#4,4
20 PRINT#4, CHR$(27)"K"CHR$(50)CHR$(0)
30 PRINT#4, CHR$(27)"1"
40 FOR G = 1 TO 50: PRINT#4, CHR$(74): NEXT G
50 CLOSE#4
```


If you are using Microsoft BASIC:

```
20 LPRINT CHR$(27)“K”CHR$(50)CHR$(0)
30 LPRINT CHR$(27)“1”
40 FOR G = 1 TO 50: LPRINT CHR$(74): NEXT G
```

This program has all the necessary elements. Line 20 has the command for single-density graphics and specifies 50 columns. (Remember that you must use two numbers to reserve columns even if you only need the first one.) Line 30 has the correct command for 7-dot line spacing, and line 40 calls for the printing of a pin pattern 50 times. (If you wish, refer back to Figure 7-2 to see a representation of the pin pattern that 74 produces.)

Although this program has all the necessary elements, it will not give you the single pin pattern that you want, as you can see in the partial printout in Figure 7-6.

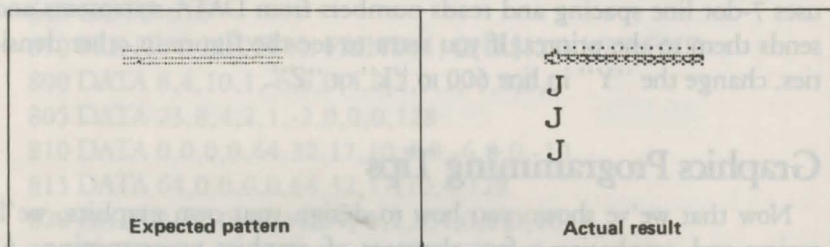


Figure 7-6. Result of incorrect program

What went wrong? To help you understand the graphics command and avoid some of the more common errors made with it, we'll look at this program in detail.

First look at line 20. ESCape “K” calls for single-density graphics, and the two CHR\$ numbers specify 50 columns of dots. Once that command is given, every number sent to the printer is interpreted as a pin pattern and printed on the paper. Since there is no semicolon at the end of the line, the numbers 13 and 10—the codes for carriage return and line feed—are sent to the printer after CHR\$(0). Because the graphics command has been issued, these codes are printed as pin patterns.

In line 40 nothing is sent to the printer until after the LPRINT. Then the desired pin pattern-74-is finally sent, but since no semicolon is after it, 13 and 10 are sent next each time the loop is executed.

Figure 7-7 is an enlarged representation of the first 14 columns of the printout. In this figure you can see exactly how the printer reacted to the first part of the incorrect program.

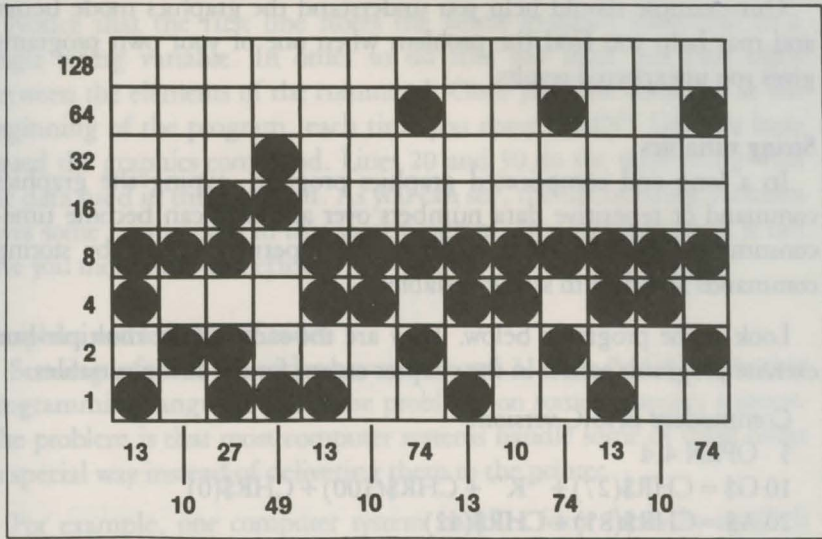


Figure 7-7. Pin patterns of incorrect program

You may also wonder why the program prints not only the different pin patterns but also the character "J" a number of times. The reason lies in the number of columns you reserved with the graphics command. After the MPS 1000 receives all the numbers reserved by a graphics command-50 in this case-it leaves the graphics mode and resumes interpreting numbers as printable characters or print commands.

Since the incorrect program has sent many extra numbers, mainly 10s and 13s, the 50 columns reserved are filled before the loop in line 40 has been executed 50 times. Therefore, during the last passes of the loop the MPS 1000 interprets CHR\$(74) as the ASCII code for "J" and prints that character each of the last 35 passes of the loop.

If you want to make the program work correctly, put the line-spacing command in line 10, delete line 30, and add two semicolons: one at the end of line 20 and one between CHR\$(74) and the colon in line 40.

We have explained this incorrect program in detail so that you will remember two important tips about using the graphics command:

- Use semicolons to prevent the MPS 1000 from printing carriage return and line feed codes as pin patterns.
- Do not put any other commands between the graphics command and its data.

Our example should help you understand the graphics mode better and may help you find the problem when one of your own programs gives you unexpected results.

String variables

In a long and complicated graphics program, typing the graphics command or repetitive data numbers over and over can become time-consuming. You can avoid much of the repetitive typing by storing commands and data in string variables.

Look at the programs below. They are the same as the multiple-line exercise programs earlier in the chapter except for the string variables.

Commodore BASIC version:

```
5 OPEN 4,4
10 G$ = CHR$(27) + "K" + CHR$(100) + CHR$(0)
20 A$ = CHR$(85) + CHR$(42)
30 B$ = CHR$(42) + CHR$(85)
35 PRINT #4, CHR$(27) "1"
40 FOR R = 1 TO 3
50 PRINT#4, G$;
60 FOR X = 1 TO 50:PRINT#4, A$;
70 NEXT X:PRINT#4
80 PRINT#4, G$;
90 FOR X = 1 TO 50:PRINT#4, B$;
100 NEXT X:PRINT#4:NEXT R
110 PRINT#4, CHR$(27) "@ "
120 CLOSE 4
```

MBASIC version:

```
10 G$ = CHR$(27) + "K" + CHR$(100) + CHR$(0)
20 A$ = CHR$(85) + CHR$(42)
30 B$ = CHR$(42) + CHR$(85)
40 LPRINT CHR$(27) "1"
50 FOR R = 1 TO 3
60 LPRINT G$;
70 FOR X = 1 TO 50: LPRINT A$;: NEXT X
80 LPRINT
90 LPRINT G$;
100 FOR X = 1 TO 50: LPRINT B$;: NEXT X
110 LPRINT: NEXT R
120 LPRINT CHR$(27) "@ "
```

Notice that the first line stores the whole graphics command in a single string variable. In order to do this you must put plus signs between the elements of the command. Once you have done this at the beginning of the program, each time you enter `LPRINT G$`; you have issued the graphics command. Lines 20 and 30 do the same thing with the data used in this program. As you can see, the use of string variables saves some typing even in this short program. In a long program, it can save you more time and effort.

Graphics and low ASCII codes

Sending a few of the codes between 0 and 31 with BASIC or another programming language can cause problems on some computer systems. The problem is that most computer systems handle some of these codes in special way instead of delivering them to the printer.

For example, one computer system handles form feeds by itself. It counts lines to keep track of the top of page. If a program sends the ASCII code for form feed (12) to the printer, the computer system intercepts it and sends instead the code for line feed (10) several times. If you are in a graphics mode with this system and send a 12 to fire pins 3 and 4, the computer system will intercept that 12 and send several 10s instead. You can see how this would ruin your graphics. You would get pins 2 and 4 (whose sum is 10) several times instead of pins 3 and 4 only once.

If your computer system screens out or changes any codes sent to it, you can often design around these problems by using other numbers with similar patterns.

Chapter 8

Troubleshooting

This chapter approaches troubleshooting from several directions. The first section uses a columnar format to match solutions with problems. Other sections cover beeper error warnings, coding, and specific solutions for popular personal computer systems.

Problem / Solution Chart

Possible problems are listed on the left and solutions on the right.

PROBLEM

Setting print styles

Can't get condensed print

Tabbing

Horizontal tabs don't work

Horizontal tabs are incorrect when changing pitch.

Graphics

Strange things print.

SOLUTION

Cancel emphasized; it has priority over condensed.

Set margins before tabs, not after

Tabs are set according to current print pitch. Changes in pitch do not affect the position of the tabs on the page.

Some systems require a WIDTH statement. See your system documentation.

Many computers have problems sending one or more of the codes between 0 and 13. Avoid any that

affect your system if possible. You can also POKE the problem codes directly to the printer port.

Seven-bit computers cannot use the eighth pin (128). If you have a seven-bit computer and any of your graphics data numbers are larger than 127, change the numbers.

Be sure that no other commands or carriage returns come between the graphics command and its data. See Chapter 6 or 7.

Printer freezes in graphics mode.

The printer expects a certain number of pin patters, determined by n1 and n2. It will wait patiently until the quota is full. Note that nine-pin graphics mode requires two bytes for each column of graphics.

Can't get a full page in width

Some systems require a WIDTH statement. See your system documentation.

Seven-bit computers are limited to width of 0-127, 256-383, 512-639, etc. See seven-bit graphics subsection.

Problems with paper feeding or irregular darkness of printing

If a self-adhesive label comes off of the backing, it may stick behind the platen and cause problems with paper feeding and printing. If this happens take your MPS 1000 to a qualified service person; do not attempt to remove the label yourself.

Paper-out sensor

Can't deactivate paper-out sensor with DIP switch 1-7 or ESCape "8".

Computer systems that monitor printer cable pin 12 will ignore both ESCape "8" and the setting of switch 1-7. These systems will

stop the printing when no paper is in contact with the paper-out sensor (a small switch located beneath the platen). Certain printer cables are designed to overcome this problem, or you can tape a business card over the switch. Be sure to remove the card if you use a cut-sheet feeder, however.

Beeper Error Warnings

When the MPS 1000 beeper sounds, it usually indicates that the printer is out of paper. The beeper can also be sounded by any program that sends the ASCII 7 code and by certain error conditions in the printer itself.

If the printer beeps and stops printing when it is not out of paper, turn the printer off and check to see if the paper is loaded correctly. If the paper is loaded correctly, turn the printer back on and try to print again. If the printer beeps and does not print again, take it to a qualified service person.

Specific Computer Problems

IBM-PC solutions

There are two problems in using the IBM Personal Computer BASIC to drive a printer. First, the IBM-PC BASIC inserts a carriage-return/line-feed (CR-LF) after each 80 characters you send it. Second, it adds an LF to each CR in an LPRINT statement.

Here is the way to adjust the width when it is the only problem. Tell the computer that the print line is wider than 80 characters with this WIDTH statement:

```
WIDTH "LPT1:",255
```

The 255 is a special number that prevents the computer system from inserting a CR-LF into the line, unless there's one in your program.

The extra line feed—CHR\$(10)—that accompanies each carriage return—CHR\$(13)—is no problem except when you need to use CHR\$(13) in a graphics program. Getting rid of the extra CHR\$(10) is rather complicated. First you open the printer as a random file:

```
OPEN "LPT1:" AS #1
```

Although this allows you to send any code to the printer, you can no longer use the LPRINT command. Instead, you must use a PRINT #1 command:

```
PRINT #1, "Now I can print anything"
```

This does allow you to print anything, but it ignores any previous WIDTH statements.

If you want to print more than 80 characters per line in a graphics program, you must therefore change your opening statement to include the appropriate WIDTH statement:

```
OPEN "LPT1:" AS #1 : WIDTH #1, 255
```

And for the MBASIC programs in this manual, don't forget to use PRINT #1 instead of LPRINT.

This won't work for those of you who have the original release of the Disk Operating System (DOS 1.0). It can't run a printer like a file. IBM has, however, issued a free update (DOS 1.05); take a disk to your dealer to get your copy.

Another printer problem with DOS 1.0 is that it doesn't send CHR\$(7) to the printer; it just rings the computer's bell. This has also been corrected in subsequent versions.

Appendix A

MPS 1000 Character Fonts, PETASCII and IBM Extended ASCII Codes

This appendix contains a number of tables showing the characters which the MPS 1000 can produce.

The tables for Commodore mode (PETASCII) show the decimal and hexadecimal value and the characters they produce.

Separate tables show the Commodore character sets for:

- Uppercase/graphic (cursor up) mode
- Lowercase/uppercase (cursor down) mode

The tables for IBM mode show the high and low byte hexadecimal value and the characters they produce.

Separate tables show the IBM (ASCII) character sets for:

- Character/graphic (CG) set 1
- Character/graphic (CG) set 2

Part I Commodore Mode

Uppercase/Graphic (Cursor Up) Mode

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	00	none	32	20	blank	64	40	@	96	60	—
1	01	SINGLE	33	21	!	65	41	A	97	61	⊞
2	02	DOUBLE	34	22	"	66	42	B	98	62	
3	03	none	35	23	#	67	43	C	99	63	—
4	04	none	36	24	\$	68	44	D	100	64	—
5	05	none	37	25	%	69	45	E	101	65	—
6	06	none	38	26	&	70	46	F	102	66	—
7	07	none	39	27	'	71	47	G	103	67	
8	08	BIT IMG	40	28	(72	48	H	104	68	
9	09	none	41	29)	73	49	I	105	69	↘
10	0A	LF	42	2A	*	74	4A	J	106	6A	↘
11	0B	none	43	2B	+	75	4B	K	107	6B	↘
12	0C	FF	44	2C	,	76	4C	L	108	6C	L
13	0D	CR	45	2D	-	77	4D	M	109	6D	↘
14	0E	EN ON	46	2E	.	78	4E	N	110	6E	↘
15	0F	EN OFF	47	2F	/	79	4F	O	111	6F	┌
16	10	POS	48	30	0	80	50	P	112	70	┌
17	11	CRSR DWN	49	31	1	81	51	Q	113	71	⊞
18	12	RVS ON	50	32	2	82	52	R	114	72	—
19	13	PAGE OFF	51	33	3	83	53	S	115	73	⊞
20	14	none	52	34	4	84	54	T	116	74	
21	15	none	53	35	5	85	55	U	117	75	↘
22	16	none	54	36	6	86	56	V	118	76	⊞
23	17	none	55	37	7	87	57	W	119	77	0
24	18	none	56	38	8	88	58	X	120	78	⊞
25	19	none	57	39	9	89	59	Y	121	79	
26	1A	SUB	58	3A	:	90	5A	Z	122	7A	⊞
27	1B	ESC	59	3B	;	91	5B	[123	7B	+
28	1C	none	60	3C	<	92	5C	£	124	7C	⊞
29	1D	SKIP	61	3D	=	93	5D]	125	7D	
30	1E	none	62	3E	>	94	5E	↑	126	7E	↑
31	1F	NLQ ON	63	3F	?	95	5F	←	127	7F	▼

Part I Commodore Mode

Uppercase/Graphic (Cursor Up) Mode

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
128	80	none	160	A0	SHT SP	192	C0	—	224	E0	blank
129	81	none	161	A1	█	193	C1	⊕	225	E1	█
130	82	none	162	A2	▒	194	C2		226	E2	▒
131	83	none	163	A3	▓	195	C3	—	227	E3	▓
132	84	none	164	A4	—	196	C4	—	228	E4	—
133	85	none	165	A5		197	C5	—	229	E5	
134	86	none	166	A6	⊗	198	C6	—	230	E6	⊗
135	87	none	167	A7		199	C7		231	E7	
136	88	none	168	A8	⊗	200	C8		232	E8	⊗
137	89	none	169	A9	▒	201	C9	↖	233	E9	▒
138	8A	none	170	AA		202	CA	↖	234	EA	
139	8B	none	171	AB	┆	203	CB	↗	235	EB	┆
140	8C	none	172	AC	█	204	CC	└	236	EC	█
141	8D	CS	173	AD	└	205	CD	↘	237	ED	└
142	8E	none	174	AE	┐	206	CE	↗	238	EE	┐
143	8F	none	175	AF	▒	207	CF	┐	239	EF	▒
144	90	none	176	B0	┐	208	D0	┐	240	F0	┐
145	91	CRSR UP	177	B1	┆	209	D1	⊕	241	F1	┆
146	92	RVS OFF	178	B2	┆	210	D2	—	242	F2	┆
147	93	PAGE ON	179	B3	┆	211	D3	⊕	243	F3	┆
148	94	none	180	B4		212	D4		244	F4	
149	95	none	181	B5	█	213	D5	↗	245	F5	█
150	96	none	182	B6	█	214	D6	⊗	246	F6	█
151	97	none	183	B7	—	215	D7	□	247	F7	—
152	98	none	184	B8	▒	216	D8	⊕	248	F8	▒
153	99	none	185	B9	▒	217	D9		249	F9	▒
154	9A	none	186	BA	└	218	DA	⊕	250	FA	└
155	9B	none	187	BB	█	219	DB	+	251	FB	█
156	9C	none	188	BC	█	220	DC	⊗	252	FC	█
157	9D	none	189	BD	└	221	DD		253	FD	└
158	9E	none	190	BE	█	222	DE	π	254	FE	█
159	9F	NLQ OFF	191	BF	█	223	DF	⊗	255	FF	π

Part I Commodore Mode

Upper/Lowercase (Cursor Down) Mode

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	00	none	32	20	blank	64	40	@	96	60	—
1	01	SINGLE	33	21	!	65	41	a	97	61	A
2	02	DOUBLE	34	22	"	66	42	b	98	62	B
3	03	none	35	23	#	67	43	c	99	63	C
4	04	none	36	24	\$	68	44	d	100	64	D
5	05	none	37	25	%	69	45	e	101	65	E
6	06	none	38	26	&	70	46	f	102	66	F
7	07	none	39	27	'	71	47	g	103	67	G
8	08	BIT IMG	40	28	(72	48	h	104	68	H
9	09	none	41	29)	73	49	i	105	69	I
10	0A	LF	42	2A	*	74	4A	j	106	6A	J
11	0B	none	43	2B	+	75	4B	k	107	6B	K
12	0C	FF	44	2C	,	76	4C	l	108	6C	L
13	0D	CR	45	2D	-	77	4D	m	109	6D	M
14	0E	EN ON	46	2E	.	78	4E	n	110	6E	N
15	0F	EN OFF	47	2F	/	79	4F	o	111	6F	O
16	10	POS	48	30	0	80	50	p	112	70	P
17	11	CRSR DWN	49	31	1	81	51	q	113	71	Q
18	12	RVS ON	50	32	2	82	52	r	114	72	R
19	13	PAGE OFF	51	33	3	83	53	s	115	73	S
20	14	none	52	34	4	84	54	t	116	74	T
21	15	none	53	35	5	85	55	u	117	75	U
22	16	none	54	36	6	86	56	v	118	76	V
23	17	none	55	37	7	87	57	w	119	77	W
24	18	none	56	38	8	88	58	x	120	78	X
25	19	none	57	39	9	89	59	y	121	79	Y
26	1A	SUB	58	3A	:	90	5A	z	122	7A	Z
27	1B	ESC	59	3B	;	91	5B	[123	7B	+
28	1C	none	60	3C	<	92	5C	£	124	7C	⌘
29	1D	SKIP	61	3D	=	93	5D]	125	7D	
30	1E	none	62	3E	>	94	5E	↑	126	7E	⌘
31	1F	NLQ ON	63	3F	?	95	5F	←	127	7F	⌘

Part I Commodore Mode

Upper/Lowercase (Cursor Down) Mode

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
128	80	none	160	A0	SHT SP	192	C0	—	224	E0	blank
129	81	none	161	A1	█	193	C1	A	225	E1	█
130	82	none	162	A2	▒	194	C2	B	226	E2	▒
131	83	none	163	A3	▓	195	C3	C	227	E3	▓
132	84	none	164	A4	░	196	C4	D	228	E4	░
133	85	none	165	A5		197	C5	E	229	E5	
134	86	none	166	A6	⊠	198	C6	F	230	E6	⊠
135	87	none	167	A7	┆	199	C7	G	231	E7	┆
136	88	none	168	A8	⊡	200	C8	H	232	E8	⊡
137	89	none	169	A9	⊢	201	C9	I	233	E9	⊢
138	8A	none	170	AA	┆	202	CA	J	234	EA	┆
139	8B	none	171	AB	┆	203	CB	K	235	EB	┆
140	8C	none	172	AC	■	204	CC	L	236	EC	■
141	8D	CS	173	AD	┆	205	CD	M	237	ED	┆
142	8E	none	174	AE	┆	206	CE	N	238	EE	┆
143	8F	none	175	AF	▒	207	CF	O	239	EF	▒
144	90	none	176	B0	┆	208	D0	P	240	F0	┆
145	91	CRSR UP	177	B1	┆	209	D1	Q	241	F1	┆
146	92	RVS OFF	178	B2	┆	210	D2	R	242	F2	┆
147	93	PAGE ON	179	B3	┆	211	D3	S	243	F3	┆
148	94	none	180	B4		212	D4	T	244	F4	
149	95	none	181	B5	█	213	D5	U	245	F5	█
150	96	none	182	B6	▒	214	D6	V	246	F6	▒
151	97	none	183	B7	▓	215	D7	W	247	F7	▓
152	98	none	184	B8	░	216	D8	X	248	F8	░
153	99	none	185	B9	▒	217	D9	Y	249	F9	▒
154	9A	none	186	BA	┆	218	DA	Z	250	FA	┆
155	9B	none	187	BB	■	219	DB	+	251	FB	■
156	9C	none	188	BC	▒	220	DC	⊠	252	FC	▒
157	9D	none	189	BD	┆	221	DD		253	FD	┆
158	9E	none	190	BE	▒	222	DE	⊠	254	FE	▒
159	9F	NLQ OFF	191	BF	▒	223	DF	⊠	255	FF	⊠

Part II IBM Mode

CG Set 1

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	00	NUL	32	20	blank	64	40	@	96	60	`
1	01	none	33	21	!	65	41	A	97	61	a
2	02	none	34	22	"	66	42	B	98	62	b
3	03	none	35	23	#	67	43	C	99	63	c
4	04	none	36	24	\$	68	44	D	100	64	d
5	05	none	37	25	%	69	45	E	101	65	e
6	06	none	38	26	&	70	46	F	102	66	f
7	07	BEL	39	27	'	71	47	G	103	67	g
8	08	none	40	28	(72	48	H	104	68	h
9	09	HT	41	29)	73	49	I	105	69	i
10	0A	LF	42	2A	*	74	4A	J	106	6A	j
11	0B	VT	43	2B	+	75	4B	K	107	6B	k
12	0C	FF	44	2C	,	76	4C	L	108	6C	l
13	0D	CR	45	2D	-	77	4D	M	109	6D	m
14	0E	SO	46	2E	.	78	4E	N	110	6E	n
15	0F	SI	47	2F	/	79	4F	O	111	6F	o
16	10	none	48	30	0	80	50	P	112	70	p
17	11	none	49	31	1	81	51	Q	113	71	q
18	12	DC2	50	32	2	82	52	R	114	72	r
19	13	none	51	33	3	83	53	S	115	73	s
20	14	DC4	52	34	4	84	54	T	116	74	t
21	15	none	53	35	5	85	55	U	117	75	u
22	16	none	54	36	6	86	56	V	118	76	v
23	17	none	55	37	7	87	57	W	119	77	w
24	18	CAN	56	38	8	88	58	X	120	78	x
25	19	none	57	39	9	89	59	Y	121	79	y
26	1A	none	58	3A	:	90	5A	Z	122	7A	z
27	1B	ESC	59	3B	;	91	5B	[123	7B	{
28	1C	none	60	3C	<	92	5C	\	124	7C	
29	1D	none	61	3D	=	93	5D]	125	7D	}
30	1E	none	62	3E	>	94	5E	^	126	7E	~
31	1F	none	63	3F	?	95	5F	_	127	7F	none

Part II IBM Mode
CG Set 1

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
128	80	NUL	160	A0	á	192	C0	┐	224	E0	α
129	81	none	161	A1	í	193	C1	┌	225	E1	β
130	82	none	162	A2	ó	194	C2	┐	226	E2	Γ
131	83	none	163	A3	ú	195	C3	┐	227	E3	π
132	84	none	164	A4	ñ	196	C4	┐	228	E4	Σ
133	85	none	165	A5	z	197	C5	┐	229	E5	σ
134	86	none	166	A6	ñ	198	C6	┐	230	E6	μ
135	87	BEL	167	A7	▯	199	C7	┐	231	E7	τ
136	88	none	168	A8	◌	200	C8	┐	232	E8	Ϸ
137	89	HT	169	A9	┐	201	C9	┐	233	E9	θ
138	8A	LF	170	AA	┐	202	CA	┐	234	EA	Ω
139	8B	VT	171	AB	┐	203	CB	┐	235	EB	δ
140	8C	FF	172	AC	┐	204	CC	┐	236	EC	Ϸ
141	8D	CR	173	AD	┐	205	CD	┐	237	ED	∅
142	8E	SO	174	AE	✱	206	CE	┐	238	EE	€
143	8F	SI	175	AF	✱	207	CF	┐	239	EF	∏
144	90	none	176	B0	▯	208	D0	┐	240	F0	≡
145	91	none	177	B1	▯	209	D1	┐	241	F1	±
146	92	DC2	178	B2	▯	210	D2	┐	242	F2	≥
147	93	none	179	B3	┐	211	D3	┐	243	F3	≤
148	94	DC4	180	B4	┐	212	D4	┐	244	F4	∫
149	95	none	181	B5	┐	213	D5	┐	245	F5	∫
150	96	none	182	B6	┐	214	D6	┐	246	F6	÷
151	97	none	183	B7	┐	215	D7	┐	247	F7	≈
152	98	CAN	184	B8	┐	216	D8	┐	248	F8	°
153	99	none	185	B9	┐	217	D9	┐	249	F9	·
154	9A	none	186	BA	┐	218	DA	┐	250	FA	-
155	9B	ESC	187	BB	┐	219	DB	▯	251	FB	√
156	9C	none	188	BC	┐	220	DC	▯	252	FC	∅
157	9D	none	189	BD	┐	221	DD	▯	253	FD	²
158	9E	none	190	BE	┐	222	DE	▯	254	FE	■
159	9F	none	191	BF	┐	223	DF	▯	255	FF	SP

Part II IBM Mode
CG Set 2

Part II IBM Mode
CG Set 1

Function

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	00	NUL	32	20	SP	64	40	@	96	60	ˆ
1	01	none	33	21	!	65	41	A	97	61	a
2	02	none	34	22	"	66	42	B	98	62	b
3	03	♥	35	23	#	67	43	C	99	63	c
4	04	♦	36	24	\$	68	44	D	100	64	d
5	05	♣	37	25	%	69	45	E	101	65	e
6	06	♠	38	26	&	70	46	F	102	66	f
7	07	BEL	39	27	'	71	47	G	103	67	g
8	08	none	40	28	(72	48	H	104	68	h
9	09	HT	41	29)	73	49	I	105	69	i
10	0A	LF	42	2A	*	74	4A	J	106	6A	j
11	0B	VT	43	2B	+	75	4B	K	107	6B	k
12	0C	FF	44	2C	,	76	4C	L	108	6C	l
13	0D	CR	45	2D	-	77	4D	M	109	6D	m
14	0E	SO	46	2E	.	78	4E	N	110	6E	n
15	0F	SI	47	2F	/	79	4F	O	111	6F	o
16	10	none	48	30	0	80	50	P	112	70	p
17	11	none	49	31	1	81	51	Q	113	71	q
18	12	DC2	50	32	2	82	52	R	114	72	r
19	13	none	51	33	3	83	53	S	115	73	s
20	14	DC4	52	34	4	84	54	T	116	74	t
21	15	§	53	35	5	85	55	U	117	75	u
22	16	none	54	36	6	86	56	V	118	76	v
23	17	none	55	37	7	87	57	W	119	77	w
24	18	CAN	56	38	8	88	58	X	120	78	x
25	19	none	57	39	9	89	59	Y	121	79	y
26	1A	none	58	3A	:	90	5A	Z	122	7A	z
27	1B	ESC	59	3B	;	91	5B	[123	7B	{
28	1C	none	60	3C	<	92	5C	\	124	7C	
29	1D	none	61	3D	=	93	5D]	125	7D	}
30	1E	none	62	3E	>	94	5E	^	126	7E	~
31	1F	none	63	3F	?	95	5F	_	127	7F	none

Part II IBM Mode
CG Set 2

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
128	80	☉	160	A0	à	192	C0	⌞	224	E0	α
129	81	ü	161	A1	í	193	C1	⌞	225	E1	β
130	82	é	162	A2	ó	194	C2	⌞	226	E2	Γ
131	83	à	163	A3	ú	195	C3	⌞	227	E3	π
132	84	ä	164	A4	ñ	196	C4	⌞	228	E4	Σ
133	85	â	165	A5	ñ	197	C5	⌞	229	E5	σ
134	86	à	166	A6	ñ	198	C6	⌞	230	E6	μ
135	87	☉	167	A7	ñ	199	C7	⌞	231	E7	τ
136	88	ë	168	A8	ñ	200	C8	⌞	232	E8	ϑ
137	89	ë	169	A9	ñ	201	C9	⌞	233	E9	θ
138	8A	ë	170	AA	ñ	202	CA	⌞	234	EA	Ω
139	8B	ï	171	AB	ñ	203	CB	⌞	235	EB	δ
140	8C	ï	172	AC	ñ	204	CC	⌞	236	EC	e
141	8D	ï	173	AD	ñ	205	CD	⌞	237	ED	θ
142	8E	Ä	174	AE	ñ	206	CE	⌞	238	EE	ε
143	8F	Ä	175	AF	ñ	207	CF	⌞	239	EF	Π
144	90	é	176	B0	ñ	208	D0	⌞	240	F0	≡
145	91	æ	177	B1	ñ	209	D1	⌞	241	F1	±
146	92	æ	178	B2	ñ	210	D2	⌞	242	F2	∞
147	93	ö	179	B3	ñ	211	D3	⌞	243	F3	∞
148	94	ö	180	B4	ñ	212	D4	⌞	244	F4	∞
149	95	ò	181	B5	ñ	213	D5	⌞	245	F5	J
150	96	ü	182	B6	ñ	214	D6	⌞	246	F6	≠
151	97	ü	183	B7	ñ	215	D7	⌞	247	F7	≈
152	98	ÿ	184	B8	ñ	216	D8	⌞	248	F8	o
153	99	ÿ	185	B9	ñ	217	D9	⌞	249	F9	"
154	9A	Ü	186	BA	ñ	218	DA	⌞	250	FA	-
155	9B	ϕ	187	BB	ñ	219	DB	⌞	251	FB	J
156	9C	£	188	BC	ñ	220	DC	⌞	252	FC	n
157	9D	£	189	BD	ñ	221	DD	⌞	253	FD	z
158	9E	£	190	BE	ñ	222	DE	⌞	254	FE	■
159	9F	f	191	BF	ñ	223	DF	⌞	255	FF	SP

Appendix B

Secondary Addresses and Control Codes

COMMODORE mode

Secondary address

The secondary address is used in Commodore mode to specify a particular printer function, such as setting the number of lines to be printed on a page or setting spaces between lines. The MPS 1000's secondary addresses are listed below.

- 0 Print data exactly as received in Uppercase/Graphics mode
- 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
- 7 Print data exactly as received in Upper/lower case
- 9 Suppress diagnostic message printing
- 10 Reset printer

Control codes

Hexadecimal	Decimal	Function
01 H	CHR\$(1)	Single density (480 DPL) bit image graphics
02 H	2	Double density (960 DPL) bit image graphics
08 H	8	Bit image (7-vertical dot) with 7/72" line feed
0AH	10	Line feed (with carriage return)
0CH	12	Form feed
0DH	13	Carriage return (with line feed)
0E H	14	Enhanced on
0F H	15	Enhanced off
10 H, n1, n2	16	Tab setting "NHNL"
11 H	17	Local Lower-/uppercase (cursor down) mode
12 H	18	Reverse on
13 H	19	Paging off
1AH	26	Bit image repeat (7-pin)
1BH, 10H, n1, n2	27 16	Dot address $n1 * 256 + n2$
1DH	29	Skip space in formatting print*
1F H	31	NLQ on
22 H	34	Quote mode on
8DH	141	CR without LF
91 H	145	Local Uppercase/graphics (cursor up) mode
92 H	146	Reverse off
93 H	147	Paging on (including set top of form)
9F H	159	NLQ off
A0H	160	Prints blank alpha field in formatting print*
FE H	254	Print Programmable Character

NOTE:

*Control codes valid for sa = 1

IBM mode

Control Codes in Numeric Order

You can use any of the methods explained in Chapter 3 to send these control codes to your MPS 1000. If you use Commodore BASIC, you send a control code using `PRINT$4,CHR$(27) CHR$(69)` or `PRINT#4,CHR$(27)“E”`. If you use MBASIC, you send a control code using `LPRINT CHR$(27)` followed by `CHR$(n)`, where `n` is a number from the decimal (Dec) column below. For those codes with a dash in the ESC column, you omit the ESCape code. For the codes above decimal 27, you can use either the character in the symbol column (in quotation marks) or `CHR$(n)`. For example, the command for turning on emphasized lettering can be entered as:

```
LPRINT CHR$(27) CHR$(69) or LPRINT CHR$(27) “E”
```

See Chapter 3 and your system documentation for other methods of sending codes to the printer.

Abbreviations: cpi = characters per inch
cps = characters per second

ESC	DEC	HEX	Symbol	Function
-	0	00	NUL	Terminated horizontal and vertical tab setting ESCape sequences.
-	7	07	BEL	Sounds Beeper
-	9	09	HT	Horizontal tab. Empties the printer buffer, then moves the print head to the next tab stop.
-	10	0A	LF	Line feed. Empties the printer buffer, performs a line feed at the current line spacing, and resets the buffer character count to 0.
-	11	0B	VT	Vertical tab. Empties the printer buffer, then advances the paper to the next vertical tab stop.
-	12	0C	FF	Form feed. Empties the printer buffer, then advances the paper to the next top of form.
-	13	0D	CR	Carriage return. Prints the contents of the buffer and resets the buffer character count to 0.

ESC	DEC	HEX	Symbol	Function
-	14	0E	SO	Shift out. Turns enlarged mode ON for the length of the line. Same as ESC S9. Can be canceled by ASCII 20 or ESC "W0". Works with pica, elite or condensed mode.
-	15	0F	SI	Shift in. Empties buffer and turns condensed (17.16 cpi) ON. Cannot mix with emphasized or pica.
-	18	12	DC2	Clears condensed mode.
-	20	14	DC4	Turns the enlarged mode set by ASCII 14 OFF.
-	24	18	CAN	Cancels all text but not control codes in the printer buffer.
-	27	1B	ESC	ESCape: Prepares the printer to receive control codes.
ESC	45	2D	-	Turns underline mode ON. Format: ESC "—"n where n toggles underline on and off: 0 turns it OFF, 1 turns it ON.
ESC	48	30	0	Sets line spacing to 1/8-inch (9-dot).
ESC	49	31	1	Sets line spacing to 7/72 inch (7-dot).
ESC	50	32	2	Sets line spacing to previous definition (ESC "A").
ESC	51	33	3	Sets line spacing to N/216 inch (1/216-inch is 1/3-dot). Stays on until changed. Format: ESC "3"n
ESC	53	35	5	Turns auto line feed mode ON. Format: ESC "5" n where n toggles auto LF on and off. 0 turns it OFF, 1 turns it ON.
ESC	54	36	6	Select IBM Character/Graphic (CG) Set 2
ESC	55	37	7	Select IBM CG Set 1
ESC	56	38	8	Disables the paper-out sensor. Can also be disabled with DIP switch 1-5.
ESC	57	39	9	Enables the paper-out sensor. Can also be enabled with DIP switch 1-5.
ESC	58	3A		Select elite mode (draft only).
ESC	60	3C	<	Turns one-line unidirectional mode ON. Prints each line from left to right.

ESC	DEC	HEX	Symbol	Function
ESC	64	40	@	Reset code, which resets the printer to its power-up state including resetting top of page. Clears all text and control codes from the print buffer.
ESC	65	41	A	Sets line spacing to $n/72$ -inch (n -dot). Format: ESC "A" n where $n = 0-255$. This value effective after sending ESC "2".
ESC	66	42	B	Sets up to 16 vertical tabs in the current line spacing. Tab settings are not affected by subsequent changes in line spacing. Format: ESC "B" $n_1 n_2 \dots n_k 0$ where $n_k = 1-255$. Terminate this tab sequence with 0 or a number less than the last defined tab (n_k).
ESC	67	43	C	Sets the form length to n lines in the current line spacing. The default is 66 lines. Also resets top of form. Format: ESC "C" n where $n = 1-127$.
ESC	67	43	C	Sets the form length to n inches, regardless of the current line spacing. The default length is 11 inches. Also resets top of form. Format: ESC "C" $0n$ where $n = 1-22$.
ESC	68	44	D	Resets the current tabs and sets up to 32 horizontal tabs in the current pitch. Tabs may range up to the maximum width for the character and printer size. For example, the maximum tab for pica characters on an 8-inch line is 79. Tab settings are not affected by subsequent changes in pitch. Format: ESC "D" $n_1 n_2 \dots n_k 0$ Terminate this tab sequence with 0 or a number less than the last defined tab (n_k).
ESC	69	45	E	Turns emphasized mode ON. Can not mix with elite or compressed mode.

ESC	DEC	HEX	Symbol	Function
ESC	70	46	F	Turns emphasized mode OFF.
ESC	71	47	G	Turns double-strike mode ON. Cannot mix with NLQ mode.
ESC	72	48	H	Turns double-strike mode OFF.
ESC	74	4A	J	Forces an immediate line feed of $n/216$ inches without changing the current line spacing. Prints the contents of the buffer without a carriage Format: ESC "J" n where $n = 0-255$.
ESC	75	4B	K	Turns single-density graphics mode ON. Prints 480 dots per 8-inch line. Format: ESC "K" $n_1 n_2$ followed by n data numbers, where n = total number of dot columns, $n_1 = n \text{MOD} 256$, $n_2 = \text{INT}(n/256)$. For example, to print 480 dots $n_1 = 244, n_2 = 1$.
ESC	76	4C	L	Turns low-speed, double-density graphics mode ON. Prints 960 dots per 8-inch line. Format: ESC "L" $n_1 n_2$ followed by n data numbers, where n = total number of dot columns $n_1 = n \text{MOD} 256$, $n_2 = \text{INT}(N/256)$. For example, to print 960 dots $n_1 = 192, n_2 = 3$.
ESC	78	4E	N	Sets skip-over-perforation to n lines. Format: ESC "N" n where $n = 1-127$.
ESC	79	4F	O	Turns skip-over-perforation OFF.
ESC	82	52	R	Initializes tabs (vertical and horizontal) to default setting. The vertical tab is equal to a line feed.

ESC	DEC	HEX	Symbol	Function
ESC	83	53	S	Turns script mode ON. Either type of script is printed in double-strike. Format: ESC "S"CHR\$(n) where n = 0 produces superscript, and 1 produces subscript.
ESC	84	54	T	Turns script mode OFF.
ESC	85	55	U	Turns unidirectional mode ON. Prints each line from left to right. Format: ESC "U"n where n toggles the mode on and off: 0 turns it OFF, 1 turns it ON.
ESC	87	57	W	Turns enlarged mode ON; stays ON until turned OFF. Cannot be turned off with ASCII 20. Format: ESC "W"n where n toggles the mode on and off: 0 turns it OFF, 1 turns it ON.
ESC	88	58	X	Setting left and right margin. Format: ESC "X" 1m rm where 1 m rm 255.
ESC	89	59	Y	Turns high-speed double density graphics mode ON; gives the same density as ESC "L" but cannot print two adjacent dots in the same row.
ESC	90	5A	Z	Turns quadruple-density graphics mode ON. Allows 1,920 dot positions per 8-inch line. Format: ESC "Z" n ₁ n ₂ followed by n data numbers, where n = total number of dot columns n ₁ = n MOD 256 n ₂ = INT (n/256). For example, to print 1,920 dots, n ₁ = 128, n ₂ = 7.
ESC	120	78	x	Selects draft or NLQ mode ESC "x"0 selects draft. ESC "x"1 selects NLQ. ESC "x0" and ESC "x1" will also work.

Control Codes by Function (IBM mode)

This appendix shows most of the control codes in Appendix B, but this time they are arranged by categories. If your computer cannot generate lowercase letters, use the equivalent decimal ASCII values.

This appendix enables you to find a particular code quickly but does not give all the details on the format and use of each code. See Appendix B or the appropriate chapter for further information.

Near Letter Quality Mode

- ESC "x" CHR\$(1) Turns NLQ mode ON.
- ESC "x" CHR\$(0) Turns draft mode ON.

Character Width (Pitch)

- ASCII 15 Turns condensed mode ON.
- ASCII 18 Turns condensed mode OFF.
- ASCII 14 Turns one-line enlarged mode ON.
- ASCII 20 Turns one-line enlarged mode OFF.
- ESC ":", Turns elite mode ON.
- ESC "W" CHR\$(0) Turns either enlarged mode OFF.
- ESC "W" CHR\$(1) Turns continuous enlarged mode ON.

Character Weight

- ESC "E" Turns emphasized mode ON.
- ESC "F" Turns emphasized mode OFF.
- ESC "G" Turns double-strike mode ON.
- ESC "H" Turns double-strike mode OFF.

Print Enhancement

- ESC "S" CHR\$(0) Turns superscript mode ON.
- ESC "S" CHR\$(1) Turns subscript mode ON.
- ESC "T" Turns either script mode OFF.
- ESC "._" CHR\$(0) Turns underline mode OFF.
- ESC "._" CHR\$(1) Turns underline mode ON.

Mode and Character-Set Selection

ESC "7"	Select CG Table 1
ESC "6"	Select CG Table 2
ESC "@"	Reset code.

Special Printer Features

ESC "<"	Turns one-line unidirectional mode ON.
ESC "U" CHR\$(0)	Turns continuous unidirectional mode OFF.
ESC "U" CHR\$(1)	Turns continuous unidirectional mode ON.

Line Spacing

ESC "0"	Sets line spacing to 1/8-inch.
ESC "1"	Sets line spacing to 7/72-inch.
ESC "2"	Executes the line spacing specified by ESC "A." If no ESC "A" command has been given, line feeding returns to 1/6-inch.
ESC "A" n	Sets line spacing to n/72-inch (effective after sending ESC "2").
ESC "3" n	Sets line spacing to n/216-inch.
ESC "J" n	Produces an immediate one-time line feed of n/216-inch without a carriage return.

Forms Control

ASCII 12	Activates a form-feed.
ESC "8"	Turns the paper-out sensor OFF.
ESC "9"	Turns the paper-out sensor ON.
ESC "C" 0 n	Sets the form length in inches.
ESC "C" n	Sets the form length in lines.
ESC "N" n	Turns variable skip-over-perforation ON.
ESC "O"	Turns skip-over-perforation OFF.

Page Format

ASCII 9	Activates a horizontal tab.
ASCII 11	Activates a vertical tab.
ESC "B"	Sets vertical tab stops.
ESC "D"	Sets horizontal tab stops.
ESC "R"	Initializes vertical and horizontal tabs to default values.
ESC "X" lm rm	Sets left margin and right margin

Carriage Return and Line Feed

ASCII 10	Activates a line feed.
ASCII 13	Activates a carriage return.
ESC "5" CHR\$(0)	Turns auto LF OFF.
ESC "5" CHR\$(1)	Turns auto LF ON.

Dot Graphics

ESC "K"	Turns single-density graphics mode ON.
ESC "L"	Turns double-density graphics mode ON.
ESC "Y"	Turns high-speed double-density graphics mode ON.
ESC "Z"	Turns quadruple-density graphics mode ON.

Miscellaneous

ASCII 0	Terminates horizontal and vertical tab setting sequence.
ASCII 7	Sounds beeper.
ASCII 24	Cancels all text but not control codes in the printer buffer.

Appendix C

Formatting Examples

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

Simple String Formatting

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

Here is a printout from the program:

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

```
10 OPEN 1,4
20 OPEN 2,4,1
30 OPEN 3,4,2
40 OPEN 4,4,4
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$(29)
120 PRINT #2,B$C$B$C$B$C$B$
130 CLOSE 1:CLOSE 2:CLOSE 3:CLOSE 4
```

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 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).
50	Enables 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	Sets C\$ equal to the format character for "skip a space"		
120	Prints the formatted data string 4 times		
130	Closes the files		

Leading Zeroes Forced

This program prints ten rows of numbers, each row beginning with a number one higher than 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.

```

10 OPEN 1,4
20 OPEN 2,4,1
30 OPEN 3,4,2
40 OPEN 4,4,4
50 PRINT #4 :REM ENABLE ERROR DIAGNOSTICS
60 F$ = "SZZZ. SZZZ. ZZZZ. ZZZZ"
70 PRINT #3,F$
80 FOR I = 1 TO 10
90 PRINT #2,I,I + 1,I + 2,I + 3

```



```
100 NEXT
110 CLOSE 1:CLOSE 2:CLOSE 3:CLOSE 4
```

+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

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.

```
10 OPEN 1,4
20 OPEN 2,4,1
30 OPEN 3,4,2
40 OPEN 4,4,4
50 PRINT #4 :REM ENABLE ERROR DIAGNOSTICS
60 F$ = "999 S999 999 - "
70 PRINT #3,F$
80 PRINT #2, -1, -1, -1
90 CLOSE 1:CLOSE 2:CLOSE 3:CLOSE 4

1 - 1 1-
```

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 reverse is true, then the formatted number is truncated.

```
10 OPEN 1,4
20 OPEN 2,4,1
30 OPEN 3,4,2
40 OPEN 4,4,4
```

```

50 PRINT #4 :REM ENABLE ERROR DIAGNOSTICS
60 F$ = "999.99 S999.99999 999"
70 PRINT #3,F$
80 FOR I= 1 TO 10
90 PRINT #2, - 10*I/2, - 5*I/4,3*I
100 NEXT I
110 CLOSE 1:CLOSE 2:CLOSE 3:CLOSE 4

```

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

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 blanks.

```

10 OPEN 1,4
20 OPEN 2,4,1
30 OPEN 3,4,2
40 OPEN 4,4,4
50 PRINT #4
60 F$ = "99.99 $$$ .99 99999"
70 PRINT #3,F$
80 FOR I= 1 TO 10
85 X = RND(1)
90 PRINT #2,X*100,X*10,X*10000
100 NEXT I
110 CLOSE 1:CLOSE 2:CLOSE 3:CLOSE 4

```

\$66.29	\$6.62	\$ 6629
\$45.83	\$4.58	\$ 4583
\$32.86	\$3.28	\$ 3286
\$53.33	\$5.33	\$ 5333
\$74.91	\$7.49	\$ 7491
\$ 7.79	\$.77	\$ 779
\$97.51	\$9.75	\$ 9751
\$27.03	\$2.70	\$ 2703
\$15.28	\$1.52	\$ 1528
\$49.95	\$4.99	\$ 4995

Mixed Format Field Types

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











In line 120, 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 OPEN 1,4
20 OPEN 2,4,1
30 OPEN 3,4,2
40 OPEN 4,4,4
50 PRINT #4                          :REM ENABLE ERROR DIAGNOSTICS
60 F$ = "ZZZZZ AAAAAAAAAAAAAAAAAAAAAA $$$$$.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 CLOSE 1:CLOSE 2:CLOSE 3:CLOSE 4
```

Formatting with Literals

In the example below, literal characters within a format line are created by preceding each literal character with the reverse on symbol.

```
10 OPEN 1,4
20 OPEN 2,4,2
30 OPEN 3,4,1
40 F$ = "     * ZZ   $$$$  
   ZZ.999   + "
50 PRINT #2,F$
60 FOR I=1 TO 10:X=10*RND(1):Y=1000*RND(1):Z=8*RND(1)
70 PRINT #3,X;Y;Z:NEXT
80 CLOSE1:CLOSE2:CLOSE3
```


* 04	◆	\$502		03.756	+
* 08	◆	\$582		02.813	+
* 04	◆	\$7		00.268	+
* 04	◆	\$670		00.969	+
* 03	◆	\$272		07.298	+
* 03	◆	\$292		00.234	+
* 04	◆	\$382		03.260	+
* 06	◆	\$76		03.091	+
* 07	◆	\$73		01.361	+
* 07	◆	\$645		03.354	+

Mixed Format Field Types

Show below are an integer with leading zeros, a float, a dollar, and integer with zero suppressed.

- in line 130, CHR\$(39) (tab) is used as a terminator for the space field so that it is formatted correctly.

Fixed and Floating Fields

In fixed dollar fields the dollar sign is placed in the field in the format of \$###,###.###.

REMEDIABLE ERROR DIAGNOSTICS

```

90 B$ = "PET COMPUTER"
100 C = 797
110 D = 70
120 PRINT "A,B,CHR$(39),C,D"
130 CLOSE 1:CLOSE 2:CLOSE 3:CLOSE 4

```

Formatting with literals

In the example below, literal characters within a format list are used by preceding each literal character with the escape character.

```

10 OPEN 1:PRINT
20 OPEN 2:PRINT
30 PRINT "A B C"
40 FOR I=1 TO 10:PRINT "I=";I;NEXT I
50 PRINT "X Y Z"
60 CLOSE 1:CLOSE 2:CLOSE 3

```

Appendix D

Serial and Parallel Interface

The MPS 1000 printer has both a serial and parallel interface to communicate with your computer: this appendix describes it.

Commodore Mode

The serial interface is used when the printer is in its Commodore mode. Connector pin assignments and a description of respective interface signals are shown in Table D-1.

Serial I/O

Pin	Type
1	SERIAL SRQIN
2	GND
3	SERIAL ATN IN/OUT
4	SERIAL CLK IN/OUT
5	SERIAL DATA IN/OUT
6	RESET

Table D-1. Pins and signals

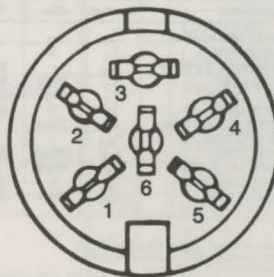
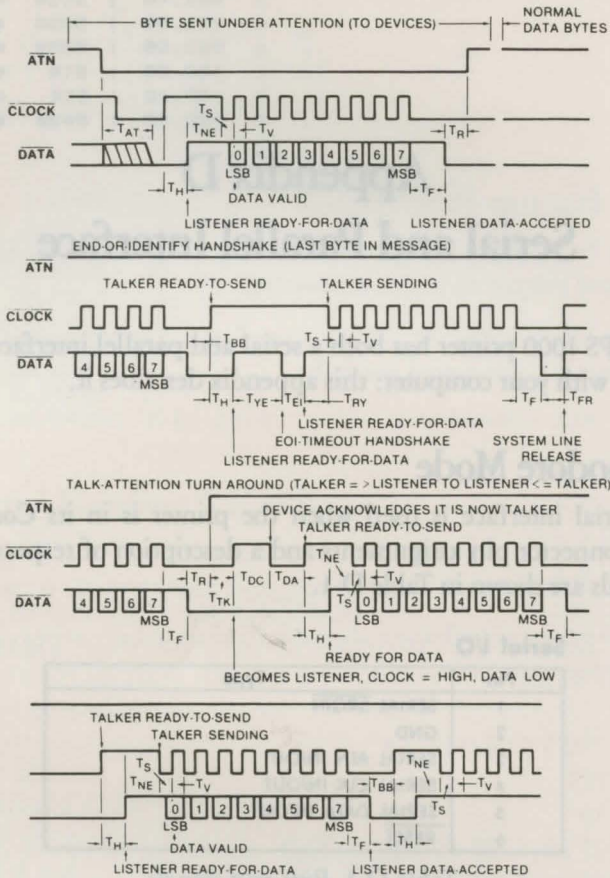


Figure D-1. The 6-pin DIN connector looks like (from outside)

Data Transfer Sequence

Figure D-2 shows the timing for serial interface.



SERIAL BUS TIMING

Description	Symbol	Min.	Typ.	Max.
ATN RESPONSE (REQUIRED) ¹	T_{AT}	—	—	1000 μ s
LISTENER HOLD OFF	T_H	0	—	∞
NON-EOI RESPONSE TO RFD ²	T_{NE}	—	40 μ s	200 μ s
BIT SET-UP TALKER ⁴	T_S	20 μ s	70 μ s	—
DATA VALID	T_V	20 μ s	20 μ s	—
FRAME HANDSHAKE ³	T_F	0	20 μ s	1000 μ s
FRAME TO RELEASE OF ATN	T_R	20 μ s	—	—
BETWEEN BYTES TIME	T_{BB}	—	—	—
EOI RESPONSE TIME	T_{VE}	200 μ s	250 μ s	—
EOI RESPONSE HOLD TIME	T_{EI}	60 μ s	—	—
TALKER RESPONSE LIMIT	T_{RY}	0	30 μ s	60 μ s
BYTE-ACKNOWLEDGE ⁴	T_{PR}	20 μ s	30 μ s	—

Notes:

1. If maximum time exceeded, device not present error.
2. If maximum time exceeded, EOI response required.
3. If maximum time exceeded, frame error.
4. T_V and T_{PR} minimum must be 60 μ s for external device to be a talker.

Figure D-2. Serial Interface timing

IBM Mode

The parallel interface is used when the printer is in its IBM 5152 mode. Connector pin assignments and a description of respective interface signals are shown in Table D-2.

Table D-2 Pins and Signals (parallel)

Signal Pin	Return Pin	Signal	Direction	Description
1	19	STROBE	IN	STROBE pulse to read data in. Pulse width must be more than 0.5 microseconds at the receiving terminal.
2	20	DATA1	IN	These signals represent information of the 1st to 8th bits of parallel data, respectively. Each signal is at HIGH level when data is logical 1 and LOW when it is logical 0.
3	21	DATA2	IN	
4	22	DATA3	IN	
5	23	DATA4	IN	
6	24	DATA5	IN	
7	25	DATA6	IN	
8	26	DATA7	IN	
9	27	DATA8	IN	
10	28	ACKNLG	OUT	Approximately 12 microsecond pulse. LOW indicates that data has been received and that the printer is ready to accept more data.
11	29	BUSY	OUT	A HIGH signal indicates that the printer cannot receive data. The signal goes HIGH in the following cases: 1) During data entry. 2) During printing. 3) When off-line. 4) During printer-error state.
12	30	PE	OUT	A HIGH signal indicates that the printer is out of paper.
13	—	—	—	Pulled up to +5 volts through 3.3K ohm resistance.
14	—	AUTO FEED XT	IN	When this signal is LOW, the paper is automatically fed 1 line after printing. (The signal level can be fixed to this by setting DIP switch 2-3 to ON.)
15	—	NC	—	Unused.
16	—	0V	—	Logic ground level.
17	—	CHASSIS GND	—	Printer's chassis ground, which is isolated from the logic ground.
18	—	NC	—	Unused.
19-30	—	GND	—	Twisted-pair return signal ground level.
31	—	INIT	IN	When this level becomes LOW, the printer controller is reset to its power-up state and the print buffer is cleared. This level is usually HIGH; its pulse width must be more than 50 microseconds at the receiving terminal.

Table D-2, continued

Signal Pin	Return Pin	Signal	Direction	Description
32	—	ERROR	OUT	This level becomes LOW when the printer is in: 1) Paper-end state. 2) Off-line. 3) Error state.
33	—	GND	—	Same as for pins 19-30.
34	—	NC	—	Unused.
35	—	—	—	Pulled up to +5V through 3.3K ohm resistance.
36	—	SLCT IN	IN	Data entry to the printer is possible only when this level is LOW; DIP switch 202 is set for this at the factory.

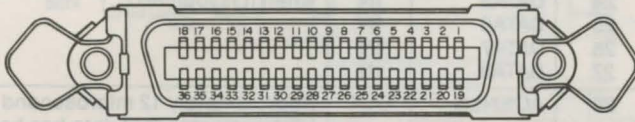


Figure D-3. Connector on the printer side and its pinout

Notes:

1. The column heading Direction refers to the direction of signal flow as viewed from the printer.
2. Return denotes the twisted-pair return, to be connected at signal ground level. For the interface wiring, be sure to use a twisted-pair cable for each signal and to complete the connection on the return side. To prevent noise, these cables should be shielded and connected to the chassis of the host computer or the printer.
3. All interface conditions are based on TTL level. Both the rise and the fall times of each signal must be less than 0.2 microseconds.
4. Data transfer must be carried out by observing the ACKNLG or BUSY signal. (Data transfer to this printer can be carried out only after receipt of the ACKNLG signal or when the level of the BUSY signal is LOW.)
5. Under normal conditions, printer cable pins 11, 12 and 32 are activated when the paper-end condition is detected. The ESCape "8" code disables pins 11 and 32, but not pin 12. Those computers that monitor pin 12 halt printing when the paper is out, making ESCape "8" ineffective.

Data Transfer Sequence

Interface timing

Figure D-4 shows the timing for the parallel interface.

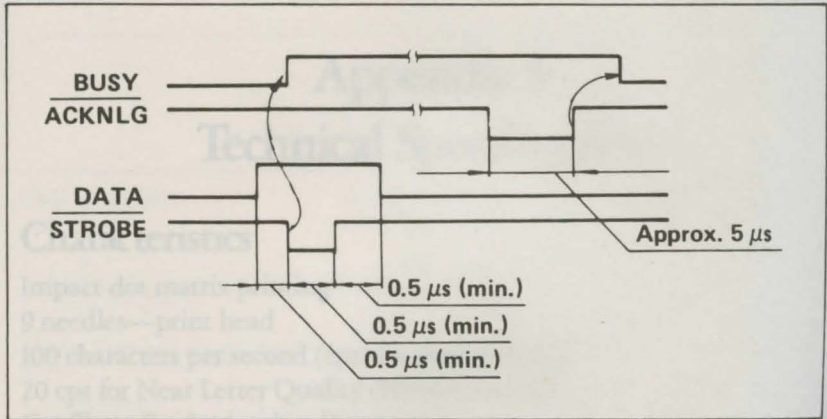


Figure D-4. Parallel interface timing

Signal relationships

Table D-3 shows the way data entry is handled in the on-line and off-line states by showing the relationships between seven signal sets.

On-Line	SLCT IN	ERROR	BUSY	ACKNLG	DATA ENTRY
OFF	HIGH/LOW	LOW	HIGH	Not generated	Disabled
ON	HIGH	HIGH	LOW/HIGH	Generated after data entry	Enabled*
ON	LOW	HIGH	Same	Same	Enabled (normal entry)

Table D-3. Signal interrelations

*When SLCT IN is at high level, data entry is enabled, but the input data will be disregarded until SLCT IN is at low level.

Note: ERROR status is assumed to rest only in off-line state, and the ERROR status does not always mean SLCT IN.

Appendix E

Technical Specifications

Characteristics

- Impact dot matrix printing
- 9 needles—print head
- 100 characters per second (cps) for draft printing
- 20 cps for Near Letter Quality (NLQ) printing
- Cut Sheet Feeder (optional)
- Two printing modes:
 - Commodore mode
 - IBM compatible mode
- Near Letter Quality 1 font resident for each mode

Printing Specifications

Commodore Mode:

- Printing method Impact dot matrix
- Printing speed Draft (pica) : 100 cps
 - Enlarged : 50 cps
 - NLQ : 20 cps
 - Enlarged NLQ : 10 cps
- Line spacing Programmable minimum 1/216" (1/6" at power on)
- Dot matrix format 12x18 NLQ characters
 - 9x8 standard characters
 - 6x8 standard characters (for Graphics characters)
- Character sets Commodore Uppercase/graphic character set
 - Commodore Lowercase/uppercase character set
 - With a NLQ font available for each character set

Character size:

Character pitch	Width (mm)	Height (mm)
Pica	2.1	3.1
Enlarged	4.2	3.1

Column width (maximum characters/line):

Character pitch	column width	column/inch
Pica	80	10
Enlarged	40	5

Printing direction: Bidirectional with logic seeking in the text mode and Graphic characters.

Unidirectional (left to right) in the bit image mode or DIP switch setting (switch 1-8 ON).

Duty cycle: Sustain a print rate of 9 dots per character on an 80-character line after temperature stabilization.

Line feed time: Approximately 150 ms/line for 1/6" line spacing.
Approximately 100 ms/line for a page feed.

Paper feed method: Friction feed.
Tractor feed.

IBM mode

Printing method: Impact dot matrix

Printing speed :

Pica	:100 cps max
Enlarged	:50 cps max
Emphasized	:50 cps max
Enlarged emphasized	:25 cps max
Condensed	:82 cps max
Enlarged condensed	:41 cps max
Elite	:60 cps max
Enlarged elite	:30 cps max
Condensed elite	:100 cps max

Line spacing: Programmable minimum 1/216" (1/6" at power on)

Dot matrix format: 9 x 9 standard characters
 6 x 12 standard characters (for Graphics characters)
 12 x 18 NLQ characters

Character sets: IBM CG Table 1 characters
 IBM CG Table 2 characters

with a NLQ mode available for each of the above character fonts.

Character size

Character pitch	Width (mm)	Height (mm)
Pica	2.1	3.1
Elite	1.4	3.1
Condensed	1.05	3.1
Enlarged Pica	4.2	3.1
Enlarged Elite	2.8	3.1
Enlarged Condensed	2.1	3.1
Emphasized	2.1	3.1
Enlarged Emphasized	4.2	3.1
Super/subscript	depends on pitch	1.6

Column width (maximum characters/line):

Character pitch	column width	column/inch
Pica	80	10
Elite	96	12
Condensed	132 (137)*	17
Enlarged Pica	40	05
Enlarged Emphasized	40	05
Enlarged Elite	48	06
Enlarged Condensed	66 (68)*	8.5
Emphasized	80	10

NOTE:

*When Left/Right margin is set with ESC X.

Printing direction: Bidirectional with logic seeking in the text mode and Graphic characters.

Unidirectional (left to right) in the bit image mode or programmable from a host computer.

Duty cycle: Sustain a print rate of 9 dots per character on an 80-character line after temperature stabilization.

Line feed time: Approximately 150 ms/line for 1/6" line spacing.
Approximately 100 ms/line for a page feed.

Paper feed method: Friction feed.
Tractor feed.

Paper handling

Paper:

Fanfold paper: The adjustable tractor feed can handle paper 101.6 mm (4") to 254 mm (10") wide.

Cut sheet: The built-in friction feed mechanism can handle paper 184.2 mm (7.25") to 215.9 mm (8.5") wide.

Copies: Up to 2 sheets (including the original)
Total paper thickness not to exceed 0.13 mm (0.005").

Paper path: rear

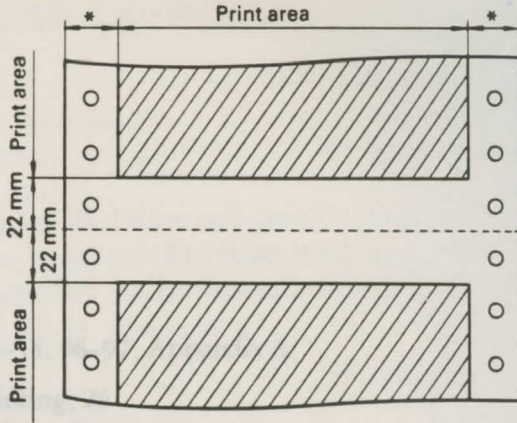
Basic weight of paper: 46.5 to 81.4 g/square m for 1 sheet.
39.5 to 52.3 g/square m for Multi-form.

Ribbons exclusive: Ribbon cartridge. black

Printable area:

Fanfold paper: The print area must be within the range indicated below.

Cut sheet paper: The print area must be within the range indicated below.



The border width indicated by the asterisks (*) varies according to the paper width.
 With 254 mm paper, the border width is 25.4 mm.
 With 241 mm paper, the border width is 19 mm.

Figure Printable Area (Fanfold Paper)

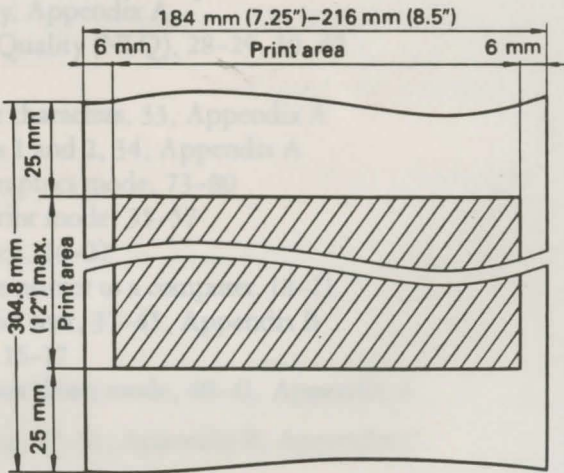


Figure Printable Area (Cut Sheet)

INDEX

- ASCII codes, 31–33, 96–97, Appendix A
- Bidirectional printing, 26
- Bit image graphic mode, 75–78
- Bold modes (IBM mode)
 - Emphasized, 63
 - Double-strike, 65
- Canceling codes (IBM mode), 61
- Carriage return without line feed, 42
- Character fonts
 - Draft quality, Appendix A
 - Near Letter Quality (NLQ), 28–29, 39, 62
- Character sets
 - Commodore characters, 33, Appendix A
 - IBM CG sets 1 and 2, 34, Appendix A
- Commodore graphics mode, 73–80
- Commodore print mode, 35–57
- Condensed pitch, 26–27
- Connecting the printer to a computer, 18–21
- Control codes for text, 37–45, Appendix B
- Control panel, 16–17
- Cursor up/cursor down mode, 40–41, Appendix A
- Data formatting, 47–51, Appendix B, Appendix C
- Density
 - Commodore mode, 45, 80
 - IBM mode, 88–89
- Designing your own characters
 - Commodore mode, 54–59, 75–79
 - IBM mode, 90–97
- Diagnostic messages
 - Disabling messages, 57, Appendix B
 - Enabling messages, 52–53, Appendix B
 - Printer diagnostic messages, 53

DIP switches, 22-23

 Setting for IBM mode, 59

Dot address determination, 79

Dot graphics, 73-97

Dot matrix printing, 25-29

Dot spacing, 43

Double density graphics (Commodore mode), 80

Double-strike characters (IBM mode), 65

Draft quality characters, 28, Appendix A

Elite pitch, 26-27

Emphasized characters, 63

Enlarged printing

 Commodore mode, 38

 IBM mode, 65-66

ESCAPE codes, 33, 63-68, Appendix B

Formatting characters, 47-50

Formatting data

 Formatting characters, 47-50, Appendix C

 Numeric, 48-49, Appendix C

 Literals in format strings, 51, Appendix C

 Storage of a format, 47-51

Formatting Examples, 49, Appendix C

Graphics

 Commodore mode, 73-80

 IBM mode, 81-97

IBM character/graphics (CG) sets 1 and 2, 34, Appendix A

IBM dot graphics mode, 83-97

IBM print mode, 51-79

Installing the printer ribbon, 7-9

Interfaces (serial and parallel), Appendix D

Line spacing

 Commodore mode, 56

 IBM mode, 69-71

Lines per page, 52

Lowercase/uppercase character set, 40-41, Appendix A

Maintenance, 24

Near Letter Quality (NLQ) mode, 28-29, 39, 62

OPEN command, 36-37, 45-46

- Page formatting (IBM mode), 68–71
- Page length, 52
- Paper feed knob, 7
- Paper handling, Appendix E
- Paper loading, 10–11, 14–16
- Paper separator, 5, 13
- Paper-out sensor, 71
- Parallel interface, Appendix D
- Parallel port connection, 20–21
- PETASCII codes, 31–33, Appendix A
- Pica, 26–28, 60
- Pitches, 26–28, 60–63
- Print density, 45
- Print head, 25, 74–75, 82–83
- Print mode
 - Commodore, 35–57
 - IBM, 59–71
- Printer ribbon, 7–10
- Printer specifications, Appendix E
- Programmable characters
 - Commodore mode, 54–57, 75–79
 - IBM mode, 90–93
- Quotation mode, 45
- Resetting the printer, 61
- Reverse printing, 39–40
- Secondary addresses, 33, 45–57, Appendix B, Appendix C
- Serial connection, 18–20
- Serial interface, Appendix D
- Setting up
 - Paper feed knob, 7
 - Ribbon cassette, 7–9
- Superscript/subscript, 68
- Tabs, 42–43
- Technical specifications, Appendix E
- Test pattern, 17
- Tractor unit
 - Installation, 12–13
 - Loading paper, 14–16
- Troubleshooting, 99–102
- Type pitches (IBM mode), 26–27, 60–63

Underlining 32-33
Commodore mode, 42
IBM mode, 67-68
Unpacking the printer, 5
Uppercase/graphic mode, 40-41, Appendix A
WIDTH statements, 89, 101-102
Page formatting (IBM mode), 68-71
Page length, 32
Paper lead knob, 7
Paper handling, Appendix B
Paper loading, 10-11, 14
Paper separator, 5, 13
Paper-on sensor, 71
Parallel interface, Appendix D
Parallel port connection, 20-21, Appendix B
PETASCI codes, 31-32, Appendix A
Pic, 26-28, 60
Pictor, 26-28, 60-62
Print density, 42
Print head, 22, 74-75, 82-83
Print mode
Commodore, 22-27
IBM, 29-71
Printer ribbon, 7-10
Printer specifications, Appendix B
Programmable character
Commodore mode, 24-27
IBM mode, 90-93
Quotation mode, 42
Restoring the printer, 61
Reverse printing, 39-40
Secondary address, 22-27, Appendix B
Serial connection, 18-20
Serial interface, Appendix D
Setting up
Paper lead knob, 7
Ribbon cassette, 7-9
Superscript/subscript, 68
Tab, 42-43
Technical specifications, Appendix B
Ten percent, 17
Terror unit
Installation, 13-15
Loading paper, 14-16
Troubleshooting, 99-102
Type punch (IBM mode), 26-27, 60-62

User's Manual Statement

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 the Federal Communications Commission's rules, which are designed to provide reasonable protection against radio and television interference in a residential installation. If not installed properly, in strict accordance with the manufacturer's instructions, it may cause such interference. If you suspect interference, you can test this equipment by turning it off and on. If this equipment does cause interference, correct it by doing any of the following:

- Reorient the receiving antenna or AC plug.
- Change the relative positions of the computer and the receiver.
- Plug the computer into a different outlet so the computer and receiver are on different circuits.

CAUTION: Only peripherals with shield-grounded cables (computer input-output devices, terminals printers, etc.) certified to comply with Class B limits, can be attached to this printer. Operation with non-certified peripherals is likely to result in communications interference.

Your house AC wall receptacle must be a three-pronged type (AC ground). If not, contact an electrician to install the proper receptacle. If a multi-connector box is used to connect the computer and peripherals to AC, the ground must be common to all units.

If necessary, consult your Commodore dealer or an experienced radio-television technician for additional suggestions. You may find the following FCC booklet helpful: "How to Identify and Resolve Radio-TV Interference Problems." The booklet is available from the U.S. Government Printing Office, Washington, D.C. 20402, stock no. 004-000-00345-4.

COMMODORE 

Commodore Business Machines, Inc.
1200 Wilson Drive • West Chester, PA 19380

Commodore Business Machines, Limited
3470 Pharmacy Avenue • Agincourt, Ontario, M1W 3G3

319795-02 Rev. B
Y56399102001

Printed in Japan