

THE FUNCTION CODES

This chapter explains the purpose, appearance, and use of the functional command codes. After you've studied this chapter, you should be able to make full use of the quick reference chart provided on the inside back cover of this manual. In addition, Appendix M provides a listing of the codes, in the same order as the reference chart, with a short description and decimal (10-base) and hexadecimal (16-base) equivalents.

The format of these descriptions is as follows:

- PURPOSE:** Tells what the function code accomplishes.
- FORMAT:** Shows the correct format for the control sequence, giving both the alphanumeric and ASCII code sequences.
- REMARKS:** Details how the command is used.
- EXAMPLE:** Shows BASIC program segment to demonstrate the use of the command.

COMMANDS USED TO CONTROL PRINT STYLE

These commands are used to control the font style, the print pitch, print mode, and special effects.

Font Style Controls

- PURPOSE:** *Select the standard character set.*
- FORMAT:** ESC "5"
CHR\$(27) CHR\$(53)
- REMARKS:** This command causes the printer to cancel the italic character set and select the standard character set.
- EXAMPLE:** LPRINT CHR\$(27);CHR\$(53);



PURPOSE: **Select the italic character set.**
FORMAT: ESC "4"
CHR\$(27) CHR\$(52)
REMARKS: This command causes the printer to select the italics character set.
EXAMPLE: LPRINT CHR\$(27);CHR\$(52);

PURPOSE: **Select the international character set.**
FORMAT: ESC 7 n
CHR\$(27) CHR\$(55) CHR\$(n)
REMARKS: This command causes the printer to select the international character set as shown below:
n = 0: USA n = 1: England
n = 2: Germany n = 3: Denmark
n = 4: France n = 5: Sweden
n = 6: Italy n = 7: Spain
EXAMPLE: LPRINT CHR\$(27); CHR\$(55); CHR\$(4)
... selects the international character set for France.

Font Pitch Controls

PURPOSE: **Set the print pitch.**
FORMAT: ESC "B" n
CHR\$(27) CHR\$(66) CHR\$(n)
REMARKS: After this command is sent to the printer, subsequent printing is done: in pica (10 characters per inch) if n = 1; in elite (12 characters per inch) if n = 2; in condensed (17 characters per inch) if n = 3.
EXAMPLE: FOR N = 1 TO 3:
 LPRINT CHR\$(27);"B"; CHR\$(N);
 FOR I = 32 TO 112: LPRINT CHR\$(I);: NEXT I
 LPRINT: NEXT N
... prints out a selection of print in each of the three print pitches.



- PURPOSE:** *Set the print pitch to condensed print.*
- FORMAT:** SI
CHR\$(15)
- REMARKS:** A single-character command to cause the printer to shift into condensed print mode. This command is cancelled by the transmission of the DC2 code CHR\$(18).
-
- PURPOSE:** *Cause the printer to print in double-width mode.*
- FORMAT:** SO or ESC "W" 1
CHR\$(14) or CHR\$(27) CHR\$(87) CHR\$(1)
- REMARKS:** After this code is sent, subsequent data are converted to 5 CPI, 12 CPI pitch is converted to 6 CPI, and 17 CPI is converted to 8.5 CPI. The SO command is cancelled by the transmission of the DC4 code CHR\$(20) or by the CR code CHR\$(13). The ESC "W" 0 code will also cancel this command, causing the printer to print in normal mode.
- EXAMPLE:** LPRINT CHR\$(14)
LPRINT "Double-Width Mode"
LPRINT "Standard-Width Mode"
... prints the first line in double-width mode and the second line in standard width since the carriage return transmitted by the first LPRINT cancels the double-width mode.
-
- PURPOSE:** *Set the print pitch to condensed print mode.*
- FORMAT:** ESC SI
CHR\$(27) CHR\$(15)
- REMARKS:** Same as the SI code. Please refer to the SI code.
-
- PURPOSE:** *Cause the printer to print in enlarged mode.*
- FORMAT:** ESC SO
CHR\$(27) CHR\$(14)
- REMARKS:** Same as the SO code. Please refer to the SO code.



- PURPOSE: ***Set the print pitch to pica-sized mode.***
FORMAT: DC2
CHR\$(18)
REMARKS: A single-character command to cause the printer to shift into pica-sized mode. This command is cancelled by the transmission of the ESC B 2 code, or ESC B 3, or SI code.
- PURPOSE: ***Cause the printer to print in normal mode.***
FORMAT: DC4
CHR\$(20)
REMARKS: A single-character command to cancel the SO code.

Special Print Modes

- PURPOSE: ***Print characters in double-strike printing mode.***
FORMAT: ESC "G"
CHR\$(27) CHR\$(71)
REMARKS: When the ESC "G" code is transmitted to the printer, subsequent characters are printed in the double-strike mode. The double strike is cancelled when the ESC "H" code is sent to the printer — CHR\$(27) CHR\$(72). When in double strike, the characters are printed once, then the paper is moved up 1/144 inch and the characters are reprinted. Shifting in and out of double-strike on the same line will cause the line to slant slightly. The double-printing mode can be used with any other print-style commands.
- EXAMPLE: LPRINT CHR\$(27);"G";
 "This is double-strike print"
LPRINT CHR\$(27);"H";
 "This is standard print"
- PURPOSE: ***Print in emphasized printing mode.***
FORMAT: ESC "E"
CHR\$(27) CHR\$(69)



REMARKS: When the ESC "E" code is sent to Gemini, all subsequent characters are printed in the emphasized print mode. The emphasized print code is cancelled by sending an ESC "F" code to the printer — CHR\$(27) CHR\$(70). Emphasized printing can only be used with the normal pica and double-width pica characters (10 CPI and 5 CPI) and cannot be used with sub/superscripts. The emphasized print mode can be combined with double-strike mode.

Special Printing Effects

PURPOSE: *Print characters with underline.*

FORMAT: ESC "-" 1
CHR\$(27) CHR\$(45) CHR\$(1)

REMARKS: When the ESC "-" 1 code is sent to the printer, all subsequent characters are printed with an underline. The underline mode is cancelled by the ESC "-" 0 code — CHR\$(27) CHR\$(45) CHR\$(0).

PURPOSE: *Print in superscript mode.*

FORMAT: ESC "S" 0
CHR\$(27) CHR\$(83) CHR\$(0)

REMARKS: When the ESC "S" 0 command is received by the printer, the subsequent characters are printed in superscript mode. While in superscript, the normal bidirectional print mode is cancelled and the unidirectional print mode is invoked. Superscripts may be used with the italic font and in pica, elite, and compressed pitches. They may not be used in double-width print mode or with emphasized printing mode. Superscripts are always printed in double-strike. Superscript mode is cancelled by sending the ESC "T" code CHR\$(27) CHR\$(84). This code also cancels the unidirectional print mode, but does not cancel the double-strike printing mode.



PURPOSE: *Print in subscript mode.*

FORMAT: ESC "S" 1
CHR\$(27) CHR\$(83) CHR\$(1)

REMARKS: When the ESC "S" 1 command is received, subsequent characters are printed in the subscript style. See the remarks for superscript mode for additional information.

EXAMPLE: FOR I = 48 TO 122
LPRINT CHR\$(I);CHR\$(27);"S";CHR\$(0);
CHR\$(I);CHR\$(27);
"T";CHR\$(27);"S";CHR\$(1);CHR\$(I);
CHR\$(27);"T"
NEXT I
... prints the upper and lowercase characters in normal superscript and subscript style.

PURPOSE: *Print in unidirectional printing mode.*

FORMAT: ESC "U" 1
CHR\$(27) CHR\$(85) CHR\$(1)

REMARKS: When the ESC "U" 1 code is sent to the printer, all subsequent lines are printed in unidirectional printing mode. This mode is invoked by the super/subscript mode and by the graphics mode. Unidirectional printing is useful in printing tables and charts and ensures that columnar numbers and other characters are in alignment. The ESC "U" 0 code is sent to the printer to cancel unidirectional printing and return to the faster (standard) bidirectional printing mode.

EXAMPLE: LPRINT "Bidirectional printing"
FOR I = 1 TO 10: LPRINT TAB(25);"\$1.29 I":
NEXT I
LPRINT "Unidirectional printing";
CHR\$(27);"U";CHR\$(1)
FOR I = 1 TO 10:
LPRINT TAB(25);"\$1.29 I": NEXT I
LPRINT CHR\$(27);"U";CHR\$(0);



COMMANDS USED TO CONTROL VERTICAL POSITION OF THE PRINT HEAD

These commands are used to control or move the paper relative to the location of the print head. Moving the paper "up" has the effect of moving the print head "down" the page.

Line Feed Controls

PURPOSE: *Advance the paper by one line (line-feed command).*

FORMAT: LF
CHR\$(10)

REMARKS: The line feed is set to 1/6 inch if DIP switch 1-5 is "on" upon "power up" or printer reset. If DIP switch 1-5 is "off" the line feed will be 1/8 inch. The size of the line feed can be changed using the codes listed below. When DIP switch 2-4 is "on," a line feed is automatically generated when the printer receives a CR code CHR\$(13).

PURPOSE: *Change the line feed to a pre-set value.*

FORMAT: ESC "n"
CHR\$(27) CHR\$(m)

REMARKS: When this command is sent to the printer, the size of the line feed is changed according to the value of "n" (CHR\$(m)).

For "n" = "0" (m = 48),
successive line feeds become 1/8.

For "n" = "1" (m = 49),
successive line feeds become 7/72.

For "n" = "2" (m = 50),
successive line feeds become 1/6.

EXAMPLE: LPRINT CHR\$(27);"0"
...changes the line-feed size to 1/8 inch, then causes a carriage return and line-feed commands to be sent to the printer.



PURPOSE: **Change the line feed to $n/72$ inches.**

FORMAT: ESC "A" n
CHR\$(27) CHR\$(65) CHR\$(n)

REMARKS: When this command is transmitted to the printer, successive line-feed commands cause the printer to advance the paper $n/72$ inch. The value of n must be between 1 and 127.

EXAMPLE: LPRINT CHR\$(27);"U";CHR\$(1);
 CHR\$(27);"A";CHR\$(6);
 FOR J = 1 TO 66: FOR I = 1 TO 80
 LPRINT CHR\$(239);: NEXT I:
 LPRINT: NEXT J

... causes the printer to fill a page with dots by using the special block-graphics character and setting the line feed to 1/12 inch.

PURPOSE: **Change the line feed to $n/144$ inches.**

FORMAT: ESC "3" n
CHR\$(27) CHR\$(51) CHR\$(n)

REMARKS: When this command is sent to the printer, successive line feeds are set to $n/144$ inches. The value of n must be between 1 and 127.

PURPOSE: **Send a one-time-only line feed of $n/144$ inches.**

FORMAT: ESC "J" n
CHR\$(27) CHR\$(74) CHR\$(n)

REMARKS: When this code is sent to the printer, it causes the paper to be advanced by $n/144$ inches. The value of n must be between 1 and 127. This command does not change the size of the existing line-feed value.

Form Feed Controls

PURPOSE: **Advance the paper to the next top of form (TOF).**

FORMAT: FF
CHR\$(12)



REMARKS: When this command is received by the printer, it causes the paper to be advanced to the top of the next page or form. The initial top of form is set upon power "on." The default value of the page is set to 66 lines, but can be changed using the ESC "C" n and the ESC "C" 0 n code.

PURPOSE: **Change the setting of the page (form) to n lines.**

FORMAT: ESC "C" n
CHR\$(27) CHR\$(67) CHR\$(n)

REMARKS: When this code is received, the length of the page is set to n lines. The value of n should be between 1 and 127.

PURPOSE: **Change the setting of the form length to n inches.**

FORMAT: ESC "C" 0 n
CHR\$(27) CHR\$(67) CHR\$(0) CHR\$(n)

REMARKS: When this code is received, the length of the page is set to n inches. The values of n must be between 1 and 32.

PURPOSE: **Change the line location of the header (first line) of the form.**

FORMAT: ESC "R" n
CHR\$(27) CHR\$(82) CHR\$(n)

REMARKS: This command is used to change the location of the first line on the next page or form. The location of the first line is given by the value of n , and must be between 1 and 16. The default value set upon "power up" or printer reset is $n = 1$.

PURPOSE: **Set and enable the skip-over perforation feature.**

FORMAT: ESC "N" n
CHR\$(27) CHR\$(78) CHR\$(n)



REMARKS: When this command is received, the skip-over perforation feature is activated. This feature causes the printer to skip to the next page or form, when the number of lines left on the page is equal to n . If $n = 0$, the skip-over form feature is deactivated. The value of n should be between 0 and 255.

EXAMPLE: LPRINT CHR\$(27);"C";CHR\$(88);
 CHR\$(27);"0";
LPRINT CHR\$(27);"R";CHR\$(8);
 CHR\$(27);"N";CHR\$(16);

... these commands allow you to print 64 lines per page. The first line starts 1 inch from the top, and the last line ends 2 inches from the bottom of the page. The line spacing is 1/8 inch.

PURPOSE: **Cancel the skip-over perforation feature.**

FORMAT: ESC "O"
 CHR\$(27) CHR\$(79)

REMARKS: When this command is received, the skip-over perforation set by "ESC N n " code and the line location of the header of the form set by "ESC R n " code are cancelled.

Vertical Tabs

PURPOSE: **Advance paper to the next vertical tab.**

FORMAT: VT
 CHR\$(11)

REMARKS: When this command is received, the paper is advanced to the next vertical tab position. The vertical tab positions are set upon "power up" or printer reset to lines 6, 12, 18, 24, 30, 36, 42, 48, 54, and 60. The vertical tab positions can be changed using the ESP "P" $nn...0$ code.

PURPOSE: **Change the position of the vertical tabs.**

FORMAT: ESP "P" $n1 n2 n3...0$
 CHR\$(27) CHR\$(80) CHR\$($n1$) CHR\$($n2$)
 CHR\$($n3$)...CHR\$(0)



REMARKS: When the ESC "P" . . . code is sent to the printer, the existing vertical tab positions are cancelled. The new vertical tabs are set to line n_1 , n_2 , n_3 , etc. The CHR\$(0) character is used to end the command. The value of n should be between 1 and 255, and n_1 should be less than n_2 , n_2 less than n_3 , etc.

EXAMPLE: LPRINT CHR\$(13);CHR\$(27);"P";CHR\$(15);
CHR\$(30);CHR\$(0);
FOR I = 1 TO 2: LPRINT CHR\$(11);
I: NEXT I
. . . advances the paper to the top of form and prints the numeral 1 on line 15 and the numeral 2 on line 30.

PURPOSE: **Send a one-time-only vertical line feed of n lines.**

FORMAT: ESC "a" n
CHR\$(27) CHR\$(97) CHR\$(n)

REMARKS: When this command is sent, it causes the paper to be advanced n lines vertical feed. Value of n must be between 1 and 127. This command does not change the present vertical tab.

EXAMPLE: LPRINT CHR\$(27);CHR\$(97);CHR\$(15)
. . . causes the paper to advance 15 vertical lines.

COMMANDS TO CONTROL THE HORIZONTAL POSITION OF THE PRINT HEAD

These commands are used to control the horizontal position of the print head.

PURPOSE: **Return the print head (carriage) to the next print position.**

FORMAT: CR
CHR\$(13)



REMARKS: When this command is sent, the print head moves to the next print position. If DIP switch 2-2 is set to "off," this carriage return command also causes the printer to empty the print buffer. If Dip switch 2-4 is set to "on," the command causes a line-feed command to be inserted after the carriage-return code.

PURPOSE: **Set the left-hand print margin.**

FORMAT: ESC "M" *n*
CHR\$(27) CHR\$(77) CHR\$(*n*)

REMARKS: After this command is sent to the printer, successive carriage return codes will cause the print head to return to the *n* print position. The value of *n* should be between 1 and 255. The maximum number of print positions on an 8-inch-wide form is 80 for 10 CPI characters, 96 for 12 CPI, and 136 for 17 CPI. For the Gemini-15X the maximum number of print positions is 136 for 10 CPI, 163 for 12 CPI, and 233 for 17 CPI.

PURPOSE: **Set the right-hand print margin.**

FORMAT: ESC "Q" *n*
CHR\$(27) CHR\$(81) CHR\$(*n*)

REMARKS: After this command is sent to the printer, attempting to print beyond the *n* print position will cause the printer to automatically perform a line feed and return before printing the remainder of the line. The value of *n* should be between 1 and 255.

PURPOSE: **Move the print head to the next horizontal tab location.**

FORMAT: HT
CHR\$(9)

REMARKS: When the HT command is received, the print head is moved to the next horizontal tab



position. Upon power "on," the horizontal tab locations are set to columns 10, 20, 30, etc.

PURPOSE: **Set the horizontal tab positions.**

FORMAT: ESC "D" *n1, n2, n3* . . . 0
CHR\$(27) CHR\$(68) CHR\$(*n1*)
CHR\$(*n2*) . . . CHR\$(0)

REMARKS: When the ESC "D" . . . command is sent to the printer, the existing horizontal tab positions are cancelled. The new horizontal tab positions are set to print column *n1, n2, n3*, . . . The maximum number of horizontal tabs is 255. The CHR\$(0) code must be at the end of this command. The value of *n1* must be less than *n2*, *n2* must be less than *n3*, etc.

EXAMPLE: LPRINT CHR\$(27);"M";CHR\$(5);CHR\$(27);
"Q";CHR\$(70);
LPRINT CHR\$(27);"D";CHR\$(10);CHR\$(30);
CHR\$(0)

. . . sets the left-hand margin at print column 5, sets the right-hand margin at print column 70, and sets the two horizontal tab positions: one at print column 10, and the other at print column 30.

PURPOSE: **Move the print head back one space.**

FORMAT: BS
CHR\$(8)

REMARKS: When the BS code is sent to the printer, the print head is shifted one print column to the left. The BS code can be used to overprint characters.

PURPOSE: **Send a one-time-only horizontal tab of *n* columns.**

FORMAT: ESC b *n*
CHR\$(27) CHR\$(98) CHR\$(*n*)



REMARKS: This command enables the print head to shift n columns to the right. The value must be between 1 and 127. This does not change the present horizontal tab.

EXAMPLE: LPRINT CHR\$(27);CHR\$(98);CHR\$(45)
... shifts the print head 45 columns to the right.

COMMANDS USED TO CONTROL GRAPHICS

PURPOSE: **Print low-resolution (normal-density) graphics.**

FORMAT: ESC "K" $n1 n2 m1 m2$...
CHR\$(27) CHR\$(75) CHR\$($n1$) CHR\$($n2$)
CHR\$($m1$) CHR\$($m2$)...

REMARKS: This command selects the 60-dots-per-inch, column-scan, bit-image graphics mode. The number of columns to be printed is given by $n1 + 256 * n2$. There must be $n1 + 256 * n2$ characters following $n2$. These characters determine which pins are fired according to the explanation found in Chapter 6 for dot graphics.

PURPOSE: **Print high-resolution (double-density) graphics.**

FORMAT: ESC "L" $n1 n2 m1 m2$...
CHR\$(27) CHR\$(76) CHR\$($n1$) CHR\$($n2$)
CHR\$($m1$) CHR\$($m2$)...

REMARKS: This command selects the 120-dots-per-inch graphics mode. See remarks for the ESC "K"... code for additional details.

PURPOSE: **Print high-resolution (double-density) graphics with double speed.**

FORMAT: ESC "y" $n1 n2 m1 m2$...
CHR\$(27) CHR\$(121) CHR\$($n1$) CHR\$($n2$)
CHR\$($m1$) CHR\$($m2$)...



- REMARKS: This command selects the 120-dots-per-inch graphics mode but with double the print speed. See ESC "K" . . . code for additional details.
- PURPOSE: **Print quadruple-density graphics.**
- FORMAT: ESC "z" n1 n2 m1 m2 . . .
CHR\$(27) CHR\$(122) CHR\$(n1) CHR\$(n2)
CHR\$(m1) CHR\$(m2) . . .
- REMARKS: This command selects the 240-dots-per-inch column scan bit image graphics mode. See ESC "K" . . . code for additional details.

COMMANDS USED FOR MACRO INSTRUCTION

These commands are used to control the macro instruction.

- PURPOSE: **Define the macro instruction.**
- FORMAT: ESC "+" . . . RS
CHR\$(27) CHR\$(43) . . . CHR\$(30)
- REMARKS: When this code is sent to the printer, the existing macro instruction is cancelled. The new macro instruction is set. The maximum number of instruction bytes is 16. The RS(CHR\$(30)) character is used to end the command.
- PURPOSE: **Select the macro instruction.**
- FORMAT: ESC "!"
CHR\$(27) CHR\$(33)
- REMARKS: When this code is sent to the printer, the macro instruction set by "ESC + . . . RS" code is selected.

COMMANDS USED TO DOWNLOAD CHARACTERS

These commands are used to control the download characters. For details, please refer to Chapter 8.



PURPOSE: **Define the download character into RAM.**

FORMAT: ESC "*" 1 *n1 n2 m1 m2* . . .
CHR\$(27) CHR\$(42) CHR\$(1) CHR\$(*n1*)
CHR\$(*n2*) CHR\$(*m1*) CHR\$(*m2*) . . .

REMARKS: When you define the download character, character patterns (*m1 m2* . . .) are inputted with 9 bytes into RAM after transmitting the "ESC * 1" code, "*n1*" as designating position to be written, and "*n2*" as the descender data. The value of *n1* must be between 32 and 126. The value of *n2* must be "1" when the character is to be shifted two dots downward, and "0" when it is not shifted.

PURPOSE: **Copy the fonts in character ROM into download RAM.**

FORMAT: ESC "*" 0
CHR\$(27) CHR\$(42) CHR\$(0)

REMARKS: When this command is received, the fonts in character ROM are copied into the download character RAM.

PURPOSE: **Select the download character set.**

FORMAT: ESC "\$" 1
CHR\$(27) CHR\$(36) CHR\$(1)

REMARKS: This command causes the printer to select the download character set. Download character sets cannot be intermixed with other character sets on the same line.

PURPOSE: **Cancel the download character set.**

FORMAT: ESC "\$" 0
CHR\$(27) CHR\$(36) CHR\$(0)

REMARKS: This command causes the printer to cancel the download character set and select the standard ASCII character set.



OTHER FUNCTION CODES

PURPOSE: *Control the value of the eight bits of data sent to printer.*

FORMAT: ESC ">" or ESC "=" or ESC "#"
CHR\$(27) CHR\$(62) or CHR\$(27) CHR\$(61) or
CHR\$(27) CHR\$(35)

REMARKS: These codes are used to control the value of the eighth bit of the data that is sent to the printer. After the ESC ">" code is received by the printer, all subsequent data is processed with the eighth regarded as set to 1. The ESC "#" code is used to cancel the ESC ">" command. After the ESC "=" code is received, all subsequent data is processed with the eighth bit regarded as set to 0. The ESC "#" code is used to cancel the ESC "=" command. These codes are provided to allow users with a 7-bit interface to access the special characters whose ASCII code is greater than 127. These codes should be used to transmit printer function codes. ESC ">" only functions with the DIP switch number 2-3 set to "on," for a 7-bit interface.

PURPOSE: *Delete the last character sent to the printer.*

FORMAT: DEL
CHR\$(127)

REMARKS: When the CHR\$(127) code is sent to the printer, it causes the printer to delete the last character it received, with the exception of function codes.

PURPOSE: *Put the printer in "online" mode.*

FORMAT: DC1
CHR\$(17)

REMARKS: This code cancels the DC3 code and puts the printer back in the "online" mode.



- PURPOSE:** *Put printer in "offline" mode.*
- FORMAT:** DC3
CHR\$(19)
- REMARKS:** When this code is sent to the printer, Gemini assumes the "offline" mode. All subsequent characters and control codes sent to the printer are disregarded. The only control code that is processed is CHR\$(17)(DC1), which cancels CHR\$(19), and puts the printer back "on line."
-
- PURPOSE:** *Activate the Gemini "buzzer."*
- FORMAT:** BEL
CHR\$(7)
- REMARKS:** When the CHR\$(7) code is sent to Gemini, the result is that the built-in buzzer "beeps" for one-fourth of a second. The buzzer function can be temporarily deactivated by sending an ESC "Y" 0 (CHR\$(27) CHR\$(89) CHR\$(0) control code) to the printer. After the ESC "Y" 0 code is received, the CHR\$(7) no longer causes the buzzer to "beep." The buzzer can be reactivated by sending an ESC "Y" 1 (CHR\$(27) CHR\$(89) CHR\$(1) code) to the printer.
-
- PURPOSE:** *Allow printing even though the "paper-out" detector indicates Gemini is out of paper.*
- FORMAT:** ESC "8"
CHR\$(27) CHR\$(56)
- REMARKS:** When the ESC "8" function code is sent to Gemini, the signal from the "paper-out" detector is disregarded. This code allows you to print on a single sheet. If the "paper-out" signal were not disregarded, Gemini would stop printing part way down the single sheet because the printer would "think" it's out of paper. The "paper-out" signal can be restored by sending an ESC "9" function code (CHR\$(27) CHR\$(57)).



PURPOSE: ***Regard the signal from the "paper-out" detector.***

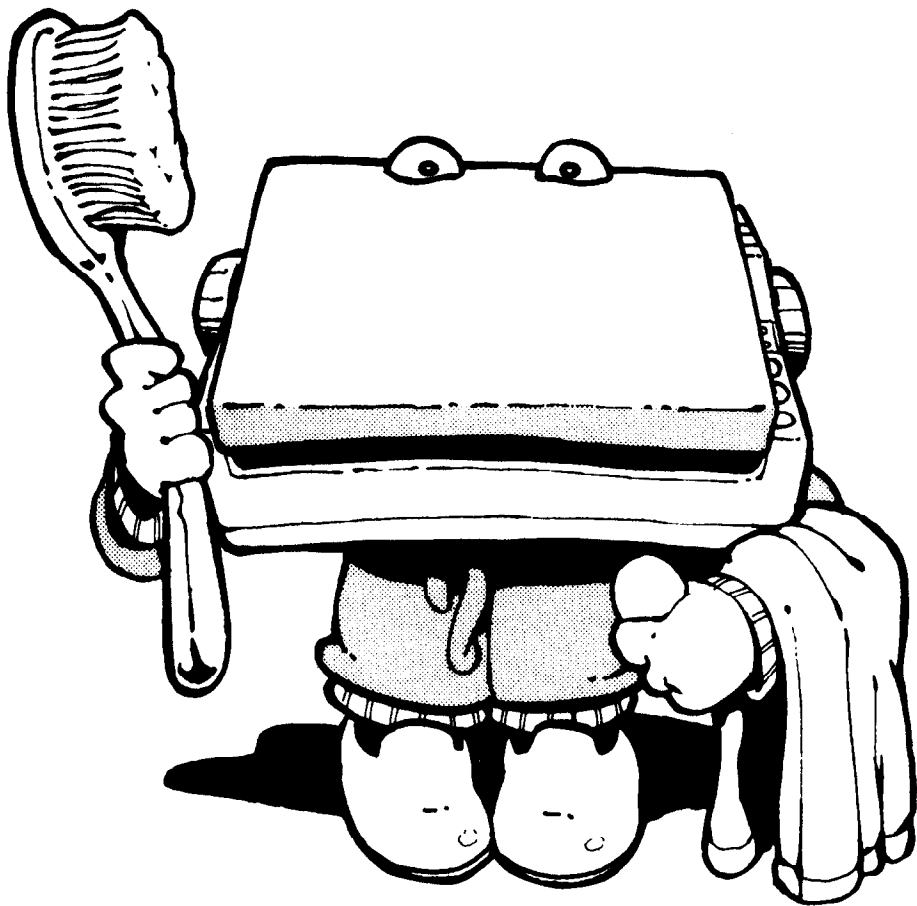
FORMAT: ESC "9"
CHR\$(27) CHR\$(57)

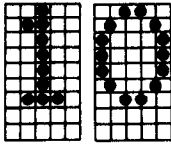
REMARKS: When this code is sent to the printer, the signal from the "paper-out" detector is restored.

PURPOSE: ***Initialize the printer conditions.***

FORMAT: ESC "@"
CHR\$(27) CHR\$(64)

REMARKS: When this code is sent to the printer, the conditions of the printer are initialized.





MAINTENANCE

The purpose of “maintenance” is to maximize the life of your Gemini. Toward that end, the best maintenance is “preventive.”

The Gemini printer is a well-engineered peripheral, and it serves as a valuable information-handling tool when used correctly. Part of using Gemini correctly is paying attention to the operating environment. Chapter 1 outlines the suggested considerations for keeping that environment clean.

Periodic cleaning of the exterior and interior of your printer will help keep Gemini going for years. How often you need to clean depends on how “dirty” the operating environment is.

CLEANING GEMINI

Exterior cleaning: When dust, grease, or other material accumulates on the outside of your printer, wipe the external surfaces of the case carefully with a clean cloth moistened with alcohol.

Note: Exercise the utmost care not to wet the electronic parts and machine body.

Interior cleaning: Remove dust and dirt from the internal mechanism by gently using a soft brush. Be very careful not to damage the electronic parts or wiring circuit.

To clean the interior of Gemini, you should remove the upper casing, as described below.

REMOVING THE UPPER CASING

Warning: Unplug your Gemini before removing the upper casing.

When you need to set Dip Switch number 1 or change other parts of the printer, the upper casing must first be removed. Figure 10-1 shows the position of the three screws that hold the upper casing in place on the Gemini-15X. Figure 10-2 shows the location of the two screws that do the same for the Gemini 10-X. In both cases, the mounting screws are Phillips-head types.



To replace the fuse, turn the power switch "off." Just to be on the safe side, make sure you unplug the power cord from the power source. Then remove the platen knob, take out the mounting screws, and remove the upper casing. To remove the upper casing, place your fingers on the inside front of the printer with your thumbs pressing on the outside front of the lower casing. Gently pull the upper-casing front toward you until it pops free. Set the upper-casing front on end (next to the printer) so that the short wires attached to the control panel remain connected.

After removing the casing, check the fuse, which is located on the fuse board near the power switch. The fuse is a commonly used type, with a metal strip suspended in a glass and metal case. If the strip is broken, the fuse is "blown."

If the fuse is blown, replace it with a BELL's 5MT1.25 (1.25A, 125V) "slo-blo" type fuse, or equivalent for 120V version, or a 630mA/250V "slo-blo" type fuse for 220V/240V version. Just slip the old fuse out and snap the replacement fuse into place in the two clasps (as shown in Figure 10-3).

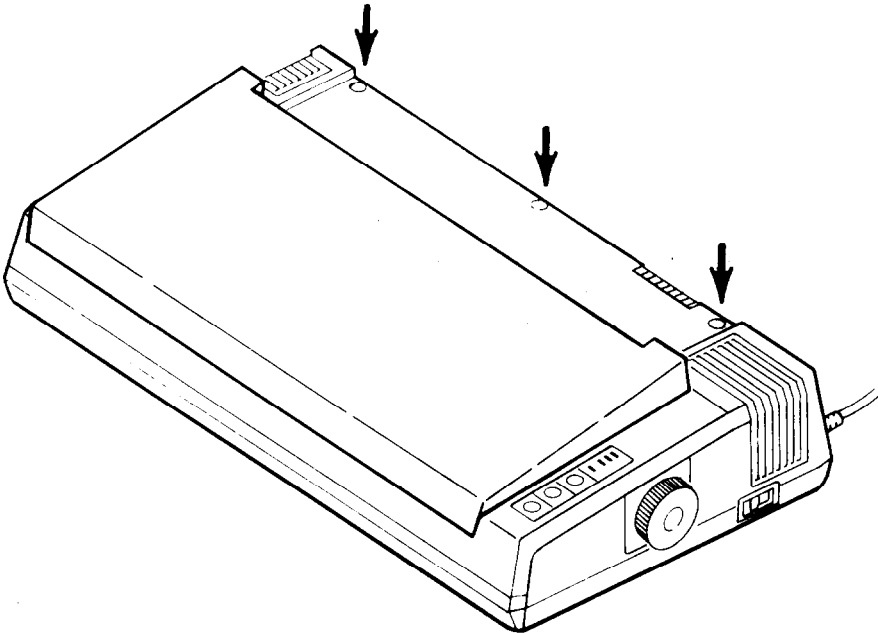


Figure 10-1. Three mounting screws hold the Gemini-15X upper casing in place.

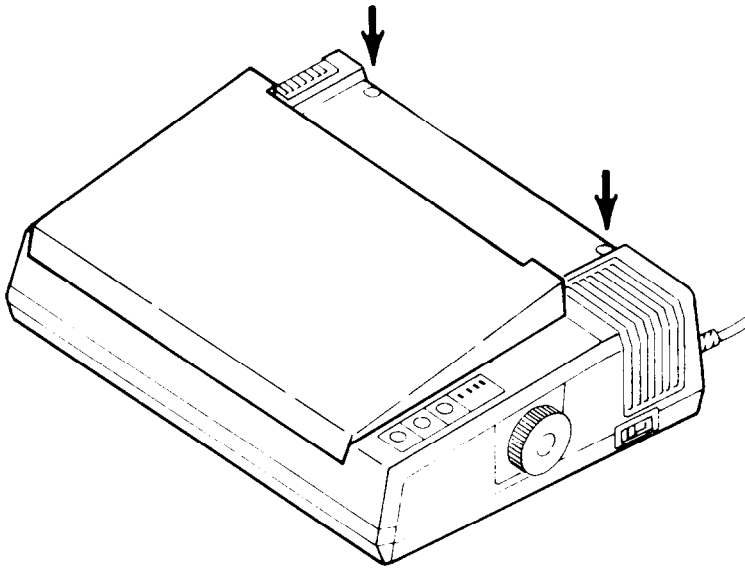


Figure 10-2. Two mounting screws hold the upper casing in place on the Gemini-10X.

After replacing (or just checking) the fuse, re-attach the upper casing (reversing the procedure described above) and insert the mounting screws and the platen knob. (Refer to Figure 1-6 in Chapter 1 for an “exploded view” of Gemini-10X with its upper casing removed.)

You probably need to replace the fuse when all of the following four statements apply to your situation:

- the printer won't operate,
- the power-on lamp isn't lit on the control panel,
- you are sure that the power switch is “on”, and
- the printer is connected to a power source.

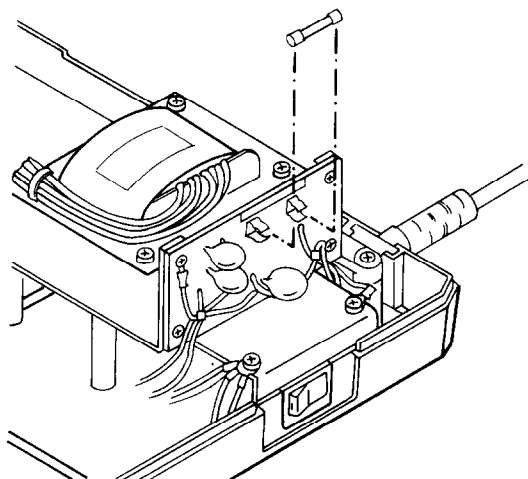


Figure 10-3. Replacing a fuse is simple, but make sure you use the correct type fuse.

REPLACING THE PRINT HEAD

When you replace the print head, start by removing the ink ribbon. You might want to refer to Figure 10-4 as you follow these steps:

Warning: The print head will be hot during operation and soon afterward, so be careful not to burn yourself.

1. Extract the head cable from the head connector. (Hold down the head cable board while performing this disconnection.)
2. Remove the two clamping screws from the print head.
3. Detach the print head.

4. Place the new print head in position, and attach it with the clamping screws. Apply "screw lock" (an adhesive) to the heads of the screws.
5. Insert the head cable tightly into the head connector. (Be sure the connection is a good one, otherwise it may cause problems.)

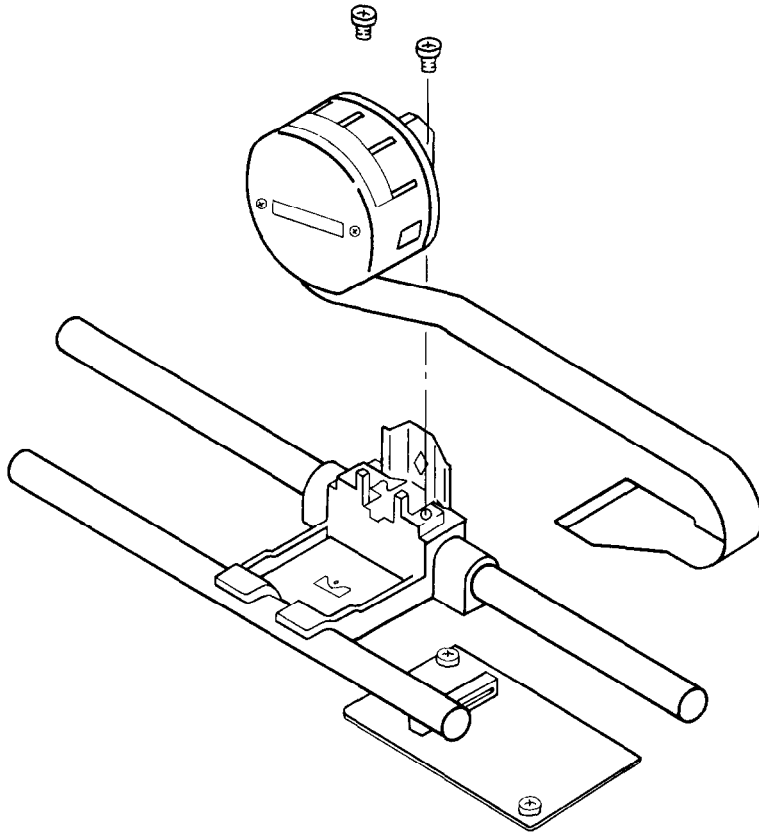


Figure 10-4. Replacing the print head by yourself is easy, quick, and avoids downtime.

PERIODIC INSPECTIONS

To keep Gemini in tip-top shape, you should make the following inspections every six months (under normal use). Expressed another way, you should "check under the hood" every half-million lines of printing. Here's what to check:

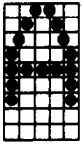
Maintenance Check List

- Clean the print head and sensor peripherals with a soft brush.
- Check the various screws and make sure they are tight. If they aren't, tighten them with a screwdriver.
- Rotate the platen, and check for possible damage. If the platen is worn or uneven, replace it.
- Remain alert to "abnormal" operating sounds. Strange noises can signal strange conditions, and the earlier you recognize and fix a problem, the better.
- Perform Self-Testing and check the movement of each mechanism.



Notes





APPLE II PLUS / IIE

The Interface Board

You need to acquire and install an Apple interface board (also called a "card") and a connecting cable. Your Gemini dealer can answer any questions you have about the board or its installation in slot number 1. Star Micronics, Inc. has created the **grafstar**[™] intelligent interface especially for you Apple II Plus or Apple IIE computers.

The Connecting Cable

Use the Apple, Centronics compatible, parallel I/F cable, modified to ground the 8th bit, or use a cable supplied by your Gemini dealer.

Setting the DIP Switches

Because Apple uses a seven-bit interface, set DIP switch number 2-3 to the "on" position. Because BASIC supplies a line feed at the end of each line, turn DIP switch number 2-4 to the "off" position (unless you want to generate "double-spaced" output, in which case turn number 2-4 "on"). Switch number 2-1 can be set to either position.

Switch number 2-2 should be set to "off" if you want Gemini to print every time it receives a CR code; set number 2-2 to "on" if you wish to have the carriage return ignored and to print only when the buffer is full.

The Use of Basic

In this manual, we used the BASIC programming language to illustrate how to control the codes that manipulate Gemini's operations. In our sample programs, the CHR\$ function is used to send control codes to the printer. Unfortunately, this function is not supported by Apple Integer BASIC. However, with the aid of the follow-



ing notes, you should be able to run all the sample programs presented in this manual if you use Applesoft BASIC.

Applesoft BASIC uses the PRINT command to send information to both the screen and the printer.

The PR # 1 command sends the information only to the printer (when the printer interface is in slot number 1), and the PR # 0 command sends the information only to the screen.

Therefore, you should use:

```
PRINT CHR$(4) "PR # 1"  
PRINT
```

instead of the LPRINT command. In addition, use:

```
PRINT CHR$(4) "PR # 0"  
PRINT
```

instead of the PRINT command.

The Apple video display is 40 columns wide. If you want the print mode to print more than 40 columns, you need to send an additional command code:

```
PRINT CHR$(4) "PR # 1"  
PRINT CHR$(9) "255N"  
PRINT
```

This message allows Gemini to print a line of any length. This is particularly important when sending bit-image graphics to Gemini.

The CHR\$(9) (or function code HT, which moves the print head to the next tab position) and CHR\$(13) (or function code CR for a carriage return) codes cause problems when used in both the Gemini escape-function code sequences and in the graphics mode. Avoid using these codes if possible.

The printer interface card does not allow us to send ASCII codes greater than 127. If you try to send these codes, the interface sets the eighth bit to zero. The result, for example, is that PRINT CHR\$(239) would be received by the printer as PRINT CHR\$(111); that is, $239 - 128 = 111$. Gemini provides the ESC ">" function code to allow 7-bit interface users to overcome this problem. See Chapter 9 for a discussion of this problem.

Apple users (and other 7-bit interface users) should also be aware of the difference between 7-bit and 8-bit interface graphics. See Chapter 6 for a discussion on this topic.



Pin Configurations

The following chart shows the pin-outs for Gemini-10X and Gemini-15X printers and the Apple II Plus, using a parallel interface board.

Gemini-10X/15X		Parallel	Apple Board	
25.	SG		1.	SG
26.	SG		2.	SG
27.	SG		3.	SG
1.	STROBE		4.	STROBE
28.	SG		5.	N/C
2.	D1		6.	D1
3.	D2		7.	D2
4.	D3		8.	D3
5.	D4		9.	D4
6.	D5		10.	D5
7.	D6		11.	D6
8.	D7		12.	D7
9	D8		13.	D8
10.	ACK		14.	ACK
29.	SG		15.	SG

Note: If you purchased a Star Micronic's "intelligent" Apple Interface Board, it comes complete with the cable. Just plug it in, and it's ready to go!

Program Listings

```

4 REM $$
5 REM $$ PRINTING EFFECTS PROG. $$
6 REM $$
10 PR# 1: PRINT CHR# (9) + "132N": PRINT
20 ESC$ = CHR# (27):LF$ = CHR# (10):FF$ = CHR# (12):CR$ = CHR# (13)
30 PICA$ = ESC$ + "B" + CHR# (1):ELITE$ = ESC$ + "B" + CHR# (12):COMP$ =
    ESC$ + "B" + CHR# (3)
40 SPR$ = ESC$ + "S" + CHR# (0):SUB$ = ESC$ + "S" + CHR# (1):CANS$ = ESC
    $ + "T"
50 ITALIC$ = ESC$ + CHR# (52)
55 STND$ = ESC$ + CHR# (53)
60 WIDE$ = CHR# (14):CWIDE$ = CHR# (20):PRSET$ = ESC$ + "0"
70 EMPH$ = ESC$ + "E":CEMPH$ = ESC$ + "F":DBLS$ = ESC$ + "G":CDBLS$ = ESC$
    + "H"
80 PRINT PRSET$:ESC$: "R": CHR# (15):FF$
90 FOR I = 1 TO 2

```



```
100 IF I = 1 THEN PRINT WIDE$;"      GEMINI STANDARD TYPE STYLES";CWIDE$
;LF$
110 IF I = 2 THEN PRINT ITALIC$;WIDE$;"      GEMINI ITALIC TYPE STYLES":
PRINT CWIDE$;LF$;LF$
120 PRINT ESC$;">": FOR N = 1 TO 40: PRINT CHR$ (231); CHR$ (232):: NEXT
N: PRINT ESC$;"=": PRINT : PRINT
130 PRINT "          17 CPI 12 CPI 10 CPI    8.5 CPI    6 CPI
5 CPI"
140 PRINT ESC$;">": FOR N = 1 TO 80: PRINT CHR$ (241):: NEXT N: PRINT ES
C$;"=": PRINT
150 M1$ = " ABC":M2$ = "X" + CHR$ (120):M3$ = "Y" + CHR$ (121)
160 FOR J = 1 TO 4
170 IF I = 2 THEN PRINT ITALIC$:
180 IF J = 1 THEN PRINT COMP$:"NORMAL      ";
190 IF J = 2 THEN PRINT COMP$:"DOUBLE STRIKE ";
200 IF J = 3 THEN PRINT COMP$:"EMPHASIZED  ";
210 IF J = 4 THEN PRINT COMP$:"DOUBLE STRIKE ";
220 FOR K = 1 TO 6
230 IF K = 1 THEN PRINT COMP$:
240 IF K = 2 THEN PRINT ELITE$:
250 IF K = 3 THEN PRINT PICA$:
260 IF K = 4 THEN PRINT COMP$;WIDE$:
270 IF K = 5 THEN PRINT ELITE$;WIDE$:
280 IF K = 6 THEN PRINT PICA$;WIDE$:
290 IF J = 2 THEN PRINT DBLS$:
300 IF J = 3 THEN PRINT EMPH$:
310 IF J = 4 THEN PRINT DBLS$;EMPH$:
320 PRINT M1$;SPR$;M2$;CANS$;SUB$;M3$;CANS$;CDBLS$:
330 IF K > 3 THEN PRINT CWIDE$:
340 NEXT K
345 IF J = 2 THEN PRINT LF$:
350 IF J = 4 THEN PRINT ESC$;"A"; CHR$ (8);LF$;COMP$;"      " + ";CR$;"EMPH
ASIZED ";
360 PRINT PRSET$
370 NEXT J
380 PRINT
390 NEXT I
400 PRINT FF$: PRINT CHR$ (9) + "80N": PR$ 0
410 END
```

Figure A-1 is the font styles program (refer to Figure 5-1) amended for the Apple II computer.



```

5 BL$ = "
7 LI$ = "-----"
10 HOME : DIM A$(22,40)
20 HOME : X = 1:Y = 1
30 GOSUB 150: VTAB X: HTAB Y: PRINT " ";
40 GET B$
50 IF B$ = CHR$(5) THEN HOME : END
60 IF B$ = CHR$(18) THEN HOME : GOSUB 200: GOTO 20
70 IF B$ = CHR$(16) THEN GOTO 210
80 IF B$ = CHR$(13) THEN A$(X,Y) = B$: VTAB X: HTAB Y: PRINT "":X = X +
    1:Y = 1: VTAB X: HTAB Y: PRINT "": GOTO 40
90 IF B$ = CHR$(4) THEN GOSUB 130: GOTO 40
100 VTAB X: HTAB Y: PRINT B$;A$(X,Y) = B$:Y = Y + 1
110 IF Y = 40 THEN A$(X,Y) = CHR$(13): PRINT CHR$(7);:X = X + 1:Y = 1
    : GOTO 40
120 VTAB X: HTAB Y: PRINT "": GOTO 40
130 VTAB X: HTAB Y: PRINT " ":Y = Y - 1: IF Y = 0 THEN Y = 1: PRINT CHR$(
    7);
135 A$(X,Y) = ""
140 VTAB X: HTAB Y: PRINT "": RETURN
150 VTAB 24: HTAB 1: INVERSE : PRINT " ^P)RINT";
160 VTAB 24: HTAB 10: PRINT " ^D)ELETE";
170 VTAB 24: HTAB 19: PRINT " ^R)ESTART";
180 VTAB 24: HTAB 29: PRINT " ^E)XIT ";
185 NORMAL
187 VTAB 22: HTAB 1: PRINT LI$;
190 RETURN
200 VTAB 10: HTAB 10: PRINT "PLEASE STAND BY"; FOR I = 1 TO X: FOR J = 1
    TO 40:A$(I,J) = " ": NEXT J: NEXT I: RETURN
210 REM
220 PR# 1: PRINT CHR$(27);"@": PR# 0
230 HOME : VTAB 24: HTAB 1: INVERSE : PRINT "C)OMMAND"; HTAB 10: PRINT "
    P)RINT";
240 HTAB 17: PRINT "R)ETURN";: NORMAL
250 GET G$
260 IF G$ = "P" THEN GOSUB 310
270 IF G$ = "C" THEN GOSUB 320: GOTO 230
280 IF G$ = "R" THEN GOSUB 300: GOTO 30
290 GOTO 250
300 HOME : FOR I = 1 TO X: FOR J = 1 TO 40: VTAB I: HTAB J: PRINT A$(I,J)
    ;: NEXT J: NEXT I: RETURN
310 PR# 1

```



```
311 PRINT CHR$(9);"40N"  
312 FOR I = 1 TO X: FOR J = 1 TO 40  
313 PRINT A$(I,J);: NEXT J: NEXT I  
314 PR# 0  
315 VTAB 24: HTAB 30  
317 RETURN  
320 VTAB 24: HTAB 1: PRINT BL$:  
330 REM  
340 VTAB 24: HTAB 1: INVERSE : PRINT "E) ESC";:  
350 HTAB 8: PRINT "C) CHR$";: HTAB 16: PRINT "Q) END COMMAND";  
355 NORMAL  
360 GET G$  
370 IF G$ = "E" THEN GOSUB 420: GOTO 400  
380 IF G$ = "C" THEN GOSUB 530: GOTO 400  
390 IF G$ = "Q" THEN GOTO 410  
400 GOTO 360  
410 RETURN  
420 I = 0: VTAB 22: HTAB 4: PRINT "ESC ";  
430 GET G$  
440 IF G$ = CHR$(13) THEN 480  
450 PRINT G$;  
455 IF I > 0 THEN 470  
460 PRINT " ";  
470 S$(I) = G$: I = I + 1: GOTO 430  
480 IF I > 2 THEN FOR J = 1 TO I: D$ = D$ + S$(J): NEXT J: ST$ = CHR$(27)  
  ) + S$(0) + CHR$( VAL (D$)): D$ = ""  
490 IF I = 2 THEN ST$ = CHR$(27) + S$(0) + CHR$( VAL (S$(1)))  
500 IF I = 1 THEN ST$ = CHR$(27) + S$(0)  
510 PR# 1: PRINT CHR$(9);"40N": PRINT ST$: ST$ = "": PR# 0: FOR J = 1 TO  
  I: S$(J) = "": NEXT J: I = 0  
520 VTAB 22: HTAB 4: PRINT BL$;: RETURN  
530 VTAB 22: HTAB 2: PRINT "CHR$ ";: ST$ = ""  
540 GET G$  
550 IF G$ = CHR$(13) THEN 570  
560 PRINT G$;: ST$ = ST$ + G$: GOTO 540  
570 VTAB 22: HTAB 2: PRINT BL$;:  
580 PR# 1: PRINT CHR$(9);"40N"  
590 PRINT CHR$( VAL (ST$));: ST$ = "": PR# 0: RETURN
```

Figure A-2 is the word processing program (refer to Figure 5-4) amended for the Apple II computer.



```

5 PR# 1
10 PRINT TAB( 30);"HIGH RESOLUTION VERTICAL SPACING"
15 PRINT TAB( 30);"USING THE ESC 3N CONTROL CODE"
20 FOR I = 1 TO 30
25 IF I = 7 THEN GOTO 110
26 IF I = 9 THEN GOTO 110
30 IF I = 13 THEN GOTO 110
40 M = INT (144 / (6 * I)); IF M < 2 THEN M = 2
50 PRINT CHR$( 27); CHR$( 51); CHR$( I);
60 FOR J = 0 TO M
70 IF J = M - 1 THEN PRINT I;"/144 INCH SPACING          "; GOTO 90
80 PRINT TAB( 30);
90 FOR N = 1 TO 40; PRINT CHR$( 27);">"; CHR$( 241); CHR$( 27);"=""; NEXT
   N; PRINT
100 NEXT J
110 NEXT I
120 PRINT CHR$( 27);"@ "
130 PR# 0; END

```

Figure A-3 is the vertical spacing program (refer to Figure 6-5) amended for the Apple II computer.

```

10 PR# 1
20 PRINT CHR$( 27);"@"; CHR$( 27);"A"; CHR$( 6);
30 FOR LI = 1 TO 2
40 PRINT CHR$( 27);"K"; CHR$( 114); CHR$( 0);
50 FOR SPACE = 1 TO 100
55 PRINT CHR$( 0);; NEXT SPACE
60 FOR COLM = 1 TO 14
70 READ C: PRINT CHR$( C);; NEXT COLM
80 PRINT
90 NEXT LI
100 DATA 14,16,34,64,78,78,64,64,78,78,64,34,16,14
110 DATA 224,16,232,68,34,18,18,18,18,34,68,232,16,224
120 PRINT CHR$( 27);"@ "
130 PR# 0
140 END

```

Figure A-4 is the dot graphics program presented in Chapter 6 (the one that puts a smile on your paper) amended for the Apple II computer.



```
10 PR# 1
20 PRINT CHR$ (27);">";
30 PRINT CHR$ (163); CHR$ (161); CHR$ (27);"A"; CHR$ (6)
40 PRINT CHR$ (162); CHR$ (160); CHR$ (27);"2"
50 PRINT CHR$ (27);"=": PR# 0: END
```

Figure A-5 is the circle program presented in chapter 7, amended for the Apple II computer.

```
10 PR# 1
30 PRINT CHR$ (32); CHR$ (27);">"; CHR$ (163); CHR$ (27);"A"; CHR$ (6)
40 PRINT CHR$ (160)
50 PRINT CHR$ (27);"="
60 PR# 0: END
```

Figure A-6 is the tilde program presented in Chapter 7 amended for the Apple II computer.

```
: PR# 1
10 PRINT TAB( 20);"N"
15 PRINT TAB( 20); CHR$ (27);">"; CHR$ (164); CHR$ (27);"="
20 PRINT TAB( 17);"W";
25 PRINT CHR$ (27);">"; CHR$ (166); CHR$ (27);"=";
30 PRINT "  "; CHR$ (27);">"; CHR$ (167); CHR$ (27);"=";"E"
35 PRINT TAB( 20); CHR$ (27);">"; CHR$ (165); CHR$ (27);"="
40 PRINT TAB( 20);"E"
45 PR# 0: END
```

Figure A-7 is the map program presented in Chapter 7 amended for the Apple II computer.

```
10 REM WY LARGE LETTERS
20 PR# 1
25 PRINT CHR$ (27);"U"; CHR$ (1);
30 PRINT CHR$ (27);"A"; CHR$ (6)
40 FOR J = 1 TO 9
50 FOR I = 1 TO 12
60 READ A
70 PRINT CHR$ (27);">"; CHR$ (A);
```



```

80 NEXT I
90 PRINT
100 NEXT J
110 DATA 239,224,224,224,239,224,224,224,224,224,224,224
111 DATA 239,224,224,224,239,224,224,224,224,224,224,224
112 DATA 239,224,224,224,239,224,239,224,224,224,239,224
113 DATA 239,224,239,224,239,224,234,233,224,234,233,224
114 DATA 239,234,239,233,239,224,224,239,224,239,224,224
115 DATA 239,239,224,239,239,224,224,234,239,233,224,224
116 DATA 234,233,224,234,233,224,224,224,239,224,224,224
117 DATA 224,224,224,224,224,224,224,234,233,224,224,224
118 DATA 224,224,224,224,224,224,239,224,224,224,224
120 PRINT CHR$(27);"@"
130 PR# 0: END

```

Figure A-8 is the "Wy" program (refer to Figure 7-1) amended for the Apple II computer.

```

1 REM **
2 REM ** BOAT PROG. **
3 REM **
10 PR# 1: PRINT CHR$(27);"A"; CHR$(6); CHR$(15)
12 PRINT CHR$(27);"U"; CHR$(1);
15 FOR K = 1 TO 3
16 PRINT CHR$(27);"B"; CHR$(K)
17 RESTORE
20 FOR J = 1 TO 9
30 FOR I = 1 TO 12
40 READ A
50 PRINT CHR$(27);" "; CHR$(A);
60 NEXT I
70 PRINT
80 NEXT J
90 DATA 224,224,224,224,224,224,224,224,224,245,224,224
91 DATA 224,224,224,224,224,224, 224,224,252,245,224,224
92 DATA 224,224,224,245,224,224, 224,252,239,245,224,224
93 DATA 224,224,252,245,224,224, 252,239,239,245,224,224
94 DATA 224,252,239,245,224,252, 239,239,239,245,224,224
95 DATA 252,239,239,245,252,239, 239,239,239,245,224,224

```



```
96 DATA 241,241,241,249,241,241, 241,241,241,249,224,224
97 DATA 239,239,239,239,239,239, 239,239,239,239,239,251
98 DATA 253,239,239,239,239,239, 239,239,239,239,251,224
99 PRINT : NEXT K
100 PRINT CHR$(27);"0": PR# 0: END
```

Figure A-9 is the boat program (refer to Figure 7-2) amended for the Apple II computer.

```
4 HOME : PRINT " ": PRINT " "
5 PRINT " ": PRINT " "
6 PRINT "THIS PROGRAM TAKES ABOUT"
7 PRINT "3 MINUTES TO RUN. PLEASE"
8 PRINT "TURN ON YOUR PRINTER AND"
9 PRINT "STAND BY....."
10 A = 24576
20 FOR I = A TO A + 12
30 READ B
35 POKE I,B
40 NEXT I
50 DATA 32,74,255,165,250,5,251
60 DATA 133,252,32,63,255,96
100 REM MICRO-PLOT
110 DIM BIT$(76,12)
1000 REM SET PROGRAM CONSTANTS
1010 MASK$(1) = 64:MASK$(4) = 8
1020 MASK$(2) = 32:MASK$(5) = 4
1030 MASK$(3) = 16:MASK$(6) = 2
1040 LX = 20:LY = 20
1050 XFAC = 72 / LX:YFAC = 75 / LY
2000 REM PLOT CURVE
2010 RAD = 9
2020 X1 = 19:Y1 = 10
2030 FOR ANG = 0 TO 360 STEP 10
2040 R1 = ANG * 6.28 / 360
2050 X2 = RAD * COS (R1) + 10:Y2 = RAD * SIN (R1) + 10
2060 GOSUB 4000
2070 NEXT ANG
2080 FOR ANG = 0 TO 360 STEP 15
```



```
2090 R1 = ANG * 6.28 / 360
2100 R2 = (ANG + 150) * 6.28 / 360
2110 X1 = RAD * COS (R1) + 10:Y1 = RAD * SIN (R1) + 10
2120 X2 = RAD * COS (R2) + 10:Y2 = RAD * SIN (R2) + 10
2130 GOSUB 4000
2140 NEXT ANG
2150 BIT%(39.0) = 4
3000 REM SEND BIT IMAGE MAP TO PRINTER
3005 PR# :
3006 PRINT CHR$(9):"255N"
3010 PRINT CHR$(27);"A": CHR$(6)
3015 PRINT CHR$(13)
3020 FOR ROW = 0 TO 11
3025 PRINT CHR$(27);"K": CHR$(119): CHR$(1);
3050 FOR COL = 1 TO 75
3055 FOR LL = 1 TO 5
3060 PRINT CHR$(BIT%(COL,ROW));
3065 NEXT LL
3070 NEXT COL
3080 PRINT
3090 NEXT ROW
3100 PRINT
3102 PR# 0
3110 END
4000 REM DRAW A LINE FROM X1,Y1 TO X2,Y2
4010 XL = X2 - X1:YL = Y2 - Y1
4020 NX = ABS (XL * XFAC):NY = ABS (YL * YFAC)
4030 IF NX < NY THEN NX = NY
4040 NS% = INT (NX + 1)
4050 DX = XL / NS%:DY = YL / NS%
4060 FOR I = 1 TO NS%
4070 X1 = X1 + DX:Y1 = Y1 + DY
4080 GOSUB 5000
4090 NEXT I
4100 RETURN
5000 REM PLOT A POINT AT X1,Y1
5010 XX = X1 * XFAC:YY = Y1 * YFAC
5020 COL = INT (XX) + 1
5030 ROW = INT (YY / 6)
5040 XIT% = INT (YY - (6 * ROW)) + 1
5042 POKE 250,BIT%(COL,ROW)
5044 POKE 251,MASK%(XIT%)
```



```
5046 CALL 24576
5050 BIT%(COL,ROW) = PEEK (252)
5060 RETURN
```

Figure A-10 is the graphics program (refer to Figure 7-4) amended for the Apple II computer.

```
10 PRINT CHR$(4);"PR#1"
20 REM *** DEFINE PARAGRAPH SYMBOL IN PLACE OF THE @ SYMBOL
30 PRINT CHR$(27); CHR$(42); CHR$(1); CHR$(64); CHR$(0); CHR$(6); CHR$(
  25); CHR$(0); CHR$(25); CHR$(118); CHR$(25); CHR$(0); CHR$(127);
  CHR$(0);
40 REM *** SELECT DOWNLOADABLE CHARACTERS
50 PRINT CHR$(27); CHR$(36); CHR$(1);
60 REM *** PRINT OUR NEW SYMBOL
70 PRINT "#####"
```

Figure A-11 is the downloading paragraph symbols program (refer to Figure 8-2) amended for the Apple II computer.

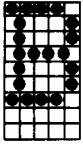
```
5 PRINT CHR$(4);"PR#1"
10 REM *** DEFINE COPYRIGHT SYMBOL IN PLACE OF A @ SYMBOL
20 PRINT CHR$(27); CHR$(42); CHR$(1); CHR$(64); CHR$(0); CHR$(28);
  CHR$(34); CHR$(73); CHR$(20); CHR$(65); CHR$(20); CHR$(65); CHR$(
  34); CHR$(20);
30 REM *** SELECT DOWNLOADABLE CHARACTERS
40 PRINT CHR$(27); CHR$(36); CHR$(1);
50 REM *** PRINT OUR NEW SYMBOL
60 PRINT "COPYRIGHT © 1983"
```

Figure A-12 is the downloading copyright program (refer to Figure 8-4) amended for the Apple II computer.



```
5 PRINT CHR$ (4);"PR#1"  
10 REM *** DEFINE MACRO TO BE EMPHASIZED AND EXPANDED TEXT  
20 PRINT CHR$ (27); CHR$ (43); CHR$ (27);"E"; CHR$ (27);"W"; CHR$ (1); CHR$  
   (30);  
30 REM *** ISSUE MACRO COMMAND AND TRY PRINTING  
40 PRINT CHR$ (27); CHR$ (33);  
50 PRINT "EMPHASIZED AND EXPANDED BY MACRO"
```

Figure A-15 is the macro-commands program (refer to Figure 8-7) amended for the Apple II computer.



ATARI 800/400

The Interface

The best way to hook up your Atari to your Gemini is with the **Universal/Atari Parallel Interface** by Star Micronics. It comes complete with its own cable. You may also use an Atari 850 interface; pin-outs to build a cable for the 850 are listed in this Appendix.

Setting the DIP Switches

Turn DIP switch number 2-3 and 2-4 "off" on your Gemini, and position switch number 2-1 as you wish.

DIP switch number 2-2 should be set "off" if you want Gemini to print every time it receives a CR code; set the switch to "on" if you wish to have the carriage return ignored, allowing for printing only when the buffer is full.

The Use of BASIC

Atari BASIC is somewhat different from the Microsoft BASIC used in this manual. If you do not have the Microsoft BASIC cartridge for your Atari, the following discussion will help you understand the modification necessary for the programs given in this manual. Then you may simply type in the amended programs listed at the back of this appendix.

Send a program listing to the Gemini using a LIST "P" command instead of LLIST (as shown in the sample programs in this manual).

All character-string variables must be dimensional if you are using Atari BASIC. The maximum length of a string is 99 characters.

If you use Atari BASIC, ending the LPRINT command with a semicolon causes Atari to insert spaces in the line until it reaches



40 characters in length. To avoid this problem, you must remember another form of the print command. Instead of

```
LPRINT CHR$(27); "5";
```

you need to open a device port, using the OPEN statement:

```
10 OPEN #7,8,0,"P"
```

Note: #7 tells the Atari to open file number 7, the #8 tells it that you are using file number 7 as an output device; 0 is not used, and the "P" assigns the device to the printer.

Once the file is assigned, you would use the following statement:

```
PRINT #7; CHR$(27);"5";
```

At the end of your program, you need to purge the printer buffer and close the file. Use the following statements:

```
PRINT #7  
CLOSE #7
```

Because Atari uses codes 0-31 for its graphics characters, you may want to add 128 to the ASCII control codes. For example, send CHR\$(141) instead of CHR\$(13) for a carriage return. As another example, send CHR\$(155) instead of CHR\$(27) for the start of an escape-function code sequence.

The Atari BASIC does not support the BASIC tab function. If you have further questions, the first step to finding the answer is to consult your Atari user's manual.



Pin Configurations

The following chart shows the pin-outs for Gemini-10X and Gemini-15X hooked up to the Atari 400/800, using the 850 module parallel interface.

Gemini-10X/15X		Parallel	Atari (850)	
1.	STROBE		1.	STROBE
2.	D1		2.	D1
3.	D2		3.	D2
4.	D3		4.	D3
5.	D4		5.	D4
6.	D5		6.	D5
7.	D6		7.	D6
8.	D7		8.	D7
16.	SG		11.	SG
32.	ERROR		12.	FAULT
11.	BUSY		13.	BUSY
9.	D8		15.	D8

Following are the program listings amended for the Atari computer. Figure B-2 is a new word processing program similar to the one presented in Chapter 5, but without the use of the single character input function, which the Atari computer does not have. The basic idea of the program is the same; you can input 22 strings of text and each string can be up to 80 characters long.

After typing your text, you can get into command level by entering the word "COM" and hitting Return. In command level, you will have a menu at the top of your screen. From this point, everything works like the word processing program in Chapter 5. You have the option of "ESC" "CHR\$" to send codes to the printer; and after that, you can print your text or return to the first level to enter more text.



Program Listings

```
5 DIM M1$(5),M2$(5),M3$(5)
10 OPEN #4,B,0,"P"
20 PRINT #4;CHR$(27);"@";CHR$(27);"R";CHR$(15);
30 FOR I=1 TO 2
40 IF I=1 THEN PRINT #4;CHR$(14);" GEMINI STANDARD TYPE STYLES";CHR$(18)
50 IF I=2 THEN PRINT #4;CHR$(14);CHR$(27);"4";" GEMINI ITALIC TYPE STYLE";CHR$(18)
60 FOR N=1 TO 40:PRINT #4;CHR$(231);CHR$(232);:NEXT N:PRINT #4:PRINT #4
70 PRINT #4;" 17 CPI 12 CPI 10 CPI 8.5 CPI 6 CPI 5 CPI"
80 FOR N=1 TO 80:PRINT #4;CHR$(241);:NEXT N:PRINT #4
90 M1$=" ABC"
92 M2$="X":M2$(LEN(M2$)+1)=CHR$(120)
94 M3$="Y":M3$(LEN(M3$)+1)=CHR$(121)
100 FOR J=1 TO 4
110 IF I=2 THEN PRINT #4;CHR$(27);"4";
120 IF J=1 THEN PRINT #4;CHR$(27);"B";CHR$(3);"NORMAL ";
130 IF J=2 THEN PRINT #4;CHR$(27);"B";CHR$(3);"DOUBLE STRIKE ";
140 IF J=3 THEN PRINT #4;CHR$(27);"B";CHR$(3);"EMPHASIZED ";
150 IF J=4 THEN PRINT #4;CHR$(27);"B";CHR$(3);"DOUBLE STRIKE ";
160 FOR K=1 TO 6
170 IF K=1 THEN PRINT #4;CHR$(27);"B";CHR$(3);
180 IF K=2 THEN PRINT #4;CHR$(27);"B";CHR$(2);
190 IF K=3 THEN PRINT #4;CHR$(27);"B";CHR$(1);
200 IF K=4 THEN PRINT #4;CHR$(27);"B";CHR$(3);CHR$(14);
210 IF K=5 THEN PRINT #4;CHR$(27);"B";CHR$(2);CHR$(14);
220 IF K=6 THEN PRINT #4;CHR$(27);"B";CHR$(1);CHR$(14);
230 IF J=2 THEN PRINT #4;CHR$(27);"G";
240 IF J=3 THEN PRINT #4;CHR$(27);"E";
250 IF J=4 THEN PRINT #4;CHR$(27);"G";CHR$(27);"E";
260 PRINT #4;M1$;CHR$(27);"S";CHR$(0);M2$;CHR$(27);"S";CHR$(1);M3$;
265 PRINT #4;CHR$(27);"T";CHR$(18);
300 IF K>3 THEN PRINT #4;CHR$(18);
310 NEXT K
320 IF J=4 THEN GOSUB 500
340 PRINT #4;CHR$(27);"@"
350 IF J=2 THEN PRINT #4;CHR$(10);
360 NEXT J
370 FOR M=1 TO 5:PRINT #4:NEXT M
380 NEXT I
400 CLOSE #4
```



```

410 END
500 PRINT #4;CHR$(27);"A";CHR$(9);CHR$(10);
510 PRINT #4;CHR$(27);"B";CHR$(3);
520 PRINT #4;"  + ";CHR$(13);"EMPHASIZED ";
530 RETURN

```

Figure B-1 is the font styles program (refer to Figure 5-1) amended for the Atari computer.

```

5 DIM A$(80),B(22,80)
6 DIM C(22),D$(10),E$(10),F$(10),S(10)
7 N=1
10 GRAPHICS 0
15 GOSUB 2000
20 INPUT A$
25 IF A$="STOP" THEN 160
26 IF A$="COM" THEN 500
30 GOSUB 1000
40 GOTO 20
100 IF N=1 THEN 160
105 FOR I=1 TO N-1
110 FOR J=1 TO C(I)
120 PRINT CHR$(B(I,J));
130 NEXT J
140 PRINT
150 NEXT I
160 END
500 REM
505 GRAPHICS 0:PRINT "ENTER YOUR COMMAND....."
507 PRINT "P)RINT Q)UIT E)SC C)HR$"
509 PRINT "-----"
510 INPUT D$
520 IF D$="E" THEN GOSUB 700
530 IF D$="C" THEN GOSUB 600
540 IF D$="Q" THEN GOTO 7
545 IF D$="P" THEN GOSUB 800
550 GOTO 500
600 REM
605 PRINT "ENTER YOUR CODE CHR$( )";
610 INPUT L

```



```
620 OPEN #4,B,0,"P"
630 PRINT #4,CHR$(L)
640 CLOSE #4
650 RETURN
700 REM
710 PRINT "ENTER YOUR ESC CODE";
720 INPUT E$
725 J=1
730 FOR I=1 TO LEN(E$)
732 X=ASC(E$(I)):IF X=32 THEN 740
735 S(J)=X:J=J+1
740 NEXT I
750 OPEN #4,B,0,"P"
751 IF J=2 THEN 760
752 AS=2
755 K=VAL(E$(AS,J-1))
760 PRINT #4;CHR$(27);CHR$(S(1));
770 PRINT #4;CHR$(K)
780 CLOSE #4
790 RETURN
800 REM
803 IF N=1 THEN 870
805 OPEN #4,B,0,"P"
810 FOR I=1 TO N-1
820 FOR J=1 TO C(I)
830 PRINT #4;CHR$(B(I,J));
840 NEXT J
850 PRINT #4
860 NEXT I
865 CLOSE #4
870 RETURN
1000 REM
1010 FOR J=1 TO LEN(A$)
1020 B(N,J)=ASC(A$(J))
1030 NEXT J
1040 C(N)=LEN(A$)
1045 N=N+1
1050 RETURN
2000 REM
2010 PRINT "YOU CAN TYPE UP TO 22 STRINGS AND "
2020 PRINT "EACH STRING CAN BE 80 CHARACTERS"
2030 PRINT "LONG. TYPE 'COM' TO GOTO COMMAND"
```




```

2040 PRINT "LEVEL...AND 'STOP' TO GET OUT"
2050 PRINT "-----"
2060 RETURN

```

Figure B-2 is the word processing program (refer to Figure 5-4) amended for the Atari computer.

```

5 DIM A$(30),B$(12)
10 OPEN #4,8,0,"P"
20 A$=""
25 B$=""
30 PRINT #4;A$;"HIGH RESOLUTION VERTICAL SPACING"
35 PRINT #4;A$;"USING THE ESC 3 N CONTROL CODE"
40 FOR I=1 TO 30
50 IF I=13 THEN 140
60 M=INT(144/(6*I)):IF M<2 THEN M=2
70 PRINT #4;CHR$(27);CHR$(51);CHR$(I);
80 FOR J=0 TO M
90 IF J=M-1 THEN PRINT #4;I;"/144 INCH SPACING";B$;:GOTO 110
100 PRINT #4;A$;
110 FOR N=1 TO 40:PRINT #4;CHR$(241);:NEXT N:PRINT #4
120 NEXT J
140 NEXT I
150 PRINT #4;CHR$(27);"@"
160 CLOSE #4
170 END

```

Figure B-3 is the vertical spacing program (refer to Figure 6-5) amended for the Atari computer.

```

10 OPEN #4,8,0,"P"
20 PRINT #4;CHR$(27);"@";CHR$(27);"A";CHR$(6)
30 FOR LINE=1 TO 2
40 PRINT #4;CHR$(27);"K";CHR$(114);CHR$(0);
50 FOR SPACE=1 TO 100
55 PRINT #4;CHR$(0);:NEXT SPACE
60 FOR COLM=1 TO 14
70 READ C:PRINT #4;CHR$(C);:NEXT COLM
80 PRINT #4;NEXT LINE

```



```
90 DATA 14,16,34,64,140,140,128,128,140,140,64,34,16,14
100 DATA 224,16,232,68,34,18,18,18,18,34,68,232,16,224
110 PRINT #4;CHR$(27);"@"
120 FOR I=1 TO 5:PRINT #4:NEXT I
130 CLOSE #4
140 END
```

Figure B-4 is the dot graphics program presented in Chapter 6 (the one that puts a smile on your paper) amended for the Atari computer.

```
10 OPEN #4,B,0,"P"
20 PRINT #4;CHR$(163);CHR$(161);
30 PRINT #4;CHR$(27);"A";CHR$(6)
40 PRINT #4;CHR$(162);CHR$(160);
50 PRINT #4;CHR$(27);"2"
60 CLOSE #4:END
```

Figure B-5 is the circle program presented in Chapter 7, amended for the Atari computer.

```
10 OPEN #4,B,0,"P"
20 PRINT #4;" ";CHR$(163);
30 PRINT #4;CHR$(27);"A";CHR$(6)
40 PRINT #4;CHR$(160)
50 CLOSE #4:END
```

Figure B-6 is the tilde program presented in Chapter 7 amended for the Atari computer.

```
10 OPEN #4,B,0,"P"
20 PRINT #4," N";CHR$(13);
30 PRINT #4," ";CHR$(164);CHR$(13);
40 PRINT #4,"W";CHR$(166);CHR$(32);CHR$(167);"E";CHR$(13);
50 PRINT #4," ";CHR$(165);CHR$(13);
60 PRINT #4," S"
70 CLOSE #4
80 END
```

Figure B-7 is the map program presented in Chapter 7 amended for the Atari computer.



```
10 OPEN #4,8,0,"P"
15 PRINT #4;CHR$(27);"U";CHR$(1)
20 PRINT #4;CHR$(27);"A";CHR$(6)
30 FOR J=1 TO 9
40 FOR I=1 TO 12
50 READ A
60 PRINT #4;CHR$(A);
70 NEXT I
80 PRINT #4
90 NEXT J
100 PRINT #4;CHR$(27);"e"
110 CLOSE #4
120 END
130 DATA 239,224,224,224,239,224,224,224,224,224,224,224
140 DATA 239,224,224,224,239,224,224,224,224,224,224,224
150 DATA 239,224,224,224,239,224,239,224,224,224,239,224
160 DATA 239,224,239,224,239,224,234,233,224,234,233,224
170 DATA 239,234,239,233,239,224,224,239,224,239,224,224
180 DATA 239,239,224,239,239,224,224,234,239,233,224,224
190 DATA 234,233,224,234,233,224,224,224,239,224,224,224
200 DATA 224,224,224,224,224,224,224,234,233,224,224,224
210 DATA 224,224,224,224,224,224,224,239,224,224,224,224
```

Figure B-8 is the "Wy" program (refer to Figure 7-1) amended for the Atari computer.

```
10 OPEN #4,8,0,"P"
20 PRINT #4;CHR$(27);"A";CHR$(6);CHR$(15)
25 PRINT #4;CHR$(27);"U";CHR$(1)
30 FOR K=1 TO 3
40 PRINT #4;CHR$(27);"B";CHR$(K)
50 RESTORE
60 FOR J=1 TO 9
70 FOR I=1 TO 12
80 READ A:PRINT #4;CHR$(A);
90 NEXT I
95 PRINT #4
100 NEXT J
110 PRINT #4
120 NEXT K
130 PRINT #4;CHR$(27);"e"
```



```
140 CLOSE #4
150 END
200 DATA 224,224,224,224,224,224,224,224,224,245,224,224
210 DATA 224,224,224,224,224,224,224,224,252,245,224,224
220 DATA 224,224,224,245,224,224,224,252,239,245,224,224
230 DATA 224,224,252,245,224,224,252,239,239,245,224,224
240 DATA 224,252,239,245,224,252,239,239,239,245,224,224
250 DATA 252,239,239,245,252,239,239,239,239,245,224,224
260 DATA 241,241,241,249,241,241,241,241,241,249,224,224
270 DATA 239,239,239,239,239,239,239,239,239,239,251
280 DATA 253,239,239,239,239,239,239,239,239,239,251,224
```

Figure B-9 is the boat program (refer to Figure 7-2) amended for the Atari computer.

```
2 GRAPHICS 0
3 PRINT " ":PRINT " ":PRINT " "
4 PRINT " THIS PROGRAM TAKES ABOUT"
5 PRINT " 4 MINUTES TO RUN, PLEASE"
6 PRINT " TURN ON YOUR PRINTER AND "
7 PRINT " STAND BY ....."
8 PRINT " ":PRINT " ":PRINT " "
10 FOR I=1536 TO 1553
20 READ 0
30 POKE I,0
40 NEXT I
50 DATA 104,104,133,205,104,133,204
60 DATA 104,5,205,133,213,104,5,204
70 DATA 133,212,96
100 REM MICRO-PLOT
110 DIM BIT(76,12),MASK(6)
120 DIM A$(76)
130 DIM B$(10)
132 FOR I=0 TO 11
133 FOR J=1 TO 76
134 BIT(J,I)=0
135 NEXT J
136 NEXT I
1000 REM SET PROGRAM CONSTANT
1010 MASK(1)=64:MASK(4)=8
1020 MASK(2)=32:MASK(5)=4
```



```
1030 MASK(3)=16:MASK(6)=2
1040 LX=20:LY=20
1050 XFAC=72/LX:YFAC=75/LY
2000 REM PLOT CURVE
2010 LET RAD=9
2020 X1=19:Y1=10
2030 FOR ANG=0 TO 360 STEP 10
2040 R1=ANG*6.28/360
2050 X2=RAD*COS(R1)+10
2055 Y2=RAD*SIN(R1)+10
2060 GOSUB 4000
2070 NEXT ANG
2080 FOR ANG=0 TO 360 STEP 15
2090 R1=ANG*6.28/360
2100 R2=(ANG+150)*6.28/360
2110 X1=RAD*COS(R1)+10
2115 Y1=RAD*SIN(R1)+10
2120 X2=RAD*COS(R2)+10
2125 Y2=RAD*SIN(R2)+10
2130 GOSUB 4000
2140 NEXT ANG
3000 REM SEND BIT IMAGE TO PRINTER
3005 OPEN #4,8,0,"P"
3010 PRINT #4;CHR$(27);"A";CHR$(6)
3020 FOR ROW=0 TO 11
3040 PRINT #4;CHR$(27);"K";CHR$(119);CHR$(1);
3050 FOR COL=1 TO 75
3052 FOR LL=1 TO 5
3054 RE=INT(BIT(COL,ROW))
3060 PRINT #4;CHR$(RE);
3065 NEXT LL
3070 NEXT COL
3080 PRINT #4
3090 NEXT ROW
3100 PRINT #4;CHR$(27);"A";CHR$(12)
3105 PRINT #4," ":PRINT #4," "
3110 CLOSE #4
3150 END
4000 REM DRAW A LINE FROM X1,Y1 TO X2,Y2
4010 XL=X2-X1:YL=Y2-Y1
4020 NX=ABS(XL*XFAC):NY=ABS(YL*YFAC)
```



```
4030 IF NX<NY THEN NX=NY
4040 NS=INT(NX+1)
4050 DX=XL/NS
4055 DY=YL/NS
4060 FOR I=1 TO NS
4070 X1=X1+DX:Y1=Y1+DY
4080 GOSUB 5000
4090 NEXT I
4100 RETURN
5000 REM PLOT A POINT AT X1,Y1
5010 XX=X1*XFAC:YY=Y1*YFAC
5020 COL=INT(XX)+1
5030 ROW=INT(YY/6)
5040 XIT=INT(YY-ROW*6)+1
5050 A1=BIT(COL,ROW)
5060 A2=MASK(XIT)
5070 BIT(COL,ROW)=USR(1536,A1,A2)
5080 RETURN
```

Figure B-10 is the graphics program (refer to Figure 7-4) amended for the Atari computer.

```
10 DIM E$(5)
20 REM DEFINE ESCAPE CHARACTERS
25 E$=CHR$(27)
30 OPEN #4,8,0,"P"
35 REM DEFINE PARAGRAPH SYMBOL IN PLACE @
40 PRINT #4;E$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(6);
45 PRINT #4;CHR$(9);CHR$(0);CHR$(9);CHR$(118);CHR$(9);CHR$(0);
47 PRINT #4;CHR$(127);CHR$(0)
50 REM SELECT DOWNLOADABLE CHARACTERS
60 PRINT #4;E$;CHR$(36);CHR$(1)
70 REM PRINT OUR NEW SYMBOL
80 PRINT #4;"#####"
90 CLOSE #4:END
```

Figure B-11 is the downloading paragraph symbols program (refer to Figure 8-2) amended for the Atari computer.



```

10 DIM E$(5)
20 REM DEFINE ESCAPE CHARACTERS
25 E$=CHR$(27)
30 OPEN #4,8,0,"P"
35 REM DEFINE COPYRIGHT SYMBOL IN PLACE @
40 PRINT #4;E$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);
45 PRINT #4;CHR$(34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);
47 PRINT #4;CHR$(65);CHR$(34);CHR$(20)
50 REM SELECT DOWNLOADABLE CHARACTERS
60 PRINT #4;E$;CHR$(36);CHR$(1)
70 REM PRINT OUR NEW SYMBOL
80 PRINT #4;"COPY RIGHT @ 1983"
90 CLOSE #4:END

```

Figure B-12 is the downloading copyright program (refer to Figure 8-4) amended for the Atari computer.

```

10 DIM E$(5)
15 OPEN #4,8,0,"P"
20 REM DEFINE ESCAPE CHARACTERS
25 E$=CHR$(27)
27 REM COPY CHARACTERS TO RAM
28 PRINT #4;E$;CHR$(42);CHR$(0)
35 REM DEFINE COPYRIGHT SYMBOL IN PLACE @
40 PRINT #4;E$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);
45 PRINT #4;CHR$(34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);
47 PRINT #4;CHR$(65);CHR$(34);CHR$(20)
50 REM SELECT DOWNLOADABLE CHARACTERS
60 PRINT #4;E$;CHR$(36);CHR$(1)
70 REM PRINT OUR NEW SYMBOL
80 PRINT #4;"COPY RIGHT @ 1983"
90 CLOSE #4:END

```

Figure B-13 is the downloading program presented in Figure 8-5 amended for the Atari computer.

```

10 DIM E$(5)
15 OPEN #4,8,0,"P"
20 REM DEFINE ESCAPE CHARACTERS
25 E$=CHR$(27)
27 REM COPY CHARACTERS TO RAM
28 PRINT #4;E$;CHR$(42);CHR$(0)
35 REM DEFINE COPYRIGHT SYMBOL IN PLACE @
40 PRINT #4;E$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);
45 PRINT #4;CHR$(34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);
47 PRINT #4;CHR$(65);CHR$(34);CHR$(20)
49 REM DEFINE COPYRIGHT SYMBOL IN PLACE &
50 PRINT #4;E$;CHR$(42);CHR$(1);CHR$(38);CHR$(1);CHR$(28);
55 PRINT #4;CHR$(34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);
57 PRINT #4;CHR$(65);CHR$(34);CHR$(20)
59 REM SELECT DOWNLOADABLE CHARACTERS
60 PRINT #4;E$;CHR$(36);CHR$(1)
70 REM PRINT OUR NEW SYMBOL
80 PRINT #4;"@%&@%&@%&@%&@%&@%&@%&"
90 CLOSE #4:END

```

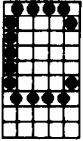
Figure B-14 is the downloading program presented in Figure 8-6 amended for the Atari computer.

```

5 DIM E$(5)
10 REM DEFINE ESC CHARACTER
20 E$=CHR$(27)
30 OPEN #4,8,0,"P"
40 PRINT #4;E$;CHR$(43);E$;"E";E$;"W";CHR$(1);CHR$(30)
50 REM ISSUE MACRO COMMAND
60 PRINT #4;E$;CHR$(33)
70 PRINT #4;"EMPHASIZED AND EXPANDED BY MACRO"
80 CLOSE #4:END

```

Figure B-15 is the macro-commands program (refer to Figure 8-7) amended for the Atari computer.



COMMODORE VIC-20 & C-64

The Connecting Cable

Star Micronics, Inc. has created a special Centronics parallel interface especially for your VIC-20 and C-64 computers. It is called the **Universal/Commodore Parallel Interface**.

If you haven't already done so, contact your Gemini dealer about obtaining and installing the interface device and cable.

Setting the DIP Switches

The only DIP switch that must be set in one position or the other is switch number 2-3, which must be "off" on your Gemini. Switch number 2-2 should be set to "off" if you want Gemini to print every time it receives a CR code; set the switch "on" if you wish to have the carriage return ignored and printing only when the buffer is full.

The Use of BASIC

We recommend the use of Commodore's Advanced BASIC Program. If you encounter difficulty in running the programs presented in this User's Manual, consult the documentation that supports the Advanced BASIC Program for Commodore.



Program Listings

```
20 E$=CHR$(27):LF$=CHR$(10):FF$=CHR$(12):CR$=CHR$(13)
30 PC$=E$+"B"+CHR$(1):EL$=E$+"B"+CHR$(2):CM$=E$+"B"+CHR$(3)
40 SP$=E$+"S"+CHR$(0):SB$=E$+"S"+CHR$(1):CS$=E$+"T"
50 IT$=E$+CHR$(52)
60 WD$=CHR$(14):CW$=CHR$(20) :PS$=E$+"@"
70 EM$=E$+"E":CE$=E$+"F":DB$=E$+"6":CD$=E$+"H"
80 OPEN#4,4,5:PRINT#4,PS$,E$,"R";CHR$(15);
90 FOR I=1 TO 2
100 IF I=1 THEN PRINT#4,WD$;" GEMINI STANDARD TYPE STYLES";CM$;LF$;
110 IF I=2 THEN PRINT#4,IT$,WD$;" GEMINI ITALIC TYPE STYLE";
115 PRINT#4,CM$;LF$
120 FOR N=1 TO 40:PRINT#4,CHR$(231);CHR$(232);:NEXT N:PRINT#4:PRINT#4
130 PRINT#4," 17 CPI 12 CPI 10 CPI 8.5 CPI 6 CPI 5 CPI"
140 FOR N=1 TO 90:PRINT#4,CHR$(241);:NEXT N:PRINT#4
150 MSG$(1)=" ABC":MSG$(2)="X"+CHR$(120):MSG$(3)="Y"+CHR$(121)
160 FOR J=1 TO 4
170 IF I=2 THEN PRINT#4,IT$;
180 IF J=1 THEN PRINT#4,CM$;"NORMAL ";
190 IF J=2 THEN PRINT#4,CM$;"DOUBLE STRIKE ";
200 IF J=3 THEN PRINT#4,CM$;"EMPHASIZED ";
210 IF J=4 THEN PRINT#4,CM$;"DOUBLE STRIKE ";
220 FOR K=1 TO 6
230 IF K=1 THEN PRINT#4,CM$;
240 IF K=2 THEN PRINT#4,EL$;
250 IF K=3 THEN PRINT#4,FC$;
260 IF K=4 THEN PRINT#4,CM$,WD$;
270 IF K=5 THEN PRINT#4,EL$,WD$;
280 IF K=6 THEN PRINT#4,PC$,WD$;
290 IF J=2 THEN PRINT#4,DB$;
300 IF J=3 THEN PRINT#4,EM$;
310 IF J=4 THEN PRINT#4,DB$,EM$;
320 PRINT#4,MSG$(1);SP$;MSG$(2);CS$;SB$;MSG$(3);CS$;CD$;
330 IF K>3 THEN PRINT#4,CM$;
340 NEXT K
350 IF J=4 THEN PRINT#4,E$;"A";CHR$(9);LF$;CM$;" + ";CR$;"EMPHASIZED ";
360 PRINT#4,PS$
365 IF J=2 THEN PRINT#4,LF$;
```



```

370 NEXT J
380 FOR M=1 TO 5:PRINT#4,LF$:NEXT M
390 NEXT I
400 END

```

Figure C-1 is the font styles program (refer to Figure 5-1) amended for the Commodore computer.

```

10 DIM A$(22,40)
12 C%=CHR$(91)
20 PRINT CHR$(147):X=0:Y=0:Z=1024
30 GOSUB 150:POKE Z+X*40+Y,ASC(C%)
40 GET B$:IF B%=""THEN 40
50 IF B%=CHR$(5)THEN PRINT CHR$(147):END
60 IF B%=CHR$(18) THEN GOSUB 200:GOTO 20
70 IF B%=CHR$(16) THEN GOTO 220
80 IF B%=CHR$(13) THEN GOSUB 700:GOTO 40
90 IF B%=CHR$(4)THEN GOSUB 130:GOTO 40
100 B=ASC(B%):IF B>64 THEN B=B-64
102 POKE Z+X*40+Y,B:A$(X,Y)=B%:Y=Y+1
104 IF Y=40 THEN A$(X,Y)=CHR$(13):X=X+1:Y=0:GOTO 40
120 POKE Z+X*40+Y,ASC(C%):GOTO 40
130 POKE Z+X*40+Y,ASC(" "):Y=Y-1:IF Y<0 THEN Y=0
135 A$(X,Y)=""
140 POKE Z+X*40+Y,ASC(C%):RETURN
150 M%="CTRL P)RINT D)ELETE R)ESTART E)XIT"
160 F=1984:GOSUB 800
170 RETURN
200 PRINT CHR$(147):M%="PLEASE STAND BY"
205 F=1555:GOSUB 800
210 FOR I=0 TO X:FOR J=1 TO 40:A$(I,J)="" :NEXT J:NEXT I
215 RETURN
220 OPEN#4,4,5:PRINT#4,CHR$(27);"@":CLOSE#4
230 PRINT CHR$(147):M%="COMMAND P)RINT R)ETURN":F=1984:GOSUB 800

```



```
250 GET G$:IF G$=""THEN 250
260 IF G$="F" THEN GOSUB 310
270 IF G$="C" THEN GOSUB 320:GOTO 230
280 IF G$="R" THEN GOSUB 300:GOTO 30
290 GOTO 250
300 PRINT CHR$(147):GOSUB 900:RETURN
310 OPEN4,4,5:CMD4
312 FOR I=0 TO X:FOR J=0 TO 40
314 PRINT A$(I,J):NEXT J
316 NEXT I:PRINT#4:CLOSE4:RETURN
320 M$=" " " :F=1984:GOSUB 800
330 M$="E) ESC C) CHR$ Q) END COMMAND":F=1984:GOSUB 800
340 ST$=""
360 GET G$:IF G$=""THEN 360
370 IF G$="E" THEN GOSUB 420:GOTO 400
380 IF G$="C" THEN GOSUB 530:GOTO 400
390 IF G$="Q" THEN GOTO 410
400 GOTO 360
410 RETURN
420 L=0:M$=" ESC " :F=1910:GOSUB 800
425 F=1915
430 GET G$:IF G$=""THEN 430
440 IF G$=CHR$(13) THEN 480
450 M$=G$+" " :F=F+2:GOSUB 900
470 S$(L)=G$:L=L+1:GOTO 430
480 IF L>2 THEN GOSUB 500:
490 IF L=2 THEN GOSUB 650
500 IF L=1 THEN ST$=CHR$(27)+S$(0)
510 OPEN4,4,5:PRINT#4,ST$:CLOSE4:ST$=""
515 FOR J=1 TO L:ST$(J)="" :NEXT J:L=0
520 M$=" " " :F=1910:GOSUB 800:RETURN
530 M$=" CHR$ " :F=1910:GOSUB 800:ST$=""
535 F=1916
540 GET G$:IF G$=""THEN 540
550 IF G$=CHR$(13) THEN 570
560 M$=G$:F=F+1:GOSUB 800:ST$=ST$+G$:GOTO 540
570 M$=" " " :F=1910:GOSUB 800:
580 OPEN4,4,5:PRINT#4,CHR$(VAL(ST$)):CLOSE4:ST$="" :RETURN
600 REM
610 FOR J=1 TO L:D$=D$+S$(J):NEXT J
620 ST$=CHR$(27)+S$(0)+CHR$(VAL(D$))
```



```
630 D$=""
640 RETURN
650 ST$=CHR$(27)+S$(0)+CHR$(VAL(S$(1)))
660 RETURN
700 REM
710 A$(X,Y)=B$:
720 POKE Z+X*40+Y,ASC(" ")
730 X=X+1:Y=0
740 POKE Z+X*40+Y,ASC(C$)
750 RETURN
800 FOR I=F TO F-1+LEN(M$)
810 J=I-(F-1)
820 M=ASC(MID$(M$,J,1))
830 IF M>64 THEN M=M-64
840 POKE I,M
850 NEXT I
860 RETURN
900 REM
910 FOR I=0 TO X:FOR J=0 TO 39
915 IF A$(I,J)=""THEN 950
920 A=ASC(A$(I,J))
925 IF A=13 THEN W=W+40:GOTO 950
930 IF A>64 THEN A=A-64
935 W=Z+I*40+J
940 POKE W,A
950 NEXT J
960 NEXT I
970 RETURN
```

Figure C-2 is the word processing program (refer to Figure 5-4) amended for the Commodore computer. NOTE: The only way to move the cursor around on the Commodore computer is to use the screen memory map. We can specify the location on the screen and then "POKE" the character into memory. Because of this limitation, we had to use "POKE" instead of "PRINT" statement for the Commodore computer. However, when we "POKE" characters into screen memory, the color will be light so you must make sure that the background color on your screen is dark.



```
10 OPEN4,4,5:CMD4
20 PRINT TAB(30);"HIGH RESOLUTION VERTICAL SPACING"
30 PRINT TAB(30);"USING THE ESC 3 M CONTROL CODE"
40 FOR I=1 TO 30
50 IF I=13 THEN GOTO 140
60 M=INT(144/(6*I)):IF M<2 THEN M=2
70 PRINT CHR$(27);CHR$(51);CHR$(1);
90 FOR J=0 TO M
90 IF J=M-1 THEN PRINT I;"/144 INCH SPACING";TAB(10);:GOTO 110
100 PRINT TAB(30);
110 FOR N=1 TO 40:PRINT CHR$(241);:NEXT N:PRINT
120 NEXT J
140 NEXT I
150 PRINT CHR$(27);"@"
160 PRINT#4:CLOSE4
170 END
```

Figure C-3 is the vertical spacing program (refer to Figure 6-5) amended for the Commodore computer.

```
10 OPEN4,4,5:CMD4
20 PRINT CHR$(27);"@";CHR$(27);"A";CHR$(6)
30 FOR LINE=1 TO 2
40 PRINT CHR$(27);"K";CHR$(114);CHR$(0);
50 FOR SPACE=1 TO 100
55 PRINT CHR$(0);:NEXT SPACE
60 FOR COLM=1 TO 14
70 READ C:PRINT CHR$(C);:NEXT COLM
90 PRINT :NEXT LINE
90 DATA 14,15,34,64,140,140,128,129,140,140,64,34,15,14
100 DATA 224,16,232,68,34,18,18,18,18,34,68,232,16,224
110 PRINT CHR$(27);"@";:PRINT:PRINT:PRINT
120 PRINT#4:CLOSE4
130 END
```

Figure C-4 is the dot graphics program presented in Chapter 6 (the one that puts a smile on your paper) amended for the Commodore computer.



```
10 OPEN4,4,5:CMD4
20 PRINT CHR$(163);CHR$(161);
30 PRINT CHR$(27);"A";CHR$(6)
40 PRINT CHR$(162);CHR$(160);
50 PRINT CHR$(27);"2"
60 PRINT#4:CLOSE4:END
```

Figure C-5 is the circle program presented in Chapter 7, amended for the Commodore computer.

```
10 OPEN4,4,5:CMD4
20 PRINT TAB(1);CHR$(163);CHR$(27);"A";CHR$(6)
30 PRINT CHR$(160)
40 PRINT#4:CLOSE4
50 END
```

Figure C-6 is the tilde program presented in Chapter 7 amended for the Commodore computer.

```
10 OPEN4,4,5:CMD4
20 PRINT TAB(20);"N";CHR$(13);
30 PRINT TAB(20);CHR$(164);CHR$(13);
40 PRINT TAB(18);"W";CHR$(166);CHR$(32);CHR$(167);"E";CHR$(13);
50 PRINT TAB(20);CHR$(165);CHR$(13);
60 PRINT TAB(20);"S"
70 PRINT#4:CLOSE4
```

Figure C-7 is the map program presented in Chapter 7 amended for the Commodore computer.



```
10 OPEN4,4,5:CMD4
15 PRINT CHR$(27);"U";CHR$(1)
20 PRINT CHR$(27);"A";CHR$(6)
30 FOR J=1 TO 9
40 FOR I=1 TO 12
50 READ A
60 PRINT CHR$(A);
70 NEXT I
80 PRINT
90 NEXT J
100 PRINT CHR$(27);"@":
110 PRINT#4:CLOSE4
120 END
130 DATA 239,224,224,224,239,224,224,224,224,224,224,224
140 DATA 239,224,224,224,239,224,224,224,224,224,224,224
150 DATA 239,224,224,224,239,224,239,224,224,224,239,224
160 DATA 239,224,239,224,239,224,234,233,224,234,233,224
170 DATA 239,234,239,233,239,224,224,239,224,239,224,224
180 DATA 239,239,224,239,239,224,224,234,239,233,224,224
190 DATA 234,233,224,234,233,224,224,224,239,224,224,224
200 DATA 224,224,224,224,224,224,224,234,233,224,224,224
210 DATA 224,224,224,224,224,224,224,239,224,224,224,224
```

Figure C-8 is the "Wy" program (refer to Figure 7-1) amended for the Commodore computer.

```
10 OPEN4,4,5:CMD4
20 PRINT CHR$(27);"A";CHR$(6);CHR$(15)
25 PRINT CHR$(27);"U";CHR$(1)
30 FOR K=1 TO 3
40 PRINT CHR$(27);"B";CHR$(K)
50 RESTORE
60 FOR J=1 TO 9
70 FOR I=1 TO 12
80 READ A:PRINT CHR$(A);
90 NEXT I
95 PRINT
100 NEXT J
110 PRINT
```




```

120 NEXT K
130 PRINT CHR$(27);"@"
140 PRINT#4:CLOSE4
150 END
200 DATA 224,224,224,224,224,224,224,224,224,245,224,224
210 DATA 224,224,224,224,224,224,224,224,252,245,224,224
220 DATA 224,224,224,245,224,224,224,252,239,245,224,224
230 DATA 224,224,252,245,224,224,252,239,239,245,224,224
240 DATA 224,252,239,245,224,252,239,239,239,245,224,224
250 DATA 252,239,239,245,252,239,239,239,239,245,224,224
260 DATA 241,241,241,249,241,241,241,241,241,249,224,224
270 DATA 239,239,239,239,239,239,239,239,239,239,251
280 DATA 253,239,239,239,239,239,239,239,239,251,224

```

Figure C-9 is the boat program (refer to Figure 7-2) amended for the Commodore computer.

```

50 PRINT CHR$(147)
60 FOR I=1 TO 5:PRINT "I":NEXT I
70 PRINT "THIS PROGRAM TAKES ABOUT TWO MINUTES"
80 PRINT "TO RUN, SO PLEASE TURN ON YOUR PRINTER"
90 PRINT "AND STAND BY....."
100 REM MICRO-PLOT
110 DIM BITX(76,12)
1000 REM SET PROGRAM CONSTANTS
1010 MASKX(1) = 64 : MASKX(4) = 8
1020 MASKX(2) = 32 : MASKX(5) = 4
1030 MASKX(3) = 16 : MASKX(6) = 2
1040 LX = 20 : LY = 20
1050 XFAC = 72/LX : YFAC = 75/LY
2000 REM PLOT CURVE
2010 RAD = 9
2020 X1 = 19 : Y1 = 10
2030 FOR ANG = 0 TO 360 STEP 10
2040 R1 = ANG*6.28/360
2050 X2 = RAD*COS(R1)+10 : Y2 = RAD*SIN(R1)+10
2060 GOSUB 4000
2070 NEXT ANG

```



```
2080 FOR ANG = 0 TO 360 STEP 15
2090 R1 = ANG*6.28/360
2100 R2 = (ANG+150)*6.28/360
2110 X1 = RAD*COS(R1)+10 : Y1 = RAD*SIN(R1)+10
2120 X2 = RAD*COS(R2)+10 : Y2 = RAD*SIN(R2)+10
2130 GOSUB 4000
2140 NEXT ANG
3000 REM SEND BIT IMAGE MAP TO PRINTER
3005 OPEN 4,4,5
3010 PRINT#4, CHR$(27) "A" CHR$(6)
3020 FOR ROW = 0 TO 11
3025 PRINT#4,CHR$(27);"K";CHR$(119);CHR$(1);
3050 FOR COL = 1 TO 75
3055 FOR LL=1 TO 5
3060 PRINT#4,CHR$(BITZ(COL,ROW));
3065 NEXT LL
3070 NEXT COL
3080 PRINT#4,""
3090 NEXT ROW
3100 PRINT#4 : CLOSE4
3110 END
4000 REM DRAW A LINE FROM X1,Y1 TO X2,Y2
4010 XL = X2 - X1 : YL = Y2 - Y1
4020 NX = ABS(XL*XFAC) : NY = ABS(YL*YFAC)
4030 IF NX < NY THEN NX = NY
4040 NSZ = INT(NX+1)
4050 DX = XL/NSZ : DY = YL/NSZ
4060 FOR I = 1 TO NSZ
4070 X1 = X1 + DX : Y1 = Y1 + DY
4080 GOSUB 5000
4090 NEXT I
4100 RETURN
5000 REM PLOT A POINT AT X1,Y1
5010 XX = X1 * XFAC : YY = Y1 * YFAC
5020 COLZ = INT(XX)+1
5030 ROWZ = INT(YY/6)
5040 XITZ = INT(YY - ROWZ * 6) + 1
5050 BITZ(COLZ,ROWZ) = BITZ(COLZ,ROWZ) OR MASKZ(XITZ)
5060 RETURN
```

Figure C-10 is the graphics program (refer to Figure 7-4) amended for the Commodore computer.



```
10 REM DEFINE ESCAPE CHARACTER
20 ESC$=CHR$(27)
30 REM DEFINE PARAGRAPH SYMBOL IN PLACE @
35 OPEN4,4,5:CMD4
40 PRINT ESC$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(6);
45 PRINT CHR$(9);CHR$(0);CHR$(9);CHR$(118);CHR$(9);CHR$(0);
47 PRINT CHR$(127);CHR$(0)
50 REM SELECT DOWNLOADABLE CHARACTERS
60 PRINT ESC$;CHR$(36);CHR$(1)
70 REM PRINT OUR NEW SYMBOL
80 PRINT "#####"
90 PRINT#4:CLOSE4:END
```

Figure C-11 is the downloading paragraph symbols program (refer to Figure 8-2) amended for the Commodore computer.

```
10 REM DEFINE ESCAPE CHARACTER
20 ESC$=CHR$(27)
30 REM DEFINE COPYRIGHT SYMBOL IN PLACE @
35 OPEN4,4,5:CMD4
40 PRINT ESC$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);
45 PRINT CHR$(34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);
47 PRINT CHR$(65);CHR$(34);CHR$(20)
50 REM SELECT DOWNLOADABLE CHARACTERS
60 PRINT ESC$;CHR$(36);CHR$(1)
70 REM PRINT OUR NEW SYMBOL
80 PRINT "COPY RIGHT @ 1983"
90 PRINT#4:CLOSE4:END
```

Figure C-12 is the downloading copyright program (refer to Figure 8-4) amended for the Commodore computer.



```
10 REM DEFINE ESCAPE CHARACTER
20 ESC$=CHR$(27)
22 OPEN4,4,5:CMD4
24 REM COPY CHARACTERS TO RAM
25 PRINT ESC$;CHR$(42);CHR$(0)
30 REM DEFINE COPYRIGHT SYMBOL IN PLACE @
40 PRINT ESC$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);
45 PRINT CHR$(34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);
47 PRINT CHR$(65);CHR$(34);CHR$(20)
50 REM SELECT DOWNLOADABLE CHARACTERS
60 PRINT ESC$;CHR$(36);CHR$(1)
70 REM PRINT OUR NEW SYMBOL
80 PRINT "COPY RIGHT @ 1983"
90 PRINT#4:CLOSE4:END
```

Figure C-13 is the downloading program presented in Figure 8-5 amended for the Commodore computer.

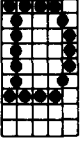
```
10 REM DEFINE ESCAPE CHARACTER
20 ESC$=CHR$(27)
22 OPEN4,4,5:CMD4
24 REM COPY CHARACTERS TO RAM
25 PRINT ESC$;CHR$(42);CHR$(0)
30 REM DEFINE COPYRIGHT SYMBOL IN PLACE @
40 PRINT ESC$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);
45 PRINT CHR$(34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);
47 PRINT CHR$(65);CHR$(34);CHR$(20)
49 REM DEFINE COPYRIGHT SYMBOL IN PLACE OF &
50 PRINT ESC$;CHR$(42);CHR$(1);CHR$(38);CHR$(1);CHR$(28);
55 PRINT CHR$(34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);
57 PRINT CHR$(65);CHR$(34);CHR$(20)
59 REM SELECT DOWNLOADABLE CHARACTERS
60 PRINT ESC$;CHR$(36);CHR$(1)
70 REM PRINT OUR NEW SYMBOL
80 PRINT "@&@&@&@&@&@&@&"
90 PRINT#4:CLOSE4:END
```

Figure C-14 is the downloading program presented in Figure 8-6 amended for the Commodore computer.



```
10 REM DEFINE ESCAPE CHARACTER
20 ESC$=CHR$(27)
30 REM CONNECT TO PRINTER
40 OPEN#4,4,5:CMD#4
50 PRINT ESC$;CHR$(43);ESC$;"E";ESC$;"W";CHR$(1);CHR$(30)
60 REM ISSUE MACRO COMMAND AND TRY PRINTING
65 PRINT ESC$;CHR$(33)
70 PRINT "EMPHASIZED AND EXPANDED BY MACRO"
80 PRINT#4:CLOSE#4
90 END
```

Figure C-15 is the macro-commands program (refer to Figure 8-7) amended for the Commodore computer.



IBM PERSONAL COMPUTER

The Interface Board

Install the IBM parallel interface board according to the directions that accompany the product, which you can obtain from your Gemini dealer or from an IBM salesperson.

Use the Gemini cable provided by your dealer or use cable information at the end of this appendix.

Setting the DIP Switches

DIP switches 2-3 and 2-4 should be turned "off" on your Gemini, and the other two may be set according to your operational requirements.

Switch 2-2 should be set "off" if you want Gemini to print every time it receives a CR code; set it "on" if you wish to have the carriage return ignored and printing only when the buffer is full.

The Use of BASIC

All the example programs provided in this User's Manual are written in BASIC for the IBM Personal Computer. You may run them in IBM Microsoft BASIC or in IBM Microsoft Advanced BASIC (BASICA).

Pin Configurations

The following chart shows the pin-outs for the Gemini-10X and Gemini-15X, interfaced with the IBM Personal Computer, using a parallel interface board.



Parallel

Gemini-10X/15X		IBM 25-Pin	
1.	<u>STROBE</u>	1.	<u>STROBE</u>
2.	D1	2.	D0
3.	D2	3.	D1
4.	D3	4.	D2
5.	D4	5.	D3
6.	D5	6.	D4
7.	D6	7.	D5
8.	D7	8.	D6
9.	D8	9.	D7
10.	<u>ACK</u>	10.	<u>ACK</u>
11.	BUSY	11.	BUSY
		12.	PAGE END
		13.	SELECT
32.	<u>ERROR</u>	14.	AUTO END
		15.	<u>ERROR</u>
		16.	INITIATE
		17.	SELECTION
16.	SG	18-25.	SG



OSBORNE

The Connecting Cable

Use the cable available from your Gemini dealer, or you can make your own cable according to the pin configuration chart below.

Setting the DIP Switches

Turn switches 2-3 and 2-4 to the "off" position on your Gemini, and set switch 2-1 as you like. Switch 2-2 should be set "off" if you want Gemini to print every time it receives a CR code; set it "on" if you wish to have the carriage return ignored and printing only when the buffer is full.

The Use of BASIC

All the sample programs provided in this user's manual will run "as is" using Microsoft BASIC. If you are a first-time user, you should start with the CPM software and reset the "printer" configuration from "serial" to "Centronics." Then load MBASIC and go for it.

When you want to change the number of columns in printing, substitute

WIDTH LPRINT 132 (*or any acceptable number*)
instead of
WIDTH "LPT #1";132 (*as used by IBM-PC*).

Pin Configurations

The following chart shows the pin-outs for Gemini printers and Osborne, using a parallel interface.



Parallel

Gemini-10X/15X

- 2. D1
- 6. D5
- 3. D2
- 7. D6
- 4. D3
- 8. D7
- 5. D4
- 9. D8
- 1. STROBE
- 11. BUSY
- 16. SG

Osborne 1

- 1. D0
- 2. D4
- 3. D1
- 4. D5
- 5. D2
- 6. D6
- 7. D3
- 8. D7
- 11. STROBE
- 15. BUSY
- 16. SG



Program Listings

```
10 WIDTH LPRINT 255
20 E$=CHR$(27);LF$=CHR$(10);FF$=CHR$(12);CR$=CHR$(13)
30 PC$=E$+"B"+CHR$(1);EL$=E$+"B"+CHR$(2);CM$=E$+"B"+CHR$(3)
40 SP$=E$+"S"+CHR$(0);SB$=E$+"S"+CHR$(1);CS$=E$+"T"
50 IT$=E$+CHR$(52)
60 WD$=CHR$(14);CW$=CHR$(20);PS$=E$+"@"
70 EM$=E$+"E";CE$=E$+"F";DB$=E$+"G";CD$=E$+"H"
80 LPRINT PS$;E$;"R";CHR$(15)
90 FOR I=1 TO 2
100 IF I=1 THEN LPRINT WD$;" GEMINI STANDARD TYPE STYLES";CM$;LF$;
110 IF I=2 THEN LPRINT IT$;WD$;" GEMINI ITALIC TYPE STYLE";
115 LPRINT CW$;LF$
120 FOR N=1 TO 40:LPRINT CHR$(231);CHR$(232);:NEXT N
130 LPRINT" 17 CPI 12 CPI 10 CPI 8.5 CPI 6 CPI 5 CPI"
140 FOR N=1 TO 80:LPRINT CHR$(241);:NEXT N
150 MSG$(1)=" ABC";MSG$(2)="X"+CHR$(120);MSG$(3)="Y"+CHR$(121)
160 FOR J=1 TO 4
170 IF I=2 THEN LPRINT IT$;
180 IF J=1 THEN LPRINT CM$;"NORMAL ";
190 IF J=2 THEN LPRINT CM$;"DOUBLE STRIKE ";
200 IF J=3 THEN LPRINT CM$;"EMPHASIZED ";
210 IF J=4 THEN LPRINT CM$;"DOUBLE STRIKE ";
220 FOR K=1 TO 6
230 IF K=1 THEN LPRINT CM$;
240 IF K=2 THEN LPRINT EL$;
250 IF K=3 THEN LPRINT PC$;
260 IF K=4 THEN LPRINT CM$;WD$;
270 IF K=5 THEN LPRINT EL$;WD$;
280 IF K=6 THEN LPRINT PC$;WD$;
290 IF J=2 THEN LPRINT DB$;
300 IF J=3 THEN LPRINT EM$;
310 IF J=4 THEN LPRINT DB$;EM$;
320 LPRINT MSG$(1);SP$;MSG$(2);CS$;SB$;MSG$(3);CS$;CD$;
330 IF K>3 THEN LPRINT CM$;
340 NEXT K
350 IF J=4 THEN LPRINT E$;"A";CHR$(8);LF$;CM$;" +";LF$;"EMPHASIZED";
360 LPRINT PS$
365 IF J=2 THEN LPRINT LF$;
```



```

370 NEXT J
380 FOR M=1 TO 5:LPRINT LF$:NEXT M
390 NEXT I
400 END

```

Figure E-1 is the font styles program (refer to Figure 5-1) amended for the Osborne computer.

```

5 CLR%=CHR$(26)
10 PRINT CLR%:DIM A$(22,52):X=1:Y=1
15 S%=CHR$(27)+CHR$(61)
17 DEF FNL$(X,Y)=S%+CHR$(X+32)+CHR$(Y+32)
20 PRINT CLR%:X=1:Y=1
30 GOSUB 150:PRINT FNL$(X,Y);:
40 B%=INKEY%:IF B%="" THEN 40
50 IF B%=CHR$(5) THEN PRINT CLR%:END
60 IF B%=CHR$(18) THEN PRINT CLR%:GOSUB 200:GOTO 20
70 IF B%=CHR$(16) THEN GOTO 210
80 IF B%=CHR$(13) THEN GOSUB 700:GOTO 40
90 IF B%=CHR$(4) THEN GOSUB 130:GOTO 40
100 PRINT FNL$(X,Y);:PRINT B%;A$(X,Y)=B%:Y=Y+1
110 IF Y=50 THEN A$(X,Y)=CHR$(13):PRINT CHR$(7);:X=X+1:Y=1:GOTO 40
120 PRINT FNL$(X,Y);:PRINT " ";:GOTO 40
130 PRINT FNL$(X,Y);:PRINT " ";:Y=Y-1:IF Y=0 THEN Y=1:PRINT CHR$(7);
135 A$(X,Y)=""
140 PRINT FNL$(X,Y);:PRINT " ";:RETURN
150 PRINT FNL$(22,2);
160 PRINT "^P)RINTER ^D)ELETE ^R)ESTART ^E)XIT";
190 RETURN
200 PRINT FNL$(10,20);"PLEASE STAND BY";
205 FOR I=1 TO X:FOR J=1 TO 50:A$(I,J)=" ";
206 NEXT J:NEXT I:RETURN
210 REM
220 LPRINT CHR$(27);"@"
230 PRINT CLR% :PRINT FNL$(23,2);
235 PRINT "C)OMMAND";:PRINT FNL$(23,11);
240 PRINT "P)RINT";FNL$(23,18);"R)ETURN";
250 G%=INKEY%:IF G%="" THEN 250
260 IF G%="P" THEN GOSUB 310

```



```
270 IF G$="C" THEN GOSUB 320:GOTO 230
280 IF G$="R" THEN GOSUB 300:GOTO 30
290 GOTO 250
300 PRINT CLR$:FOR I=1 TO X:FOR J=1 TO 50
302 PRINT FNL$(I,J);:
304 IF A$(I,J)=CHR$(13) THEN 308
306 PRINT A$(I,J);:
308 NEXT J:NEXT I
309 RETURN
310 FOR I=1 TO X:FOR J=1 TO 50:LPRINT A$(I,J);
312 NEXT J:LPRINT :NEXT I:RETURN
320 PRINT FNL$(23,1);SPC(47);
340 PRINT FNL$(23,2);" E) Esc";
350 PRINT FNL$(23,10);" C) CHR$ ";
355 PRINT FNL$(23,20);" Q) END COMMAND ";:ST$=""
360 G$=INKEY$:IF G$="" THEN 360
370 IF G$="E" THEN GOSUB 420:GOTO 400
380 IF G$="C" THEN GOSUB 530:GOTO 400
390 IF G$="Q" THEN GOTO 410
400 GOTO 360
410 RETURN
420 I=0:PRINT FNL$(20,4);:PRINT "Esc ";
430 G$=INKEY$:IF G$="" THEN 430
440 IF G$=CHR$(13) THEN 480
450 PRINT G$;
460 PRINT " ";
470 S$(I)=G$:I=I+1:GOTO 430
480 IF I>2 THEN GOSUB 800
490 IF I=2 THEN GOSUB 850
500 IF I=1 THEN ST$=CHR$(27)+S$(0)
510 LPRINT ST$:ST$="":FOR J=1 TO I:S$(J)="
512 NEXT J:I=0
520 PRINT FNL$(20,4);:PRINT STRING$(20," ")
521 RETURN
530 PRINT FNL$(20,4);" CHR$ ";:ST$=""
540 G$=INKEY$:IF G$="" THEN 540
550 IF G$=CHR$(13) THEN 570
560 PRINT G$;:ST$=ST$+G$:GOTO 540
570 PRINT FNL$(20,4);:PRINT STRING$(10," ")
580 LPRINT CHR$(VAL(ST$));:ST$="":RETURN
700 REM
```



```

710 A$(X,Y)=B$
720 PRINT FNL$(X,Y); " ";
730 X=X+1:Y=Y+1:
740 PRINT FNL$(X,Y);"";:RETURN
800 FOR J=1 TO I:D$=D$+S$(J):NEXT J:
810 ST$=CHR$(27)+S$(0)+CHR$(VAL(D$)):D$=""
820 RETURN
850 ST$=CHR$(27)+S$(0)+CHR$(VAL(S$(1)))
860 RETURN

```

Figure E-2 is the word processing program (refer to Figure 5-4) amended for the Osborne computer.

```

10 LPRINT TAB(30);"HIGH RESOLUTION VERTICAL SPACING"
20 LPRINT TAB(30);"USING THE ESC 3 N CONTROL CODE"
30 FOR I=1 TO 30
40 IF I=13 THEN GOTO 130
50 M=INT(144/(6*I)):IF M<2 THEN M=2
60 LPRINT CHR$(27);CHR$(51);CHR$(I)
70 FOR J=0 TO M
80 IF J=M-1 THEN LPRINT I;"/144 INCH SPACING      ";:GOTO 100
90 LPRINT TAB(30);
100 FOR N=1 TO 40:LPRINT CHR$(241);:NEXT N:LPRINT " "
110 NEXT J
130 NEXT I
140 LPRINT CHR$(27);"@"
150 END

```

Figure E-3 is the vertical spacing program (refer to Figure 6-5) amended for the Osborne computer.



```
10 LPRINT CHR$(27);"@";CHR$(27);"A";CHR$(6)
20 FOR LI=1 TO 2
30 LPRINT CHR$(27);"K";CHR$(114);CHR$(0);
40 FOR SPACE=1 TO 100
50 LPRINT CHR$(0);:NEXT SPACE
60 FOR COLM=1 TO 14
70 READ C:LPRINT CHR$(C);:NEXT COLM
80 LPRINT :NEXT LI
90 DATA 14,16,34,64,140,140,128,128,140,140,64,34,16,14
100 DATA 224,16,232,68,34,18,18,18,18,34,68,232,16,224
110 LPRINT CHR$(27);"@":
120 LPRINT :LPRINT :LPRINT
130 END
```

Figure E-4 is the dot graphics program presented in Chapter 6 (the one that puts a smile on your paper) amended for the Osborne computer.

```
10 LPRINT CHR$(163);CHR$(161);
20 LPRINT CHR$(27);"A";CHR$(3)
30 LPRINT CHR$(162);CHR$(160);
40 LPRINT CHR$(27);"2"
50 END
```

Figure E-5 is the circle program presented in Chapter 7, amended for the Osborne computer.

```
10 LPRINT TAB(2);CHR$(163);CHR$(27);"A";CHR$(6)
20 LPRINT CHR$(160)
30 END
```

Figure E-6 is the tilde program presented in Chapter 7 amended for the Osborne computer.

```
10 LPRINT TAB(20);"N";CHR$(13)
20 LPRINT TAB(20);CHR$(164);CHR$(13)
30 LPRINT TAB(18);"W";CHR$(166);CHR$(32);CHR$(167);"E";CHR$(13)
40 LPRINT TAB(20);CHR$(165);CHR$(13)
50 LPRINT TAB(20);"S"
60 END
```

Figure E-7 is the map program presented in Chapter 7 amended for the Osborne computer.



```

15 LPRINT CHR$(27);"U";CHR$(1)
20 LPRINT CHR$(27);"A";CHR$(6)
30 FOR J=1 TO 9
40 FOR I=1 TO 12
50 READ A
60 LPRINT CHR$(A);
70 NEXT I
80 LPRINT
90 NEXT J
100 LPRINT CHR$(27);"@":
120 END
130 DATA 239,224,224,224,239,224,224,224,224,224,224,224
140 DATA 239,224,224,224,239,224,224,224,224,224,224,224
150 DATA 239,224,224,224,239,224,239,224,224,224,239,224
160 DATA 239,224,239,224,239,224,234,233,224,234,233,224
170 DATA 239,234,239,233,239,224,224,239,224,239,224,224
180 DATA 239,239,224,239,239,224,224,234,239,233,224,224
190 DATA 234,233,224,234,233,224,224,224,239,224,224,224
200 DATA 224,224,224,224,224,224,224,234,233,224,224,224
210 DATA 224,224,224,224,224,224,224,239,224,224,224,224

```

Figure E-8 is the "Wy" program (refer to Figure 7-1) amended for the Osborne computer.

```

10 LPRINT CHR$(27);"A";CHR$(6);CHR$(15)
20 LPRINT CHR$(27);"U";CHR$(1)
30 FOR K=1 TO 3
40 LPRINT CHR$(27);"B";CHR$(K)
50 RESTORE
60 FOR J=1 TO 9
70 FOR I=1 TO 12
80 READ A:LPRINT CHR$(A);
90 NEXT I
100 LPRINT
110 NEXT J
120 LPRINT:NEXT K
130 LPRINT CHR$(27);"@":END
140 DATA 224,224,224,224,224,224,224,224,245,224,224
150 DATA 224,224,224,224,224,224,224,252,245,224,224
160 DATA 224,224,224,245,224,224,224,252,239,245,224,224

```



```
170 DATA 224,224,252,245,224,224,252,239,239,245,224,224
180 DATA 224,252,239,245,224,252,239,239,239,245,224,224
190 DATA 252,239,239,245,252,239,239,239,239,245,224,224
200 DATA 241,241,241,249,241,241,241,241,241,249,224,224
210 DATA 239,239,239,239,239,239,239,239,239,239,239,251
220 DATA 253,239,239,239,239,239,239,239,239,239,251,224
```

Figure E-9 is the boat program (refer to Figure 7-2) amended for the Osborne computer.

```
10 PRINT CHR$(26)
20 PRINT "":PRINT "":PRINT ""
30 PRINT "THIS PROGRAM TAKES ABOUT TWO"
40 PRINT "MINUTES TO RUN, PLEASE TURN"
50 PRINT "ON YOUR PRINTER AND STAND BY"
100 REM MICRO-PLOT
110 DIM BITX(75,12)
1000 REM SET PROGRAM CONSTANTS
1010 MASKZ(1)=128:MASKZ(4)=16
1020 MASKZ(2)=64 :MASKZ(5)=8
1030 MASKZ(3)=32 :MASKZ(6)=4
1040 LX=20 :LY=20
1050 XFAC=72/LX :YFAC=75/LY
2000 REM PLOT CURVE
2010 RAD=9
2020 X1=19 :Y1=10
2030 FOR ANGZ=0 TO 360 STEP 10
2040 R1=ANGZ*6.28/360
2050 X2=RAD*COS(R1)+10 :Y2=RAD*SIN(R1)+10
2060 GOSUB 4000
2070 NEXT ANGZ
2080 FOR ANGZ=0 TO 360 STEP 15
2090 R1=ANGZ*6.28/360
2100 R2=(ANGZ+150)*6.28/360
2110 X1=RAD*COS(R1)+10 :Y1=RAD*SIN(R1)+10
2120 X2=RAD*COS(R2)+10 :Y2=RAD*SIN(R2)+10
2130 GOSUB 4000
2140 NEXT ANGZ
3000 REM SEND BIT IMAGE MAP TO PRINTER
3010 LPRINT CHR$(27);"A";CHR$(6)
3020 FOR ROWZ=0 TO 11
```




```

3030 A$=""
3040 LPRINT CHR$(27);"K";CHR$(119);CHR$(1);
3050 FOR COLZ = 1 TO 75
3055 FOR LL=1 TO 5
3060 LPRINT CHR$(BITZ(COLZ,ROWZ));
3065 NEXT LL
3070 NEXT COLZ
3080 LPRINT
3090 NEXT ROWZ
3100 LPRINT CHR$(27);"A";CHR$(12)
3110 END
4000 REM DRAW A LINE FROM X1,Y1 TO X2,Y2
4010 XL=X2-X1 :YL=Y2-Y1
4020 NX=ABS(XL*XFAC) : NY=ABS(YL*YFAC)
4030 IF NX<NY THEN NX=NY
4040 NSZ=INT(NX+1)
4050 DX=XL/NSZ : DY=YL/NSZ
4060 FOR IZ=1 TO NSZ
4070 X1=X1+DX : Y1=Y1+DY
4080 GOSUB 5000
4090 NEXT IZ
4100 RETURN
5000 REM PLOT A POINT AT X1,Y1
5010 XX=X1*XFAC :YY=Y1*YFAC
5020 COLX=INT(XX)+1
5030 ROWZ=INT(YY/6)
5040 XITZ=INT(YY-ROWZ*6)+1
5050 BITZ(COLZ,ROWZ)=BITZ(COLZ,ROWZ) OR MASKX(XITZ)
5060 RETURN

```

Figure E-10 is the graphics program (refer to Figure 7-4) amended for the Osborne computer.

```

10 REM DEFINE ESCAPE CHARACTER
20 ESC$=CHR$(27)
30 REM DEFINE PARAGRAPH SYMBOL IN PLACE OF @
40 LPRINT ESC$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(6);
45 LPRINT CHR$(25);CHR$(0);CHR$(25);CHR$(11B);CHR$(25);CHR$(0);
47 LPRINT CHR$(127);CHR$(0)
50 REM SELECT DOWNLOADABLE CHARACTERS

```



```
60 LPRINT ESC$;CHR$(36);CHR$(1)
70 REM PRINT OUR NEW SYMBOL
80 LPRINT "#####"
90 END
```

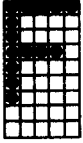
Figure E-11 is the downloading paragraph symbols program (refer to Figure 8-2) amended for the Osborne computer.

```
10 REM DEFINE ESCAPE CHARACTER
20 ESC$=CHR$(27)
30 REM DEFINE COPYRIGHT SYMBOL IN PLACE OF ©
40 LPRINT ESC$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);
45 LPRINT CHR$(34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);CHR$(65);
47 LPRINT CHR$(34);CHR$(20)
50 REM SELECT DOWNLOADABLE CHARACTERS
60 LPRINT ESC$;CHR$(36);CHR$(1)
70 REM PRINT OUR NEW SYMBOL
80 LPRINT "COPY RIGHT © 1983"
90 END
```

Figure E-12 is the downloading copyright program (refer to Figure 8-4) amended for the Osborne computer.

```
10 REM DEFINE ESCAPE CHARACTER
20 ESC$=CHR$(27)
24 REM COPY CHARACTERS TO ROM
25 LPRINT ESC$;CHR$(42);CHR$(0)
30 REM DEFINE COPYRIGHT SYMBOL IN PLACE OF ©
40 LPRINT ESC$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);
45 LPRINT CHR$(34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);CHR$(65);
47 LPRINT CHR$(34);CHR$(20)
50 REM SELECT DOWNLOADABLE CHARACTERS
60 LPRINT ESC$;CHR$(36);CHR$(1)
70 REM PRINT OUR NEW SYMBOL
80 LPRINT "COPY RIGHT © 1983"
90 END
```

Figure E-13 is the downloading program presented in Figure 8-5 amended for the Osborne computer.



TRS-80

The Connecting Cable

Use the cable supplied by your Gemini dealer for your TRS Model II or Model III computer. No interface kit is required for these models.

Setting the DIP Switches

Set switch number 2-3 to the "off" position on your Gemini. You may set switch number 2-1 in either position, at your option.

Switch number 2-2 should be set "off" if you want Gemini to print every time it receives a CR code; set to "on" if you wish to have the carriage return ignored and printing only when the buffer is full.

The TRS-80 computer does not automatically send a line feed at the end of a line. If you want to insert a line feed (when, for example, listing a program), turn DIP switch number 2-4 to the "on" position.

The Use of BASIC

The Model II computer may need to be initialized to properly run the sample programs in this user's manual. From BASIC, type:

SYSTEM "FORMS"

and respond to the questions asked by the computer. Then, type:

Q

and the system will correctly route LPRINT commands to your Gemini.



The TRS-80 Microsoft BASIC is quite similar to the Microsoft BASIC used in the demonstration programs. The only exception is the width commands, which are explained in the documentation accompanying the TRS-80 Microsoft BASIC programs. You may also refer to the amended program listing at the back of this Appendix.

Pin Configurations

The following chart shows the pin-outs for Gemini printers and the TRS-80 Models I and II with expansion interface.

Parallel			
Gemini-10X/15X		Model I	
1.	<u>STROBE</u>	1.	<u>STROBE</u>
2.	D1	3.	D1
3.	D2	5.	D2
4.	D3	7.	D3
5.	D4	9.	D4
6.	D5	11.	D5
7.	D6	13.	D6
8.	D7	15.	D7
9.	D8	17.	D8
11.	BUSY	21.	READY

Parallel			
Gemini-10X/15X		Model II	
1.	<u>STROBE</u>	1.	<u>STROBE</u>
2.	D1	3.	D1
3.	D2	5.	D2
4.	D3	7.	D3
5.	D4	9.	D4
6.	D5	11.	D5
7.	D6	13.	D6
8.	D7	15.	D7
9.	D8	17.	D8
10.	<u>ACK</u>	19.	<u>ACK</u>
11.	BUSY	21.	BUSY



Program Listings

```
10 POKE 16427,255
20 E$=CHR$(27):LF$=CHR$(10):FF$=CHR$(12):CR$=CHR$(13)
30 PC$=E$+"B"+CHR$(1):EL$=E$+"B"+CHR$(2):CM$=E$+"B"+CHR$(3)
40 SP$=E$+"S"+CHR$(0):SB$=E$+"S"+CHR$(1):CS$=E$+"T"
50 IT$=E$+CHR$(52)
60 WD$=CHR$(14):CW$=CHR$(20):PS$=E$+"@"
70 EM$=E$+"E":CE$=E$+"F":DB$=E$+"G":CD$=E$+"H"
80 LPRINT PS$;E$;"R";CHR$(15);
90 FOR I=1 TO 2
100 IF I=1 THEN LPRINT WD$;" GEMINI STANDARD TYPE STYLES";CW
$;LF$
110 IF I=2 THEN LPRINT IT$;WD$;" GEMINI ITALIC TYPE STYLE";
CW$;LF$
120 FOR N=1 TO 40:LPRINT CHR$(231);CHR$(232);:NEXT N:LPRINT:LP
RINT
130 LPRINT "      17 CPI  12 CPI  10 CPI   8.5 CPI   6 CPI
      5 CPI"
140 FOR N=1 TO 80:LPRINT CHR$(241);:NEXT N:LPRINT
150 M1$=" ABC":M2$="X"+CHR$(120):M3$="Y"+CHR$(121)
160 FOR J=1 TO 4
170 IF I=2 THEN LPRINT IT$;
180 IF J=1 THEN LPRINT CM$;"NORMAL      ";
190 IF J=2 THEN LPRINT CM$;"DOUBLE STRIKE ";
200 IF J=3 THEN LPRINT CM$;"EMPHASIZED  ";
210 IF J=4 THEN LPRINT CM$;"DOUBLE STRIKE ";
220 FOR K=1 TO 6
230 IF K=1 THEN LPRINT CM$;
240 IF K=2 THEN LPRINT EL$;
250 IF K=3 THEN LPRINT PC$;
260 IF K=4 THEN LPRINT CM$;WD$;
270 IF K=5 THEN LPRINT EL$;WD$;
280 IF K=6 THEN LPRINT PC$;WD$;
290 IF J=2 THEN LPRINT DB$;
300 IF J=3 THEN LPRINT EM$;
310 IF J=4 THEN LPRINT DB$;EM$;
320 LPRINT M1$;SP$;M2$;CS$;SB$;M3$;CS$;CD$;
330 IF K>3 THEN LPRINT CW$;
340 NEXT K
```



```

350 IF J=4 THEN LPRINT E$;"A";CHR$(8);LF$;CN$;"    + ";CR$;"E
MPHASIZED";
360 LPRINT PS$
370 IF J=2 THEN LPRINT LF$;
375 NEXT J
380 FOR M=1 TO 3:LPRINT :NEXT M
390 NEXT J
400 END

```

Figure F-1 is the font styles program (refer to Figure 5-1) amended for the TRS-80.

```

5 CLEAR 2000
7 C$=CHR$(179)
10 CLS:DIM A$(16,62)
20 CLS:X=0:Y=0
30 GOSUB 150:PRINT @X$64+Y,C$;
40 B$=INKEY$:IF B$="" THEN 40
50 IF B$=CHR$(64) THEN CLS:END
60 IF B$=CHR$(31) THEN CLS:GOSUB 200:GOTO 20
70 IF B$=CHR$(9) THEN GOTO 210
80 IF B$=CHR$(13) THEN A$(X,Y)=B$:PRINT @X$64+Y," ":X=X+1:Y=0:PRINT @X$64+Y,C$;:GOTO 40
90 IF B$=CHR$(8) THEN GOSUB 130:GOTO 40
100 PRINT @X$64+Y,B$;:A$(X,Y)=B$:Y=Y+1:
110 IF Y=62 THEN A$(X,Y)=CHR$(13):X=X+1:Y=0:GOTO 40
120 PRINT @X$64+Y,C$;:GOTO 40
130 PRINT @X$64+Y," ";:Y=Y-1:IF Y<0 THEN Y=0:
140 A$(X,Y)="" :PRINT @X$64+Y,C$;:RETURN
150 PRINT @15$64+0," <-- BACK SPCAE --> PRINT <CLEAR>RESTART @ EXIT";:RETURN
200 PRINT @7$64+20,"PLEASE STAND BY";
205 FOR I=0 TO X:FOR J=0 TO 62:A$(I,J)="" :NEXT J:NEXT I
206 RETURN
210 REM
220 LPRINT CHR$(27);"@
230 CLS:PRINT @15$64+2,"COMMAND P)PRINT R)RETURN";
250 G$=INKEY$:IF G$="" THEN 250
260 IF G$="P" THEN GOSUB 310
270 IF G$="C" THEN GOSUB 320:GOTO 230
280 IF G$="R" THEN GOSUB 300:GOTO 30
290 GOTO 250
300 CLS:FOR I=0 TO X:FOR J=0 TO 62

```



```
302 PRINT @I#64+J,A$(I,J);:NEXT J:NEXT I:RETURN
310 FOR I=0 TO X:FOR J=0 TO 62:LPRINT A$(I,J);
312 NEXT J:NEXT I:RETURN
320 PRINT @15#64+2,STRING$(50," ");
330 PRINT @15#64+2," E) ESC  C) CHR$  Q) END COMMAND ";
340 ST$=""
360 G$=INKEY$:IF G$="" THEN 360
370 IF G$="E" THEN GOSUB 420:GOTO 400
380 IF G$="C" THEN GOSUB 530:GOTO 400
390 IF G$="Q" THEN GOTO 410
400 GOTO 360
410 RETURN
420 I=0:PRINT @13#64+3,"Esc ";
430 G$=INKEY$:IF G$="" THEN 430
440 IF G$=CHR$(13) THEN 480
450 PRINT G$;" ";
470 S$(I)=G$:I=I+1:GOTO 430
480 IF I>2 THEN FOR J=1 TO I:D$=D$+S$(J):NEXT J:ST$=CHR$(27)+S$(0)+CHR$(VAL(D$)):D$="";
490 IF I=2 THEN ST$=CHR$(27)+S$(0)+CHR$(VAL(S$(1)));
500 IF I=1 THEN ST$=CHR$(27)+S$(0)
510 LPRINT ST$:ST$="":FOR J=1 TO I:S$(J)="":NEXT J:I=0
520 PRINT @13#64+2,STRING$(20," ");:RETURN
530 PRINT @13#64+3," CHR$ ";:ST$=""
540 G$=INKEY$:IF G$="" THEN 540
550 IF G$=CHR$(13) THEN 570
560 PRINT G$;:ST$=ST$+G$:GOTO 540
570 PRINT @13#64+2,STRING$(20," ");
580 LPRINT CHR$(VAL(ST$));:ST$="":RETURN
```

Figure F-2 is the word processing program (refer to Figure 5-4) amended for the TRS-80.

```
10 LPRINT TAB(30);"HIGH RESOLUTION VERTICAL SPACING "  
15 LPRINT TAB(30);"USING THE ESC 3 N CONTROL CODE "  
20 FOR I = 1 TO 30  
25 IF I=12 THEN 110  
30 IF I = 13 THEN GOTO 110  
40 M = INT(144/(68I)): IF M < 2 THEN M = 2  
50 LPRINT CHR$(27);CHR$(51);CHR$(I);  
60 FOR J = 0 TO M
```




```

70 IF J = M-1 THEN LPRINT I;"/144 INCH SPACING           ";:GOT
D 90
80 LPRINT TAB(30);
90 FOR N = 1 TO 40 : LPRINT CHR$(241); : NEXT N:LPRINT
100 NEXT J
110 NEXT I
120 LPRINT CHR$(27);"@":END
    
```

Figure F-3 is the vertical spacing program (refer to Figure 6-5) amended for the TRS-80.

```

10 RESTORE
20 LPRINT CHR$(27);"@";CHR$(27);"1"
30 FOR LI = 1 TO 2
40 LPRINT CHR$(27);"K";CHR$(114);CHR$(0);
50 FOR SPACE = 1 TO 100:LPRINT CHR$(0);:NEXT SPACE
60 FOR COLM = 1 TO 14:READ C:LPRINT CHR$(C);:NEXT COLM
70 LPRINT : NEXT LI
80 DATA 14,16,34,64,140,140,128,128,140,140,64,34,16,14
90 DATA 224,16,232,68,34,18,18,18,18,34,68,232,16,224
100 LPRINT CHR$(27);"@":LPRINT:LPRINT:LPRINT
    
```

Figure F-4 is the dot graphics program presented in Chapter 6 (the one that puts a smile on your paper) amended for the TRS-80.

```

10 LPRINT CHR$(163);CHR$(161);CHR$(27);"A";CHR$(6)
20 LPRINT CHR$(162);CHR$(160);CHR$(27);"2"
30 END
    
```

Figure F-5 is the circle program presented in Chapter 7, amended for the TRS-80.

```

10 LPRINT TAB(1);CHR$(163);CHR$(27);"A";CHR$(6)
20 LPRINT CHR$(160)
30 END
    
```

Figure F-6 is the tilde program presented in Chapter 7 amended for the TRS-80.



```
10 LPRINT TAB(20);"N";CHR$(13);TAB(20);CHR$(164);CHR$(13);TAB(
16);"W ";CHR$(166);CHR$(32);TAB(22);CHR$(167);" E";CHR$(13);TA
B(20);CHR$(165);CHR$(13);TAB(20);"S"
15 END
```

Figure F-7 is the map program presented in Chapter 7 amended for the TRS-80.

```
10 LPRINT CHR$(27);"A";CHR$(6)
15 LPRINT CHR$(27);"U";CHR$(1)
20 FOR J = 1 TO 9
30 FOR I = 1 TO 12
40 READ A
50 LPRINT CHR$(A);
60 NEXT I
70 LPRINT
80 NEXT J
90 DATA 239,224,224,224,239,224, 224,224,224,224,224,224
91 DATA 239,224,224,224,239,224, 224,224,224,224,224,224
92 DATA 239,224,224,224,239,224, 239,224,224,224,239,224
93 DATA 239,224,239,224,239,224, 234,233,224,234,233,224
94 DATA 239,234,239,233,239,224, 224,239,224,239,224,224
95 DATA 239,239,224,239,239,224, 224,234,239,233,224,224
96 DATA 234,233,224,234,233,224, 224,224,239,224,224,224
97 DATA 224,224,224,224,224,224, 224,234,233,224,224,224
98 DATA 224,224,224,224,224,224, 224,239,224,224,224,224
100 LPRINT CHR$(27);"@":END
```

Figure F-8 is the "Wy" program (refer to Figure 7-1) amended for the TRS-80.

```
5 LPRINT CHR$(27);"U";CHR$(1)
10 LPRINT CHR$(27);"A";CHR$(6);CHR$(15)
15 FOR K = 1 TO 3
16 LPRINT CHR$(27);"B";CHR$(K)
17 RESTORE
20 FOR J = 1 TO 9
30 FOR I = 1 TO 12
40 READ A
50 LPRINT CHR$(A);
```



```

60 NEXT I
70 LPRINT
80 NEXT J
90 DATA 224,224,224,224,224,224, 224,224,224,245,224,224
91 DATA 224,224,224,224,224,224, 224,224,252,245,224,224
92 DATA 224,224,224,245,224,224, 224,252,239,245,224,224
93 DATA 224,224,252,245,224,224, 252,239,239,245,224,224
94 DATA 224,252,239,245,224,252, 239,239,239,245,224,224
95 DATA 252,239,239,245,252,239, 239,239,239,245,224,224
96 DATA 241,241,241,249,241,241, 241,241,241,249,224,224
97 DATA 239,239,239,239,239,239, 239,239,239,239,239,251
98 DATA 253,239,239,239,239,239, 239,239,239,239,251,224
99 LPRINT :NEXT K
100 LPRINT CHR$(27);"@":END

```

Figure F-9 is the boat program (refer to Figure 7-2) amended for the TRS-80.

```

10 CLS:PRINT "":PRINT "":PRINT ""
20 PRINT "THIS PROGRAM TAKES ABOUT 4 MINUTES TO RUN"
30 PRINT "PLEASE TURN ON YOUR PRINTER AND STAND BY"
100 DIM BIT$(76,12)
110 MASK$(1)=128:MASK$(4)=16
120 MASK$(2)=64 :MASK$(5)=8
130 MASK$(3)=32 :MASK$(6)=4
140 LX=20:LY=20
150 XFAC=72/LX:YFAC=75/LY
1000 REM PLOT CURVE
1010 RAD=°
1020 X1=19:Y1=10
1030 FOR ANG=0 TO 360 STEP 10
1040 R1=ANG*6.28/360
1050 X2=RAD*COS(R1)+10:Y2=RAD*SIN(R1)+10
1060 GOSUB 3000
1070 NEXT
1080 FOR ANG=0 TO 360 STEP 15
1090 R1=ANG*6.28/360
1100 R2=(ANG+150)*6.28/360
1110 X1=RAD*COS(R1)+10:Y1=RAD*SIN(R1)+10
1120 X2=RAD*COS(R2)+10:Y2=RAD*SIN(R2)+10

```



```
1130 GOSUB 3000
1140 NEXT
2000 GOSUB 5000
2010 LPRINT CHR$(27);"A";CHR$(6)
2020 FOR ROW=0 TO 11
2040 LPRINT CHR$(27);"K";CHR$(119);CHR$(1);
2050 FOR COL=1 TO 75
2060 FOR LL=1 TO 5
2070 LPRINT CHR$(BITZ(COL,ROW));
2080 NEXT:NEXT:LPRINT:NEXT:LPRINT
2090 END
3000 REM DRAW A LINE FROM X1,Y1 TO X2,Y2
3010 XL=X2-X1:YL=Y2-Y1
3020 NX=ABS(XL*XFAC):NY=ABS(YL*YFAC)
3030 IF NX<NY THEN NX=NY
3040 NSZ=INT(NX+1)
3050 DX=XL/NSZ:DY=YL/NSZ
3060 FOR I=1 TO NSZ
3070 X1=X1+DX:Y1=Y1+DY
3080 GOSUB 4000
3090 NEXT I
3100 RETURN
4000 REM PLOT A POINT AT X1,Y1
4010 XX=X1*XFAC:YY=Y1*YFAC
4020 COL=INT(XX+1)
4030 ROW=INT(YY/6)
4040 XITZ=INT(YY-(6*ROW))+1
4050 BITZ(COL,ROW)=BITZ(COL,ROW) OR MASKZ(XITZ)
4060 RETURN
5000 FOR I=0 TO 11:FOR J=1 TO 75
5010 IF BITZ(J,I)=12 THEN BITZ(J,I)=14
5020 NEXT:NEXT:RETURN
```

Figure F-10 is the graphics program (refer to Figure 7-4) amended for the TRS-80.

```
10 REM DEFINE THE ESCAPE CHARACTER
20 ESC%=CHR$(27)
30 REM DEFINE PARAGRAPH SYMBOL IN PLACE OF @
40 LPRINT ESC%;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(6);CHR$(9);CHR$(0);
```



```

CHR$(9);CHR$(118);CHR$(9);CHR$(0);CHR$(127);CHR$(0)
50 REM SELECT DOWNLOADABLE CHARACTERS
60 LPRINT ESC$;CHR$(36);CHR$(1)
70 REM PRINT OUR NEW SYMBOL
80 LPRINT "#####"
90 END

```

Figure F-11 is the downloading paragraph symbols program (refer to Figure 8-2) amended for the TRS-80.

```

10 REM DEFINE THE ESCAPE CHARACTER
20 ESC$=CHR$(27)
30 REM DEFINE COPYRIGHT SYMBOL IN PLACE OF ©
40 LPRINT ESC$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);CHR$(34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);CHR$(65);CHR$(34);CHR$(20)
50 REM SELECT DOWNLOADABLE CHARACTERS
60 LPRINT ESC$;CHR$(36);CHR$(1);
70 REM PRINT OUR NEW SYMBOL
80 LPRINT "COPY RIGHT © 1983"
90 END

```

Figure F-12 is the downloading copyright program (refer to Figure 8-4) amended for the TRS-80.

```

10 REM DEFINE THE ESCAPE CHARACTER
20 ESC$=CHR$(27)
24 REM COPY CHARACTERS TO RAM
25 LPRINT ESC$;CHR$(42);CHR$(0)
30 REM DEFINE COPYRIGHT SYMBOL IN PLACE OF ©
40 LPRINT ESC$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);CHR$(34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);CHR$(65);CHR$(34);CHR$(20)
50 REM SELECT DOWNLOADABLE CHARACTERS
60 LPRINT ESC$;CHR$(36);CHR$(1);
70 REM PRINT OUR NEW SYMBOL
80 LPRINT "COPY RIGHT © 1983"
90 END

```

Figure F-13 is the downloading program presented in Figure 8-5 amended for the TRS-80.

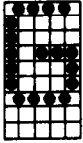


```
10 REM DEFINE THE ESCAPE CHARACTER
20 ESC%=CHR$(27)
24 REM COPY CHARACTERS TO RAM
25 LPRINT ESC%;CHR$(42);CHR$(0)
30 REM DEFINE COPYRIGHT SYMBOL IN PLACE OF ©
40 LPRINT ESC%;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);CHR$(
  34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);CHR$(65);CHR$(34);CHR
  $(20)
44 REM DEFINE COPYRIGHT SYMBOL IN PLACE OF &
45 LPRINT ESC%;CHR$(42);CHR$(1);CHR$(38);CHR$(1);CHR$(28);CHR$(
  34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);CHR$(65);CHR$(34);CHR
  $(20)
50 REM SELECT DOWNLOADABLE CHARACTERS
60 LPRINT ESC%;CHR$(36);CHR$(1);
70 REM PRINT OUR NEW SYMBOL
80 LPRINT "#####"
90 END
```

Figure F-14 is the downloading program presented in Figure 8-6 amended for the TRS-80.

```
10 REM DEFINE ESCAPE CHARACTER
20 ESC%=CHR$(27)
30 REM DEFINE MACRO TO BE EMPHASIZED AND EXPANDED TEXT
40 LPRINT ESC%;CHR$(43);ESC%;"E";ESC%;"M";CHR$(1);CHR$(30)
50 REM ISSUE MACRO COMMAND AND TRY PRINTING
60 LPRINT ESC%;CHR$(33)
70 LPRINT "EMPHASIZED AND EXPANDED BY MACRO"
80 END
```

Figure F-15 is the macro-commands program (refer to Figure 8-7) amended for the TRS-80.



WHEN YOU FIRST GET YOUR NEW GEMINI

This appendix serves the needs of Gemini's first-time users. On the following pages, you will learn what to look for when initially opening the box containing your new printer. In addition to learning what the pieces are, you'll gain instruction in how the pieces fit together. Specifically, this appendix will show you how to remove the packing and attach the paper-handling accessories.

CHECKLIST FOR FIRST OPENING THE BOX

When you first open the shipping/purchase carton for your Gemini printer, you should find the following items:

CHECKLIST FOR YOUR NEW GEMINI

- one Gemini printer
- one paper guide (a smaller metal rack)
- one paper separator (a metal rack)
- one roll-paper holder (a formed metal bar)
- one holder shaft (a long plastic cylinder)
- one ink ribbon (mounted on two spools)
- one spare fuse (small, glass & metal)
- one user's manual (this one, right here)



WHAT THE PARTS LOOK LIKE

Figure G-1 shows these items for the Gemini-10X, and Figure G-2 presents them for the Gemini-15X.

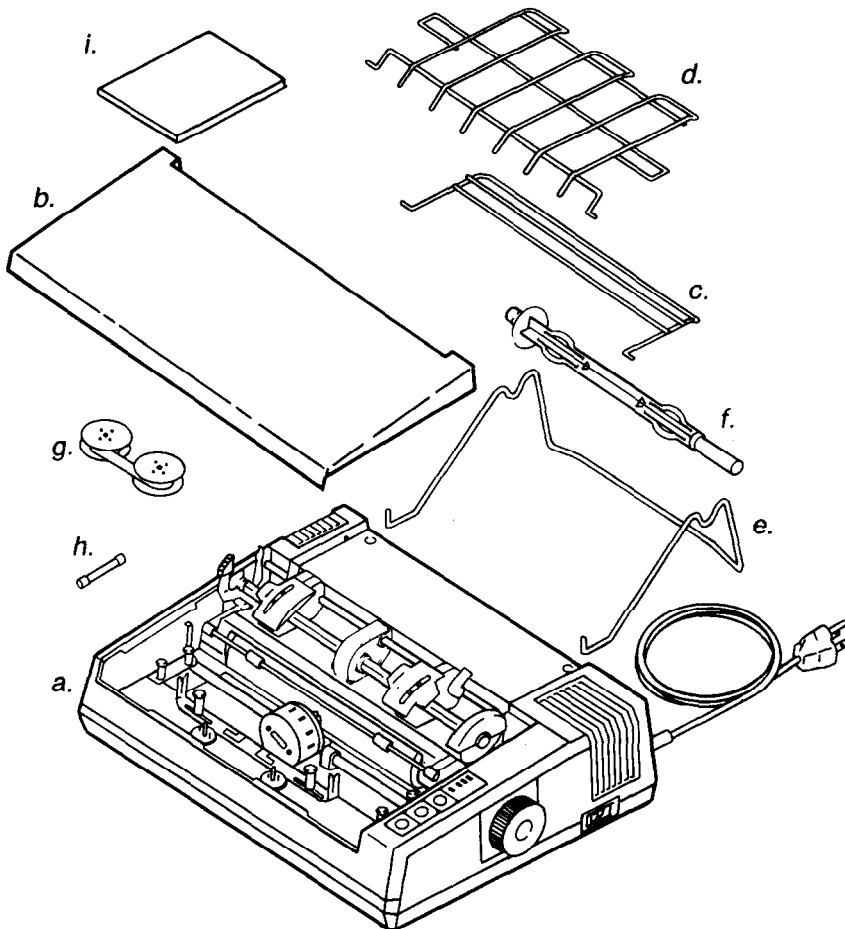


Figure G-1. The Gemini-10X packing list.

- Legend:
- | | |
|----------------------|------------------|
| a. printer | f. holder shaft |
| b. printer cover | g. ink ribbon |
| c. paper guide | h. spare fuse |
| d. paper separator | i. user's manual |
| e. roll paper holder | |

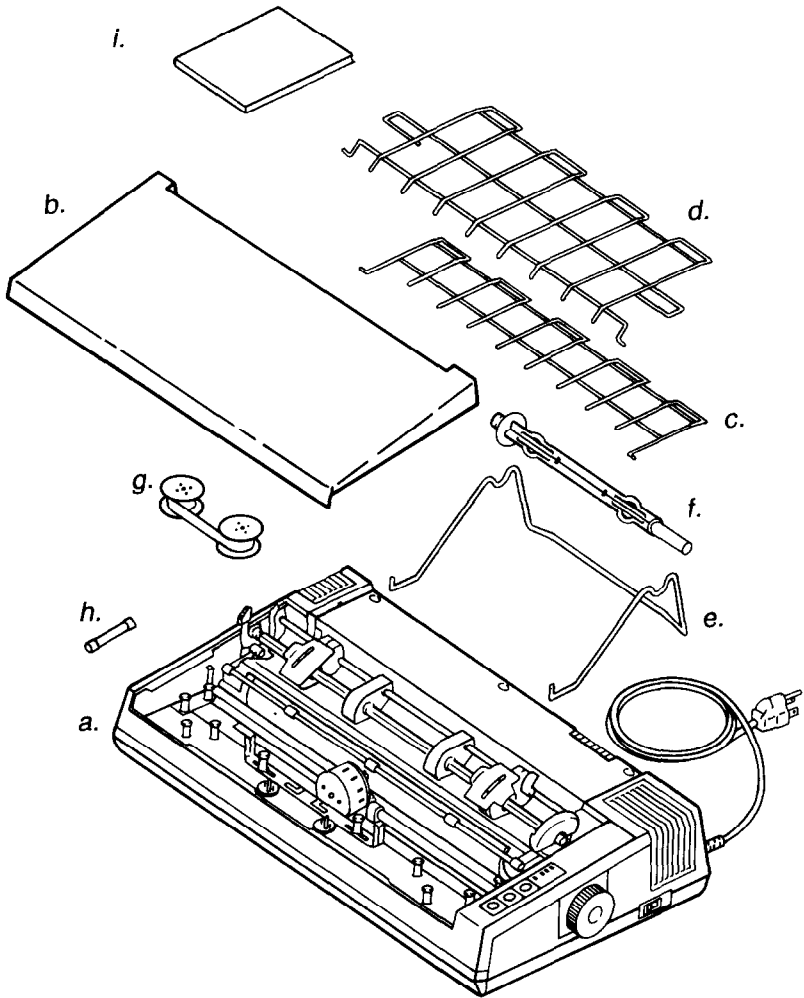


Figure G-2. The Gemini-15X packing list.

REMOVING THE PACKING FROM INSIDE THE PRINTER

Your Gemini came to you nestled in foam plastic cushions. Packing material was also placed inside your printer to secure the print head and platen during shipping. Figure G-3 identifies the packing material that you need to remove from inside Gemini prior to operation.

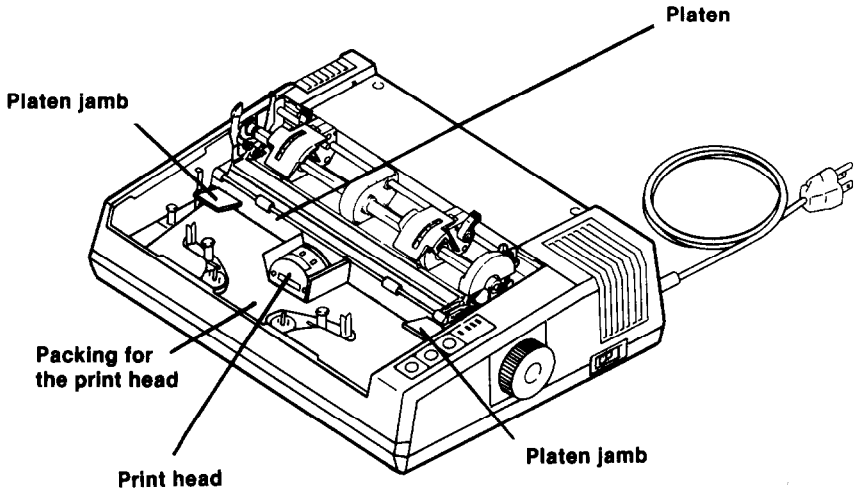


Figure G-3. The Gemini-10X (shown here) and the Gemini-15X both use a cardboard form to secure the print head, cardboard inserts to secure the platen from jostling around.

Note: Be sure you remove all the cardboard packing, before using your Gemini.

First, you need to remove the cardboard packing shown in Figure G-3.

Note: You might want to save these packing materials, in case you move, or in case you need to ship your Gemini printer.

After you remove the platen-securing packing, while your printer is turned "off," turn the platen knob to roll out the sheet of paper that is wrapped around the platen (to protect the "out of paper" sensor).



REMOVAL OF THE "FIXING" SCREWS

When your new printer arrives, there are two screws that "fix" the internal chassis to the external frame. This is similar to "fixing" screws on many phonographic turntables. You need to remove these screws to allow the chassis to "float" on its internal shock absorbers made of rubber-like soft plastic.

To remove the fixing screws, you first need to get to them. Turn your printer over and set it on a pillow or fabric-covered surface. (This is to keep from "skinning" the top of your printer while you make this quick adjustment.)

As shown in Figure G-4, remove the two packing screws. Figure G-4 illustrates the bottom of the Gemini-15X, which has a slot for feeding sprocket paper through the printer's underneath (called "bottom pass"). If you have a Gemini-10X the underside of your printer will not look exactly like the Gemini-15X but there are only two screws that can be removed, so you can't go wrong.

Note: You should save the screws, too. It's a good idea to tape the saved screws together, perhaps with a note indicating what they are for. In the excitement of getting your new Gemini up and running, it's easy to lose the fixing screws. Plus, you will want to re-position these screws if you ever ship your printer.

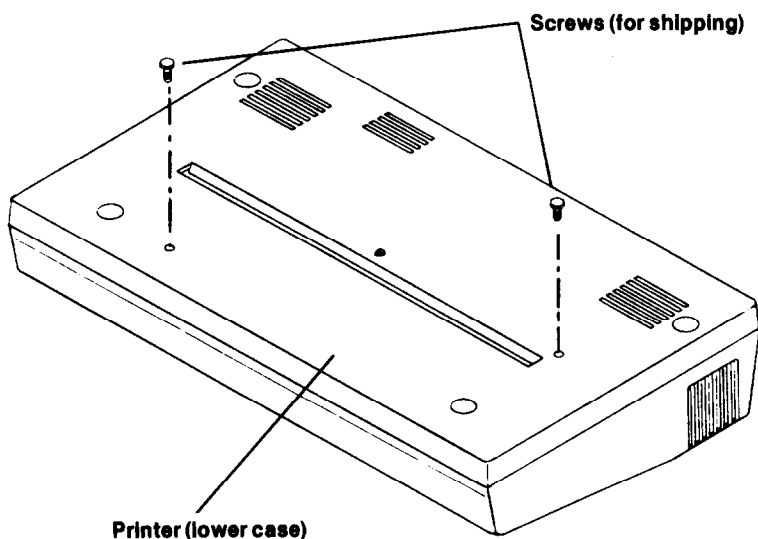


Figure G-4. Turn your printer over and place it on a padded cushion, then remove the two "fixing" screws with a Phillips-head screwdriver.



ATTACHING THE PAPER SEPARATOR AND GUIDE

After you have removed the fixing screws, turn your printer back over so it's right-side-up.

In Chapter 2, you'll learn how to install the ribbon, set the print head gap, and load three types of paper. Those three types of paper are:

- ★ Individual sheets (such as stationery)
- ★ Sprocket-fed computer fanfold paper
- ★ Rolls of paper

For individual sheets of paper, you don't need any of these paper-feeding attachments. However, for sprocket-fed paper, you need to attach the paper separator and paper guide, as shown in Figures G-5 and G-6.

Note: To insert these paper-feeding accessories, insert one side, then gently bend back the other protruding end until it slips into place.

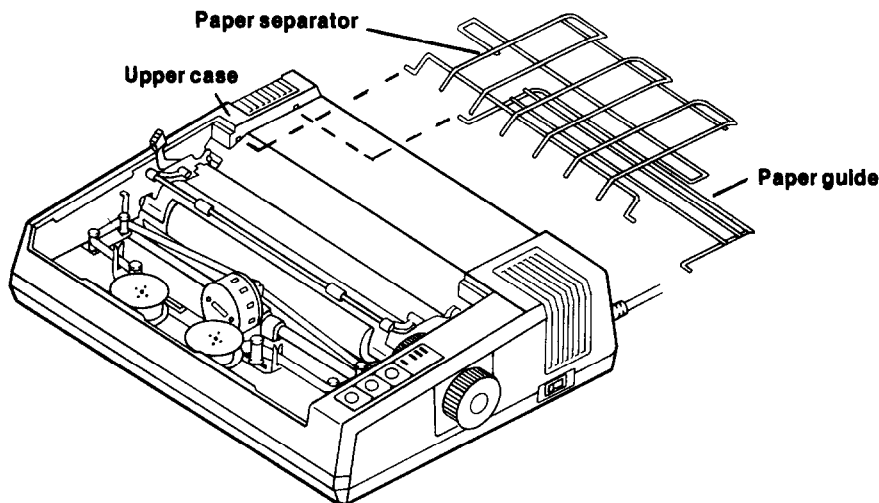


Figure G-5. The paper guide and paper separator are inserted in the notches (right and left) indicated for one side in this illustration of the Gemini-10X

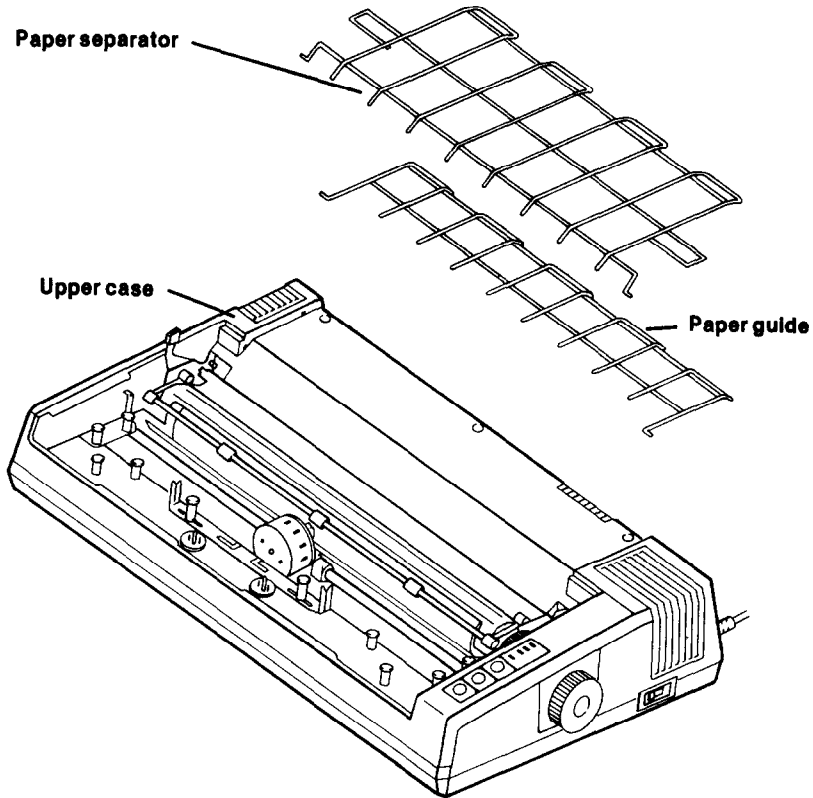


Figure G-6. For the Gemini-15X the paper guide and separator are wider because the Gemini-15X is wider than the Gemini-10X. But these accessories attach in the same manner on both printers.

Another note: Be sure you don't get these racks in upside down. When you are finished, they should look like the Gemini's illustrated in Chapter 2 (Figures 2-4 through 2-6, and 2-8 through 2-13).

The purpose of these racks is to uniformly feed the paper (roll-type or sprocket-fed) between them and into the printing unit. Thus, you should keep these racks free from obstacles such as paper clips or pencils.



ATTACHING THE ROLL PAPER HOLDER & SHAFT

Roll paper requires another accessory in addition to the paper separator and paper guide just introduced. As shown in Figure G-7 (for the Gemini-10X) the hooks of the holder should be inserted in the two holes on the undercarriage on the back side of the printer. (On the Gemini-15X the identical holder attaches in the same manner, but instead of in the middle, toward one side — away from the electrical plug.)

The paper holder is the rack that holds the roll of paper, much the same as some types of paper-towel dispensers. The roll of paper is placed on a holder shaft (as shown in Figure G-8).

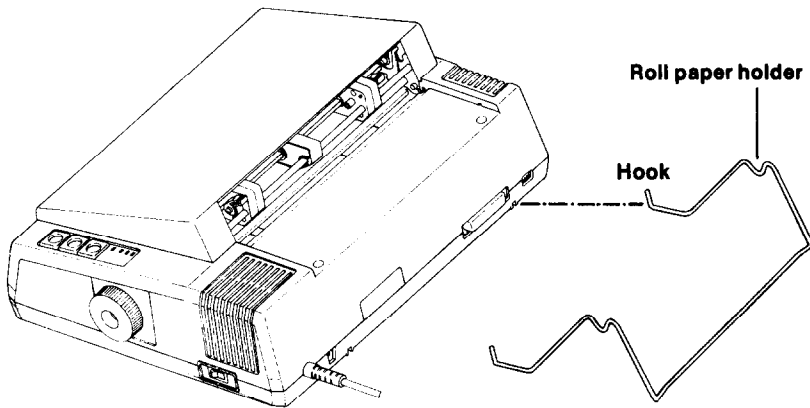


Figure G-7. The roll-paper holder is attached to the back of the printer — in the center for the Gemini-10X and away from the electrical plug for the Gemini-15X.

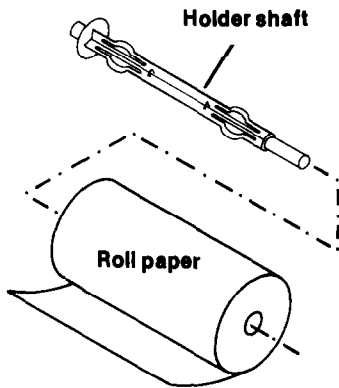


Figure G-8. The holder shaft is inserted in the roll of paper, which is then mounted on the roll-paper holder, as shown in Figure G-9.

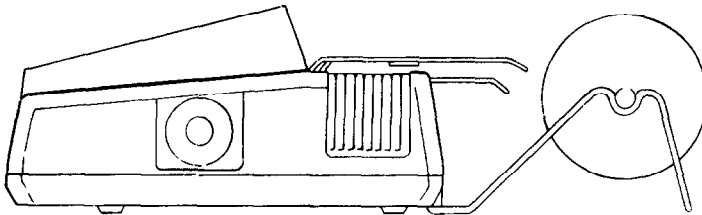
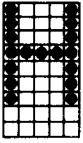


Figure G-9. Side-view of the Gemini printer, showing the three paper-feed accessories in place.

This concludes your introduction to the items contained in the Gemini printer package. Now you should turn to the beginning of this manual.

Chapter 1 will explain the basic parts of the printer. Chapter 2 will show you how to load the paper and the ink ribbon. Chapter 3 introduces the Gemini Self-Test. Chapter 4 shows you how to connect Gemini to your computer (supported by Appendices A through F, which are responsive to specific computer systems). Chapter 5 is a "quick course" in putting Gemini to work for you. The remaining chapters and appendices provide specialized and technical information. If you have a question, consult the index and the table of contents.



ASCII CODE CHARTS

Standard ASCII Character Set

NUL		SP	0	@	P	`	p
0	16	32	48	64	80	96	112
	DC1	!	1	A	Q	a	q
1	17	33	49	65	81	97	113
	DC2	"	2	B	R	b	r
2	18	34	50	66	82	98	114
	DC3	#	3	C	S	c	s
3	19	35	51	67	83	99	115
	DC4	\$	4	D	T	d	t
4	20	36	52	68	84	100	116
		%	5	E	U	e	u
5	21	37	53	69	85	101	117
		&	6	F	V	f	v
6	22	38	54	70	86	102	118
BEL		'	7	G	W	g	w
7	23	39	55	71	87	103	119
BS		(8	H	X	h	x
8	24	40	56	72	88	104	120
HT)	9	I	Y	i	y
9	25	41	57	73	89	105	121
LF		*	:	J	Z	j	z
10	26	42	58	74	90	106	122
VT	ESC	+	;	K	[k	{
11	27	43	59	75	91	107	123
FF		,	<	L	\	l	
12	28	44	60	76	92	108	124
CR		-	=	M]	m	}
13	29	45	61	77	93	109	125
SO	RS	.	>	N	^	n	~
14	30	46	62	78	94	110	126
SI		/	?	O	_	o	DEL
15	31	47	63	79	95	111	127



Block Graphic and Special Character Set

		⌒	T _x	Ā	¥	SP	┌
128	144	160	176	192	208	224	240
	DC1	⌒	Å	â	Ä	■	—
129	145	161	177	193	209	225	241
	DC2	⌒	ø	ç	Ö	■	┐
130	146	162	178	194	210	226	242
	DC3	⌒	θ	£	Ü	■	└
131	147	163	179	195	211	227	243
	DC4	↑	°C	ā	¢	■	┘
132	148	164	180	196	212	228	244
		↓	°F	μ	Ñ	■	
133	149	165	181	197	213	229	245
		←	Ω	°	ä	■	L
134	150	166	182	198	214	230	246
BEL		→	ϖ	'	ö	■	└┘
135	151	167	183	199	215	231	247
BS		○	Σ	†	ü	■	└┘
136	152	168	184	200	216	232	248
HT		△	σ	§	β	■	└┘
137	153	169	185	201	217	233	249
LF		▽	∞	Ē	ē	■	+
138	154	170	186	202	218	234	250
VT	ESC	▷	π	©	é	■	▲
139	155	171	187	203	219	235	251
FF		◁	±	¼	ú	■	▲
140	156	172	188	204	220	236	252
CR		◇	○	Ī	è	■	▲
141	157	173	189	205	221	237	253
SO	RS	◆	×	½	ñ	■	▲
142	158	174	190	206	222	238	254
SI		□	÷		f	■	
143	159	175	191	207	223	239	255

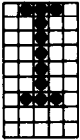


Italic ASCII Character Set

NUL		SP	0	@	P	'	p
0	16	32	48	64	80	96	112
	DC1	!	1	A	Q	a	q
1	17	33	49	65	81	97	113
	DC2	"	2	B	R	b	r
2	18	34	50	66	82	98	114
	DC3	#	3	C	S	c	s
3	19	35	51	67	83	99	115
	DC4	\$	4	D	T	d	t
4	20	36	52	68	84	100	116
		%	5	E	U	e	u
5	21	37	53	69	85	101	117
		&	6	F	V	f	v
6	22	38	54	70	86	102	118
BEL		'	7	G	W	g	w
7	23	39	55	71	87	103	119
BS		(8	H	X	h	x
8	24	40	56	72	88	104	120
HT)	9	I	Y	i	y
9	25	41	57	73	89	105	121
LF		*	:	J	Z	j	z
10	26	42	58	74	90	106	122
VT	ESC	+	;	K	[k	{
11	27	43	59	75	91	107	123
FF		,	<	L	\	l	/
12	28	44	60	76	92	108	124
CR		-	=	M]	m	}
13	29	45	61	77	93	109	125
SO	RS	.	>	N	^	n	~
14	30	46	62	78	94	110	126
SI		/	?	O	_	o	DEL
15	31	47	63	79	95	111	127



Notes

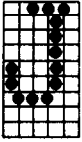


ASCII CODE TABLES

DEC	0	16	32	48	64	80	96	112	128	144	160	176	192	208	224	240	
HEX	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
0	0	NUL		SP	0	@	P	`	p			✓	T _x	Ā	≠	SP	Γ
1	1		DC1	!	1	A	Q	a	q		DC1	✓	Ā	à	Ä	■	—
2	2		DC2	"	2	B	R	b	r		DC2	✓	ø	ç	Ö	■	⌈
3	3		DC3	#	3	C	S	c	s		DC3	✓	ø	£	Ü	■	⌈
4	4		DC4	\$	4	D	T	d	t		DC4	†	°C	ā	€	■	⌈
5	5			%	5	E	U	e	u			↓	°F	μ	Ñ	■	
6	6			&	6	F	V	f	v			←	Ω	°	ä	■	L
7	7	BEL		'	7	G	W	g	w	BEL		→	ö	'	ö	■	⌋
8	8	BS		(8	H	X	h	x	BS		○	Σ	†	ü	■	⌋
9	9	HT)	9	I	Y	i	y	HT		△	σ	§	β	■	⌋
10	A	LF		*	:	J	Z	j	z	LF		▽	∞	Ē	ē	■	+
11	B	VT	ESC	+	;	K	[k	{	VT	ESC	▷	π	©	é	■	↗
12	C	FF		,	<	L	\	l	;	FF		◁	±	' ₄	ú	■	↘
13	D	CR		-	=	M]	m	}	CR		◇	○	×	è	■	↘
14	E	SO	RS	.	>	N	^	n	~	SO	RS	◆	x	' ₂	ñ	■	↘
15	F	SI		/	?	O	_	o	DEL	SI		□	+		f	■	

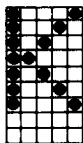
0	0	NUL		SP	0	@	P	`	p
1	1		DC1	!	1	A	Q	a	q
2	2		DC2	"	2	B	R	b	r
3	3		DC3	#	3	C	S	c	s
4	4		DC4	\$	4	D	T	d	t
5	5			%	5	E	U	e	u
6	6			&	6	F	V	f	v
7	7	BEL		'	7	G	W	g	w
8	8	BS		(8	H	X	h	x
9	9	HT)	9	I	Y	i	y
10	A	LF		*	:	J	Z	j	z
11	B	VT	ESC	+	;	K	[k	{
12	C	FF		,	<	L	\	l	;
13	D	CR		-	=	M]	m	}
14	E	SO	RS	.	>	N	^	n	~
15	F	SI		/	?	O	_	o	DEL

} italic type



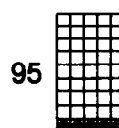
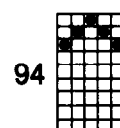
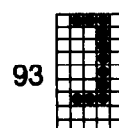
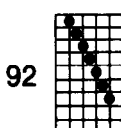
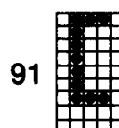
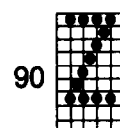
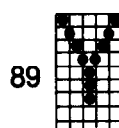
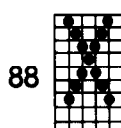
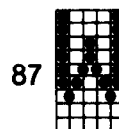
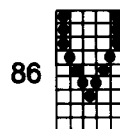
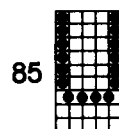
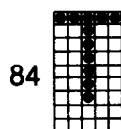
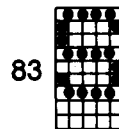
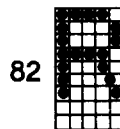
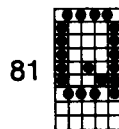
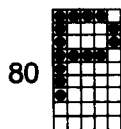
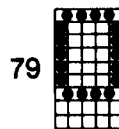
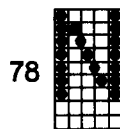
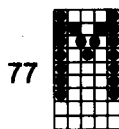
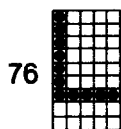
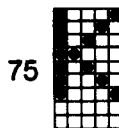
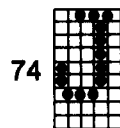
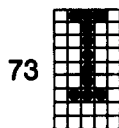
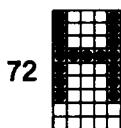
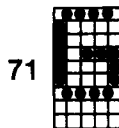
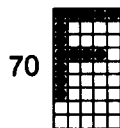
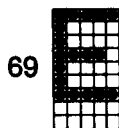
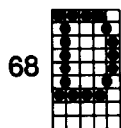
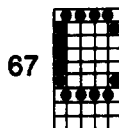
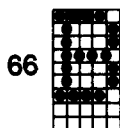
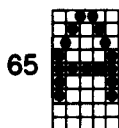
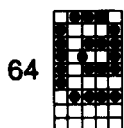
Data Code List of International Character Set

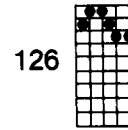
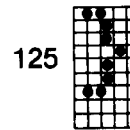
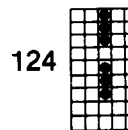
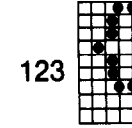
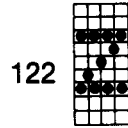
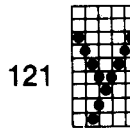
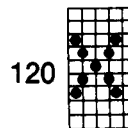
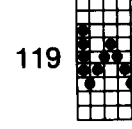
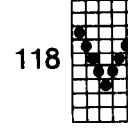
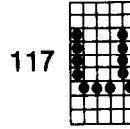
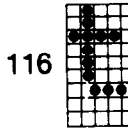
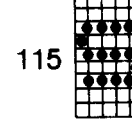
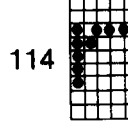
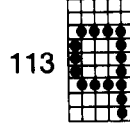
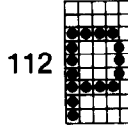
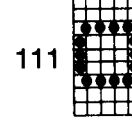
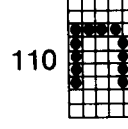
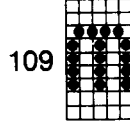
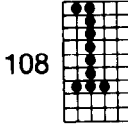
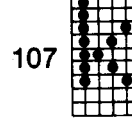
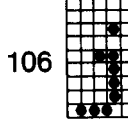
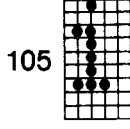
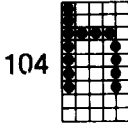
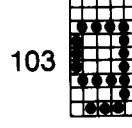
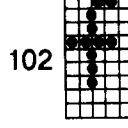
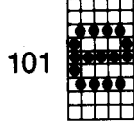
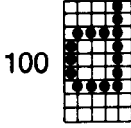
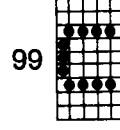
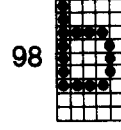
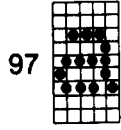
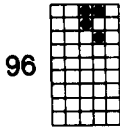
DIP Switch No.			Control Code	Country	HEX	23	40	5B	5C	5D	5E	60	7B	7C	7D	7E
1-6	1-7	1-8			DEC	35	64	91	92	93	94	96	123	124	125	126
ON	ON	ON	ESC 7 0	U.S.A.	#	@	[\]	^	'		:		~	
OFF	ON	ON	ESC 7 1	England	£	@	[\]	^	'		:		~	
ON	OFF	ON	ESC 7 2	Germany	#	§	Ä	Ö	Ü	^	'	ä	ö	ü	β	
OFF	OFF	ON	ESC 7 3	Denmark	#	@	Æ	Φ	Å	^	'	æ	φ	å	~	
ON	ON	OFF	ESC 7 4	France	£	à	°	ç	§	^	'	é	ù	è	¨	
OFF	ON	OFF	ESC 7 5	Sweden	#	É	Ä	Ö	Å	Ü	é	a	ö	å	ü	
ON	OFF	OFF	ESC 7 6	Italy	#	§	°	ç	é	^	ù	à	ò	è	ì	
OFF	OFF	OFF	ESC 7 7	Spain	#	@	i	Ñ	¿	^	'	¨	ñ		~	

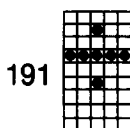
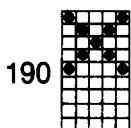
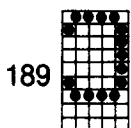
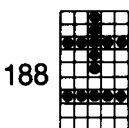
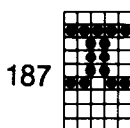
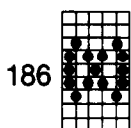
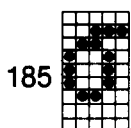
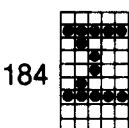
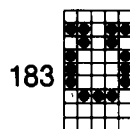
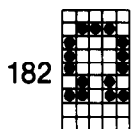
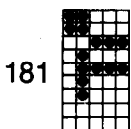
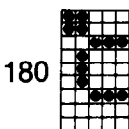
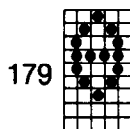
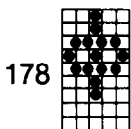
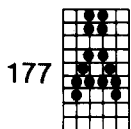
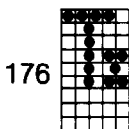
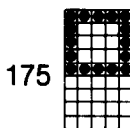
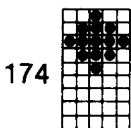
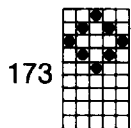
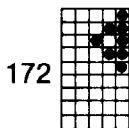
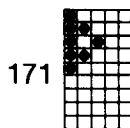
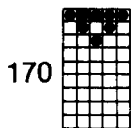
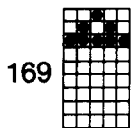
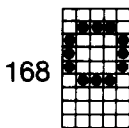
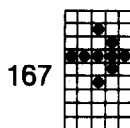
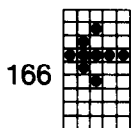
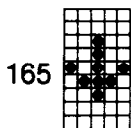
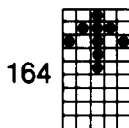
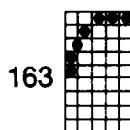
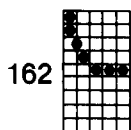
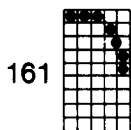
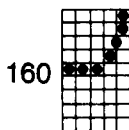


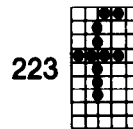
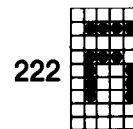
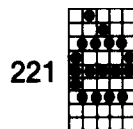
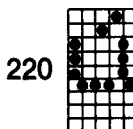
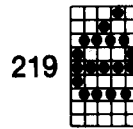
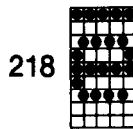
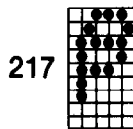
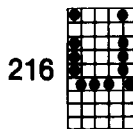
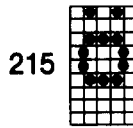
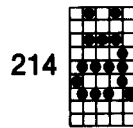
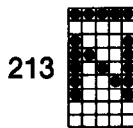
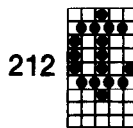
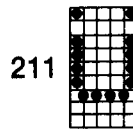
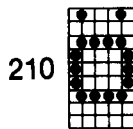
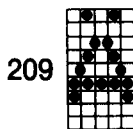
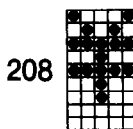
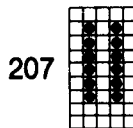
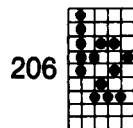
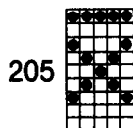
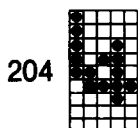
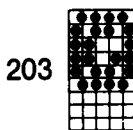
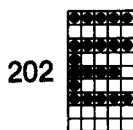
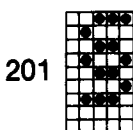
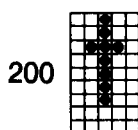
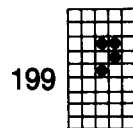
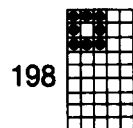
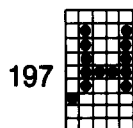
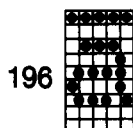
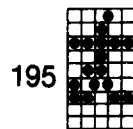
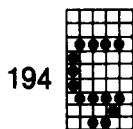
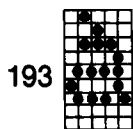
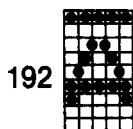
CHARACTER STYLE TABLES

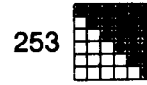
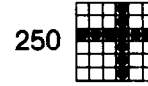
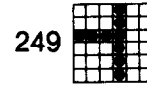
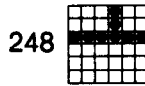
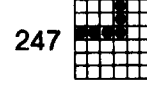
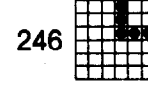
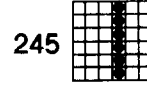
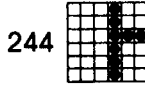
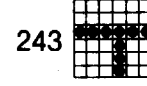
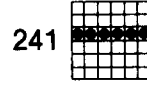
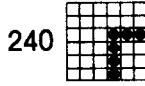
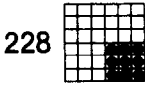
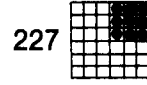
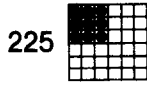
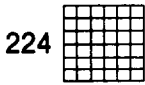
32		33		34		35	
36		37		38		39	
40		41		42		43	
44		45		46		47	
48		49		50		51	
52		53		54		55	
56		57		58		59	
60		61		62		63	

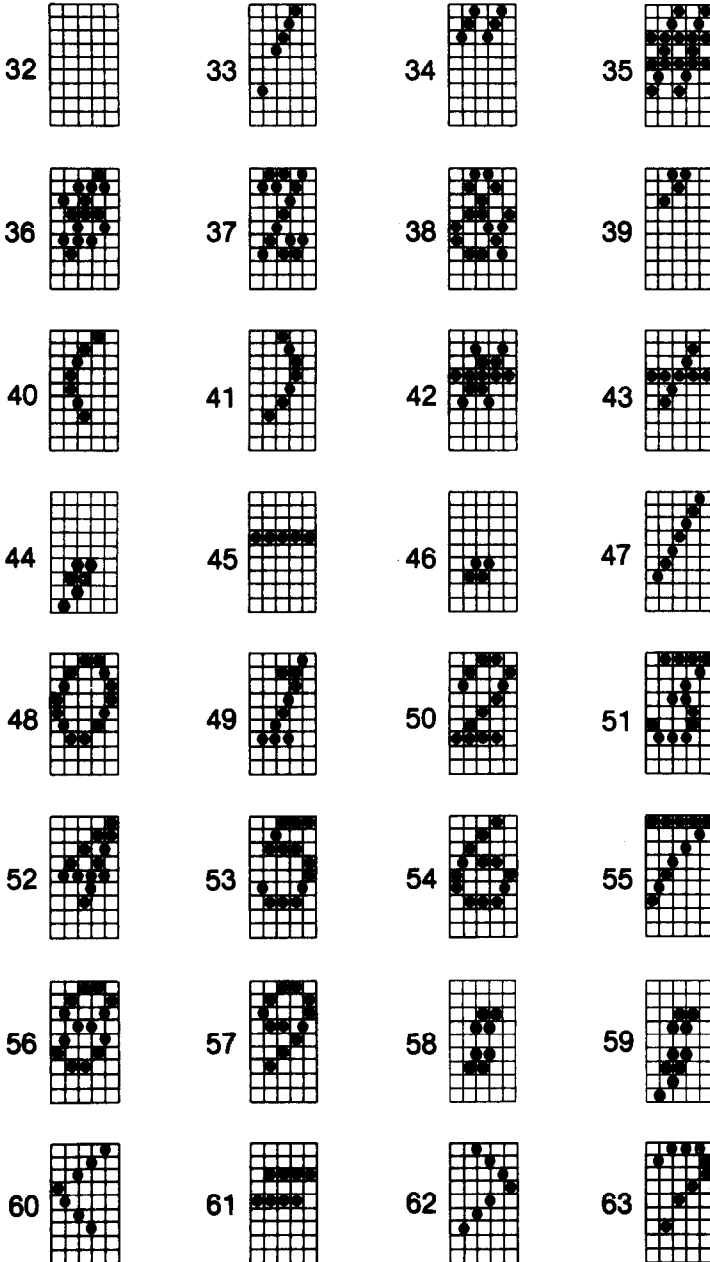


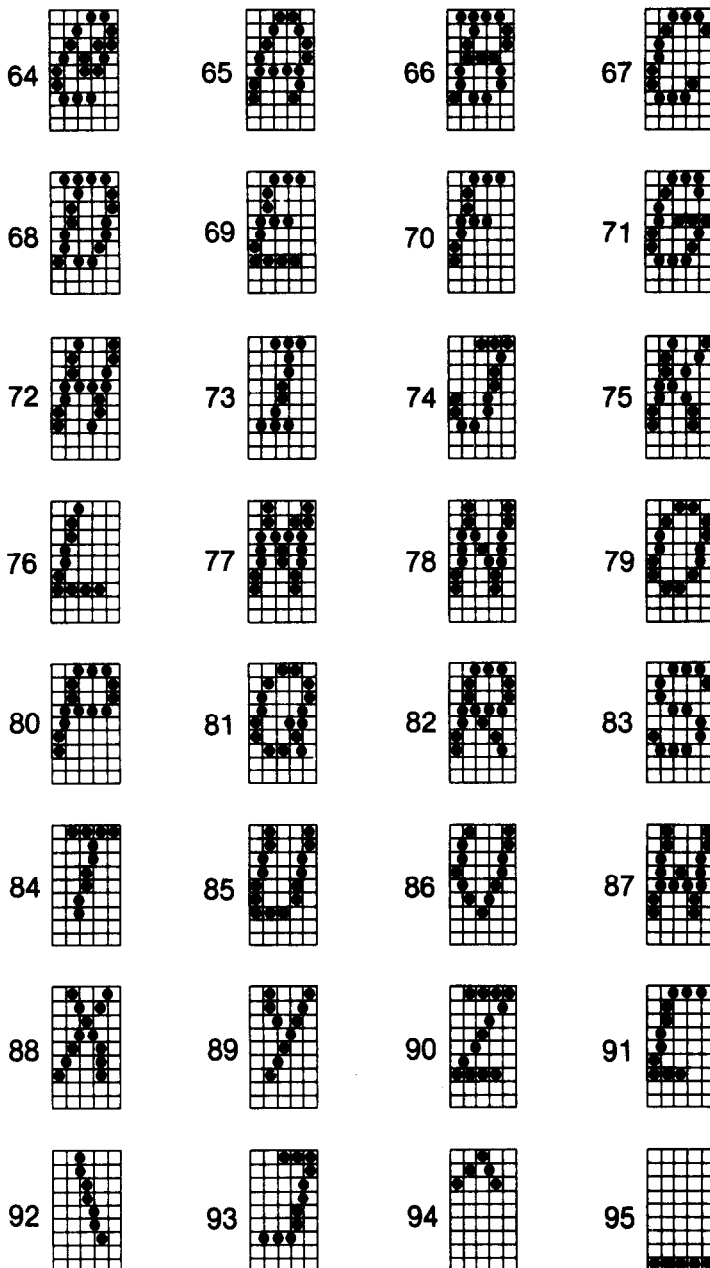


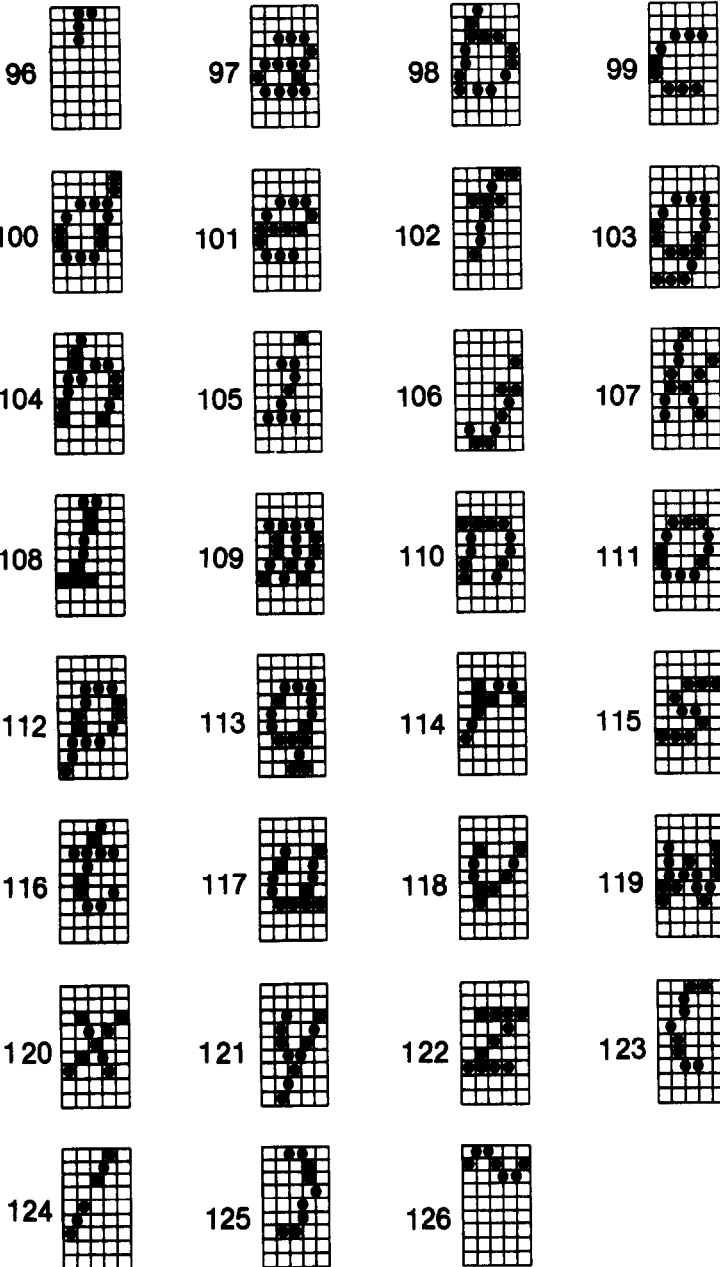


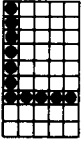




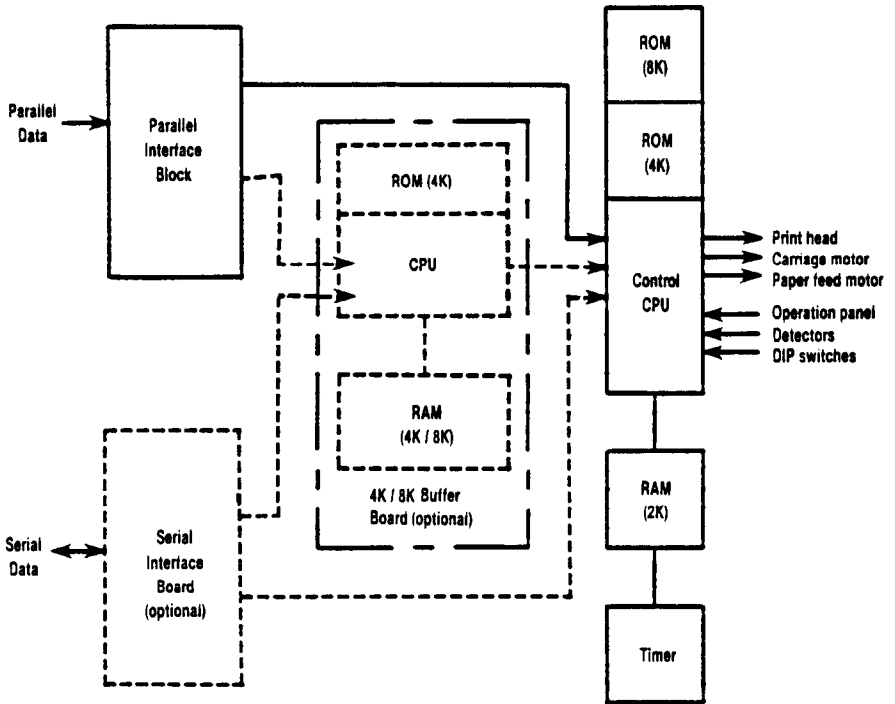


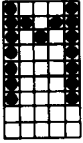






CONTROL CIRCUIT BLOCK DIAGRAM





FUNCTION CODE SUMMARY

Control Code	Decimal	Hexi-decimal	Function
NUL	0	00	Used to terminate tab setting programming of ESC-D and ESC-P.
BEL	7/135	07/87	Buzzer sounds for ¼ second when this code is input. Input of "ESC Y 0" code negates BEL code. Input of "ESC Y 1" makes BEL code effective.
BS	8/136	08/88	Buffer data is printed by this code and this code shifts the print start position to left side by one character space.
HT	9/137	09/89	Moves the print head to the next tab set position.
LF	10/138	0A/8A	One line paper feed is done by an input of LF code after buffer data is printed.
VT	11/139	0B/8B	Feed paper to the next VT position where programmed by "ESC P nn . . . n NUL." Default value is set every 6 lines.
FF	12/140	0C/8C	Paper is fed to the print start line on next page by preset program (header line) after buffer data is printed.
CR	13/141	0D/8D	CR code causes printout of buffer, and if DIP switch 2-4 (auto line feed) is in "on" position, CR and 1 line is fed automatically.
SO, ESC SO	14/142	0E/8E	Turns on double-width character print mode. This mode is cancelled by using DC4 or CR.
SI, ESC S1	15/143	0F/8F	Turns on compressed mode (17 CPI). This mode cancelled by using DC2.
DC1	17/145	11/91	This code causes printer to be selected.
DC2	18/146	12/92	Cancel SI-mode.
DC3	19/147	13/93	This code causes printer to be deselected.
DC4	20/148	14/94	Cancel SO-mode.
RS	30/158	1F/9E	This code ends the macro instruction.
DEL	127	7F	Delete last character input to buffer.

Function Code Summary



Control Code	Decimal	Hexi- decimal	Function
ESC	27/155	1B/9B	This code used for expansion control codes.
ESC #	35	23	Accepts 8th bit "as is" from host CPU.
ESC * 0	42 0	2A, 30	Copies fonts in character ROM into download RAM
ESC * 1 n1 n2 m1 m2	42 1 n1 n2 m1 m2	2A, 31 n1 n2 m1 m2	This code defines the download character into RAM
ESC \$ 0	36 0	24,30	Cancels the download character set.
ESC \$ 1	36 1	24,31	Selects the download character set
ESC-1	45 1	2D 01	All data after this code is printed with underlining.
ESC-0	45 0	2D 00	Cancels "ESC-1" underline mode.
ESC 0	48	30	This code changes line feed length to 1/8".
ESC 1	49	31	This code changes line feed length to 7/72".
ESC 2	50	32	This code changes line feed length to 1/6".
ESC 3 n	51 n	33 n	This code changes line feed length to n/144". The value of n can be from 1 to 127.
ESC 4	52	34	This code selects character ROM set of italic characters.
ESC 5	53	35	This code cancels ESC 4.
ESC 7 n	55 n	37 n	This code selects the international character set.
ESC 8	56	38	This code enables printing even though paper "out" sensor switch is "on!"
ESC 9	57	39	Cancels "ESC 8" mode.
ESC =	61	3D	This code sets 8th bit to logic-0.
ESC >	62	3E	This code sets 8th bit to logic-1.
ESC @	64	40	Software "Reset" command, re-initialize all parameters to the power "on" condition.
ESC !	33	21	This code selects the macro instruction.
ESC a n	97 n	61 n	This code sends a one-time-only vertical feed of n lines.
ESC b n	98 n	62 n	This code sends a one-time-only horizontal tab of n columns.
ESC A n	65 n	41 n	This code changes line feed length to n/72". Value of n can be from 1 to 127.

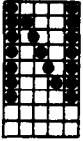


Control Code	Decimal	Hexi-decimal	Function
ESC B 1	66 1	42 01	Sets 10 CPI.
ESC B 2	66 2	42 02	Sets 12 CPI.
ESC B 3	66 3	42 03	Sets 17 CPI.
ESC C n	67 n	43 n	Sets form length to n lines. Value of n can be from 1 to 127.
ESC C 0 n	67 0 n	43 30 n	Sets form length to n inches. Value of n can be from 1 to 32.
ESC D nn...n NUL	68 n n 0	44 n n 00	Clears current tab set and sets new tabs. Value of n is acceptable 1 to 255. (In case of 7-bit interface, 1 to 127.)
ESC E	69	45	Selects emphasized print mode.
ESC F	70	46	Cancel's emphasized print mode.
ESC G	71	47	Selects double strike print mode.
ESC H	72	48	Cancel's double strike print mode.
ESC J n	74 n	4A n	One-time line feed of $n/144$ "
ESC K n1 n2	75 n1 n2	4B n1 n2	480 dots pitch of bit-image-column-scan mode is selected. Number of columns are determined $n1 + n2 \times 256$.
ESC L n1 n2	76 n1 n2	4C n1 n2	960 dots pitch of bit-image-column-scan mode is selected. Number of columns are determined $n1 + n2 \times 256$.
ESC M n	77 n	4D n	Sets LH-margin. Value of n can be up to maximum number of CPI. But in case of a 7-bit interface, up to 127.
ESC N n	78 n	4E n	Sets skip-over perforation to n lines.
ESC O	79	4F	Resets skip-over perforation to zero lines.
ESC P nn...n NUL	80 n n 0	50 n n 00	Sets VT positions (vertical tab); default is every 6 lines.
ESC Q n	81 n	51 n	Sets RH-margin. Value of n can be up to maximum number of CPI. But in case of a 7-bit interface, up to 127.
ESC R n	82 n	52 n	Sets header line position. Value of n can be from 1 to 16.
ESC + ...RS	43, 30	2 B, 1 E	Defines the macro-instruction
ESC S 0	83 0	53 00	Superscript mode is selected.
ESC S 1	83 1	53 01	Subscript mode is selected.
ESC T	84	54	Cancel's "ESC S" and "ESC U".
ESC U 1	85 1	55 01	Selects uni-directional print mode.
ESC U 0	85 0	55 00	Cancel's uni-directional print mode.

Function Code Summary



Control Code	Decimal	Hexi- decimal	Function
ESC W 1	87 1	57 01	Double-wide pitch print mode is selected.
ESC W 0	87 0	57 00	Cancels "ESC W 1" mode.
ESC Y 1	89 1	59 01	Enables buzzer.
ESC Y 0	89 0	59 00	Disables buzzer.
ESC y n1 n2 m1 m2	121 n1 n2 m1 m2	79 n1 n2 m1 m2	Prints dual-density graphics with double-speed.
ESC z n1 n2 m1 m2	122 n1 n2 m1 m2	7A n1 n2 m1 m2	Prints quadruple-density graphics.



GEMINI'S PARALLEL INTERFACE

Parallel Interface Specifications

The Gemini-10X and Gemini-15X printers have the following operational specifications:

Parallel Interface Specs

Data transfer rate:	1,000 to 6,000 CPS
Synchronization system:	Via externally supplied STROBE pulses
Handshake:	$\overline{\text{ACK}}$ and BUSY signals
Logic level:	Compatible with TTL level

The Connector Signals and Their Functional Descriptions

To understand the "mating" of Gemini with your computer, let's consider the function of the pins and the signals they convey. Figure N-1 provides a summary of the signals.

Pin 1 carries the STROBE pulse signal, which is normally kept at a high level (after the signal goes low, the data is read). Pins 2 through 9 carry the data, in 7-bit or 8-bit bytes. When the data is "1," the signal level is high, and the level goes down when the data is "0."

Pins 10 and 11 carry the "handshake" signals: $\overline{\text{ACK}}$ and BUSY. $\overline{\text{ACK}}$ is a pulse signal with a pulse-width of approximately 9 microseconds, which is generated when the data is received. When this pulse signal has been completed, the next data acceptance is permitted. BUSY is a signal at DC level that indicates the operating condition of the printer. When this signal goes low, the next data acceptance is permitted.

Pin 12 is the PAPER END signal connector. This DC signal is normally low, and goes high when there is a "no-paper" condition. (Note: DIP switch number 2-1 also controls the paper-empty status.)

Pin 13 connects another output DC signal, this one for indicating when the printer is online.



Pin no.	Signal name	I/O	Function
1	<u>STROBE</u>	Input	Reads in the data
2-9	DATA 1-8	Input	Communicates bytes of data
10	<u>ACK</u>	Output	Acknowledges receipt of data
11	BUSY	Output	Indicates receptiveness to more data
12	PAPER END	Output	Notifies that you're out of paper
13	SELECTED	Output	Online mode indicator
14	N/C	—	(unused)
15	N/C	—	(unused)
16	SIGNAL GND	—	Signal ground
17	CHASSIS GND	—	Frame ground
18	+ 5V DC	Output	External +5VDC Power (max. 50ma)
19-30	Twisted-pair returns for Pins 1-12	—	Return signals, conveyed by twisted-pair cable
31	<u>INPUT PRIME</u>	Input	Resets printer: clears buffer and initializes
32	<u>ERROR</u>	Output	Indicates offline mode
33	EXT GND	—	External ground
34-35	N/C	—	(unused)
36	N/C	—	(TTL high level)

Figure N-1. The parallel interface signals and their names, pins, I/O classifications, and functional summary.



The groundings are provided by three pins. Pin 16 is the signal ground, 17 is the chassis ground, and pin 33 is the external ground. Pin 18 carries 5 volts of direct current.

Twisted-pair return signals for pins 1 through 12 are conveyed by pins 19 through 30, respectively.

Pin 31, INPUT PRIME, carries the signal for resetting the control logic. When this signal goes low, the printer is initialized and the memory buffer is cleared. Another output signal, ERROR, is carried by pin 32. When Gemini is off-line or disabled by an abnormal state, the ERROR signal goes low.

Pins 14, 15, 34, 35, and 36 are unused.

Signal Name	Circuit Example
Input	
DATA 1-8	
$\overline{\text{STROBE}}$	
Output	
BUSY, $\overline{\text{ACK}}$	

Figure N-2. Sample circuit for parallel interface.

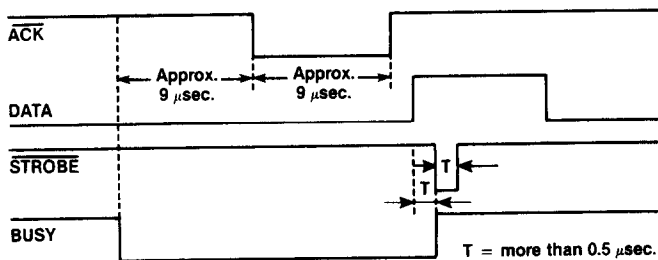
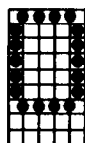


Figure N-3. Data-transfer timing chart for parallel interface.



Notes



Code Chart

This chart shows the code numbers in decimal, hexadecimal, and binary form.

Decimal	Binary	Hexa- decimal	Decimal	Binary	Hexa- decimal
0	00000000	00	36	00100100	24
1	00000001	01	37	00100101	25
2	00000010	02	38	00100110	26
3	00000011	03	39	00100111	27
4	00000100	04	40	00101000	28
5	00000101	05	41	00101001	29
6	00000110	06	42	00101010	2A
7	00000111	07	43	00101011	2B
8	00001000	08	44	00101100	2C
9	00001001	09	45	00101101	2D
10	00001010	0A	46	00101110	2E
11	00001011	0B	47	00101111	2F
12	00001100	0C	48	00110000	30
13	00001101	0D	49	00110001	31
14	00001110	0E	50	00110010	32
15	00001111	0F	51	00110011	33
16	00010000	10	52	00110100	34
17	00010001	11	53	00110101	35
18	00010010	12	54	00110110	36
19	00010011	13	55	00110111	37
20	00010100	14	56	00111000	38
21	00010101	15	57	00111001	39
22	00010110	16	58	00111010	3A
23	00010111	17	59	00111011	3B
24	00011000	18	60	00111100	3C
25	00011001	19	61	00111101	3D
26	00011010	1A	62	00111110	3E
27	00011011	1B	63	00111111	3F
28	00011100	1C	64	01000000	40
29	00011101	1D	65	01000001	41
30	00011110	1E	66	01000010	42
31	00011111	1F	67	01000011	43
32	00100000	20	68	01000100	44
33	00100001	21	69	01000101	45
34	00100010	22	70	01000110	46
35	00100011	23	71	01000111	47

Code Chart



Decimal	Binary	Hexa- decimal	Decimal	Binary	Hexa- decimal
72	01001000	48	116	01110100	74
73	01001001	49	117	01110101	75
74	01001010	4A	118	01110110	76
75	01001011	4B	119	01110111	77
76	01001100	4C	120	01111000	78
77	01001101	4D	121	01111001	79
78	01001110	4E	122	01111010	7A
79	01001111	4F	123	01111011	7B
80	01010000	50	124	01111100	7C
81	01010001	51	125	01111101	7D
82	01010010	52	126	01111110	7E
83	01010011	53	127	01111111	7F
84	01010100	54	128	10000000	80
85	01010101	55	129	10000001	81
86	01010110	56	130	10000010	82
87	01010111	57	131	10000011	83
88	01011000	58	132	10000100	84
89	01011001	59	133	10000101	85
90	01011010	5A	134	10000110	86
91	01011011	5B	135	10000111	87
92	01011100	5C	136	10001000	88
93	01011101	5D	137	10001001	89
94	01011110	5E	138	10001010	8A
95	01011111	5F	139	10001011	8B
96	01100000	60	140	10001100	8C
97	01100001	61	141	10001101	8D
98	01100010	62	142	10001110	8E
99	01100011	63	143	10001111	8F
100	01100100	64	144	10010000	90
101	01100101	65	145	10010001	91
102	01100110	66	146	10010010	92
103	01100111	67	147	10010011	93
104	01101000	68	148	10010100	94
105	01101001	69	149	10010101	95
106	01101010	6A	150	10010110	96
107	01101011	6B	151	10010111	97
108	01101100	6C	152	10011000	98
109	01101101	6D	153	10011001	99
110	01101110	6E	154	10011010	9A
111	01101111	6F	155	10011011	9B
112	01110000	70	156	10011100	9C
113	01110001	71	157	10011101	9D
114	01110010	72	158	10011110	9E
115	01110011	73	159	10011111	9F



Decimal	Binary	Hexa- decimal	Decimal	Binary	Hexa- decimal
160	10100000	A0	184	10111000	B8
161	10100001	A1	185	10111001	B9
162	10100010	A2	186	10111010	BA
163	10100011	A3	187	10111011	BB
164	10100100	A4	188	10111100	BC
165	10100101	A5	189	10111101	BD
166	10100110	A6	190	10111110	BE
167	10100111	A7	191	10111111	BF
168	10101000	A8	192	11000000	C0
169	10101001	A9	193	11000001	C1
170	10101010	AA	194	11000010	C2
171	10101011	AB	195	11000011	C3
172	10101100	AC	196	11000100	C4
173	10101101	AD	197	11000101	C5
174	10101110	AE	198	11000110	C6
175	10101111	AF	199	11000111	C7
176	10110000	B0	200	11001000	C8
177	10110001	B1	201	11001001	C9
178	10110010	B2	202	11001010	CA
179	10110011	B3	203	11001011	CB
180	10110100	B4	204	11001100	CC
181	10110101	B5	205	11001101	CD
182	10110110	B6	206	11001110	CE
183	10110111	B7	207	11001111	CF



Decimal	Binary	Hexa- decimal	Decimal	Binary	Hexa- decimal
208	11010000	D0	232	11101000	E8
209	11010001	D1	233	11101001	E9
210	11010010	D2	234	11101010	EA
211	11010011	D3	235	11101011	EB
212	11010100	D4	236	11101100	EC
213	11010101	D5	237	11101101	ED
214	11010110	D6	238	11101110	EE
215	11010111	D7	239	11101111	EF
216	11011000	D8	240	11110000	F0
217	11011001	D9	241	11110001	F1
218	11011010	DA	242	11110010	F2
219	11011011	DB	243	11110011	F3
220	11011100	DC	244	11110100	F4
221	11011101	DD	245	11110101	F5
222	11011110	DE	246	11110110	F6
223	11011111	DF	247	11110111	F7
224	11100000	E0	248	11111000	F8
225	11100001	E1	249	11111001	F9
226	11100010	E2	250	11111010	FA
227	11100011	E3	251	11111011	FB
228	11100100	E4	252	11111100	FC
229	11100101	E5	253	11111101	FD
230	11100110	E6	254	11111110	FE
231	11100111	E7	255	11111111	FF

GLOSSARY

Alternating current (AC) is the type of electricity used in household wiring. It's called "alternating" because the electrical flow rapidly alternates in direction.

Alphanumeric is a character that is a letter of the *alphabet*, a *numerical* symbol (a numeral), or any other symbol in a set of characters.

American Standard Code for Informational Interchange (ASCII) is a widely accepted system for using the numerals 0 through 127 to represent alphanumeric and symbolic characters.

Beginner's All-purpose Symbolic Instruction Code (BASIC) is a computer language found on personal computers and used (with some variations) for most small-computer programs.

Binary is a term that, in a general sense, pertains to a choice or condition in which there are two possibilities. Specifically, it is used relative to computers to refer to the binary mathematical numeration system involving only two digits: one and zero. Computers code data in binary.

Bit is the building block of computer memory that represents one of two values: zero or one, yes or no, on or off. A bit is a binary digit (0 or 1) that is grouped into a 7-unit or 8-unit entity called a byte, which conveys data in a computer system.

Byte is the standard unit for measuring computer memory that consists of a grouping of adjacent binary digits (such as 00000111 to represent the quantity "7"), which are manipulated as a unit by the computer. A byte can represent one character.

Buffer is a temporary storage area that is used to balance the difference between the operating speed of your computer and Gemini. The standard buffer capacity of Gemini is 816 bytes, with an optional capability of over 8,000 more bytes.

Capacitor is an electrical device that usually consists of layers of conducting and nonconducting materials, used to store energy and to regulate electrical charges.

Cathode ray tube (CRT) is the display screen on your microcomputer.

Central processing unit (CPU) is the information-processing "traffic cop" of the printer. The **CPU** receives data from the interface circuit (from your computer) and transfers that data to the print-head circuitry.

Character is the word for a representative symbol (such as the letter "A" or the numeral "5" or the "\$" sign), which conveys all or part of an idea. The Gemini printers offer 96 standard ASCII characters, 96 italic characters, 64 special characters, 32 block graphic characters, 96 downloadable characters as well as 88 foreign characters.

Glossary

Characters per second (CPS) is a measurement of printing speed, like miles per hour for an automobile. The Gemini printers have a print speed of 120 CPS at 10 **characters per inch (CPI)**.

Chip is a small integrated circuit that is made from multiple layers of silicon and other materials, which conducts electricity in a manner that renders the device capable of handling computer-logic tasks.

Computer is a device that can automatically process and execute a sequence of mathematical or logical operations.

Computer language is the set of software instructions understood by the computer that allows the user to communicate human ideas through electronic devices.

Data is a collection of factual information, converted to a form (coding) that allows it to be stored, transmitted, and manipulated by computers and their peripherals.

Diode is a semiconductor, such as the **light-emitting diode (LED)** that is each lamp on the printer's control panel.

Direct current (DC) is electrical current that flows in one direction ("directly"), such as with devices powered by flashlight batteries. The Gemini power unit converts the input AC power into voltages of **+5V DC** (which is used as a power supply for the CPU, ROM, and RAM), **+8V DC** (used to hold the positions of the motors and as a power supply to reset the timing circuit), **+24V DC** (used to drive the solenoids of the print head, and the motors).

Down-Loading (Down-Loadable Character Sets) is the ability to transfer programmed character codes from a computer to a receptive peripheral device (Gemini-10X/15X) that contains storage RAM for such "down-loading" of specially created symbols.

Dual In-line Package (DIP) switch is a set of "microswitches" that allows you to quickly convert your machine to perform important functions.

Dump is a term for copying a storage file from the computer onto a line printer or other peripheral device.

Escape character is the signal sent from your computer to Gemini, via your pressing the **ESC** key on the keyboard, that initiates one of the **escape-function codes** that control some of the printer's key operating parameters. In BASIC, sending **CHR\$(27)**, which is the code for ESC, initiates one of the ESC function control commands.

File is a collection of related records that are treated as a single set of data for purposes of storage, retrieval, and transmission.

Form Feed (FF) is a paper-advancing function that is initiated by pressing the FF button on the control panel while the printer is in the "local" mode. FF advances the paper around the platen and stops at the first line of the next page. FF may also be activated through software control.

Glossary

Fuse is an electrical safety device consisting of a metal wire (within a glass tube) that melts and interrupts the circuit when the amperage exceeds a predetermined point.

Hardware is the set of tangible parts of a computer system: the computer, printer, and other peripherals — everything in a computer system except the information contained in that system.

Input/Output (I/O) is the exchange of data between the computer and the printer. For example, your Gemini takes *input* from the computer in the form of data; when it cannot accept any more for the time being, Gemini sends *output* in the form of a "BUSY" signal, telling the computer to wait.

Interface is the common boundary (physical connection) between two hardware items (such as your computer and Gemini) or between two systems. The interface permits the transfer of data and intelligent communication.

Hexadecimal is the mathematical numeration system with a base of 16 (compared with 2 in the binary system and 10 in the decimal system). Appendix I presents the hexadecimal and decimal equivalents for the ASCII coded characters.

K stands for **kilobyte**, specifically 1,024 bytes. Computer memory is often expressed in terms of K.

LED, see Diode.

Line Feed (LF) is a paper-advancing function that is initiated by pressing the LF button on the control panel of your Gemini. When this button is depressed, the paper advances one line; when this switch is held down, consecutive paper-feeds result for as long as the button is depressed.

Local defines a condition in which the communication link between the printer and the computer has been interrupted and the printer is not on-line. The **on-line control panel button** is used to switch the printer between being local and on-line with the computer.

Macro Instruction is the ability of a peripheral device (Gemini-10X/15X) to accept a single control code to do the work of a multi-command series of codes.

Main memory refers to the electronic circuits that are instantaneously available to the computer, circuits containing storage of the software and data currently being used.

Microprocessors are the "brains" of the personal computer. The most commonly used types, each about the size of a stack of three dimes, are the 8080, Z-80, and the 6502.

Megabyte is equivalent to 1,024 kilobytes of memory. *Note:* Because computers work with binary numbers, K equals 2^{10} (or 1,024) bytes, rather than a thousand bytes as "K" would imply; similarly, a megabyte equals 2^{10} K.

Glossary

Off-line is a condition in which your printer can relate to itself but is unable to communicate with your computer.

On-line is a condition in which the printer is interfaced with the computer and is ready to receive data transmission.

Parallel transmission is a type of interface in which each byte of data is transmitted from the computer to the printer, arranged so that each bit is sent on a separate wire, side-by-side (in "parallel").

Peripheral is any device used by a computer to communicate, including printers, monitors, and disk drives.

Random Access Memory (RAM) is an array of semiconductor switches that stores information in the form of bits. RAM storage changes as the printer performs its tasks. The RAM is the main memory that the user can alter and might be called the "Read/Write Memory." In the Gemini printers, the RAM consists of 816 bytes of storage for data transferred from the interface circuit to the print head circuit.

Read is to obtain data from a computer storage media or from user input.

Read-Only Memory (ROM) is a type of main memory that comes from the manufacturer with data or programs that cannot be modified by the user. Unlike most RAMs, the contents of a ROM are not lost by turning the power "off."

Resolution is the degree of detail that the user can obtain with different printing modes. The Gemini printers offer 60 x 72 low-resolution, 120 x 144 high-resolution and 240 x 144 ultra high resolution bit image graphics.

Serial transmission is a type of communication in which each byte of data is transmitted from the computer to the printer, arranged so that each bit is sent on one wire, end-to-end ("serially").

Software is the set of tasks or programs that makes hardware perform particular functions.

Subscripts are those characters that are printed beneath the baseline for standard-size characters. The "2" is a subscript in H₂O, the chemical notation for water.

Superscripts are characters above the baseline of characters. The "2" is a superscript in $E = mc^2$, Einstein's simple equation for the relationship of energy to mass.

Throughput is the total printing time, including data acquisition and line feed time.

Write is to copy, generally from internal to external storage.

Index

- alphanumeric 71-72
- Apple computers 151-164
- ASCII code 58, 82, 226, 230
- Atari computers 165-178

- “back-pass” loading 28-30
- backspacing 135
- BASIC 59-60
- bidirectional printing 128
- binary 59, 93
- bit image column scan
(see dot graphics)
- bit-selection
(see DIP switches)
- block graphics 84, 104-108
- “bottom-pass” 28-31
- buffer 7-8
- buzzer control 139-140

- casing removal 143-145
- carriage return (CR) 68, 133
- character strings, CHR\$ 59-60
- character styles 232-240
- checklist for a new Gemini 217
- circuit for parallel interface 248
- clamp lever 23, 26
- cleaning Gemini 143
- column settings 84-85, 91-92
- Commodore computers 179-191
- condensed type 62-63, 66, 125
- connecting cables 53-54
- consumer response 261
- control board 6-8
- control codes 123-140, 242-245
 - back cover
- control circuit block
 - diagram 251
- control panel 1-2, 5-6
- cover 217-219
- CPU 47-48

- decimal codes 59, 226-230, 250-253
- DIP switches 1-2, 49-53
- dot graphics 90-97, 108, 136
- double-strike mode 64, 67, 73, 85
- downloadable characters 115-121,
137-138

- elite printing (12 CPI)
65, 67, 85, 125
- emphasized mode 66-67, 126-127
- enlarged mode (5, 6, & 8.5 CPI)
66-67, 125
- escape codes 123-140
- exploded view 6-7

- fixing screws 231
- font style control 123-124
- font pitch control 124-126
- form feed (F.F.) 5-6, 130-132
- French characters 99-100
- function codes 58, 123-140
- fuse replacement 144-145

- German characters 100
- glossary 254-257
- grounding 3, 248

- hexadecimal 59
- horizontal print-head
 - position control 133-135

IBM-PC 192-193
 italic characters 45, 66, 124
 impact printing 4-5
 initialization 55
 interfacing 2, 47-48, 54
 international character sets 51,
 124, 231

 line feed (L.F.) 5-6, 71, 129-130
 line spacing 70-71
 logic-seeking 6

 macro commands 120-121, 139
 maintenance 143-149
 margin codes 134
 mathematical and scientific
 characters 101-102
 matrix printing 45, 72, 83-85
 monetary characters 101

 normal printing (10 CPI)
 64-65, 67

 online button 5-6, 62
 on/off power switch 1-2
 operating environment 9
 Osborne computers 194-205

 paper-out indicator 5
 paper separator & guide
 4, 217-219
 paper loading 13-31
 parallel interface
 2, 48, 53-54, 246-249
 pin-out charts
 163, 177, 193, 195, 207
 pins 246-248
 platen knob 1-2
 plotting characters 102-104
 power unit 6-7
 print commands (see
 BASIC and function codes)
 print head 3-4, 37, 81-83, 146-147
 programs in BASIC
 63, 76-77, 89, 97, 105, 107, 111-112,
 117-119, 121

RAM 241
 ready lamp 5
 release lever 16-17
 reset function
 (see initialization)
 ribbon loading 32-37
 roll paper holder & shaft
 4, 217-219
ROM 241

 self-test 39-45
 serial interface 2, 55
 snap levers 14-15
 Spanish characters 100
 specifications for Gemini
 10-11
STROBE pulse 246-249
 subscripts 66, 73, 85, 128
 superscripts
 66, 73, 85, 127

 timing chart 258
 tractor unit 4-5, 14-16, 22-23
TRS-80 206-216
 type styles 64-65, 71-74

 underlining 84, 127
 uni-directional printing 128

 vertical spacing 85-90
 vertical tabs 132

 warnings for print head 17, 146
 word-processing software 74-78

DIP SWITCH GUIDE

Dip Switch 1 Pin No.	Function	When "ON"	When "OFF"	Factory Setting
1-1	Selection of Character Set	Contingent on pin no. 4		ON
		(See Figure 4-5)		
1-2	Print Mode	Normal/Emphasized		ON
1-3	Set of Print Pitch	10 CPI/17 CPI		ON
1-4	Selection of Character Set	Contingent on pin no. 1		ON
		(See Figure 4-5)		
1-5	Line Feed Value	$\frac{1}{8}$ inch / $\frac{1}{4}$ Inch		ON
1-6	Selection of International	Combinations of		ON
1-7	Character Set and	switch settings		ON
1-8	Form Length	produce varied output		ON
		(See Figure 4-6)		

Dip Switch 2 Pin No.	Function	When "ON"	When "OFF"	Factory Setting
2-1	No-paper detection	Signals paper-out/No signal		ON
2-2	Buffer full	Prints at full buffer	Prints at each CR code	OFF
2-3	Bit selection	7-bit interface/8-bit interface		OFF
2-4	Automatic Line Feed	Automatic feed/No feed		OFF

DIP SWITCH STATUS CHART

	ON	OFF
DIP SWITCH 1:		
Pin 1:	_____	_____
Pin 2:	_____	_____
Pin 3:	_____	_____
Pin 4:	_____	_____
Pin 5:	_____	_____
Pin 6:	_____	_____
Pin 7:	_____	_____
Pin 8:	_____	_____
DIP SWITCH 2:		
Pin 1:	_____	_____
Pin 2:	_____	_____
Pin 3:	_____	_____
Pin 4:	_____	_____

You may wish to use this chart to keep track of the DIP Switch settings of the Gemini. This may come in handy as a good reference if things are not going as planned.