Xetec

Serial Printer Interface

FOR THE VIC 20 AND C-64
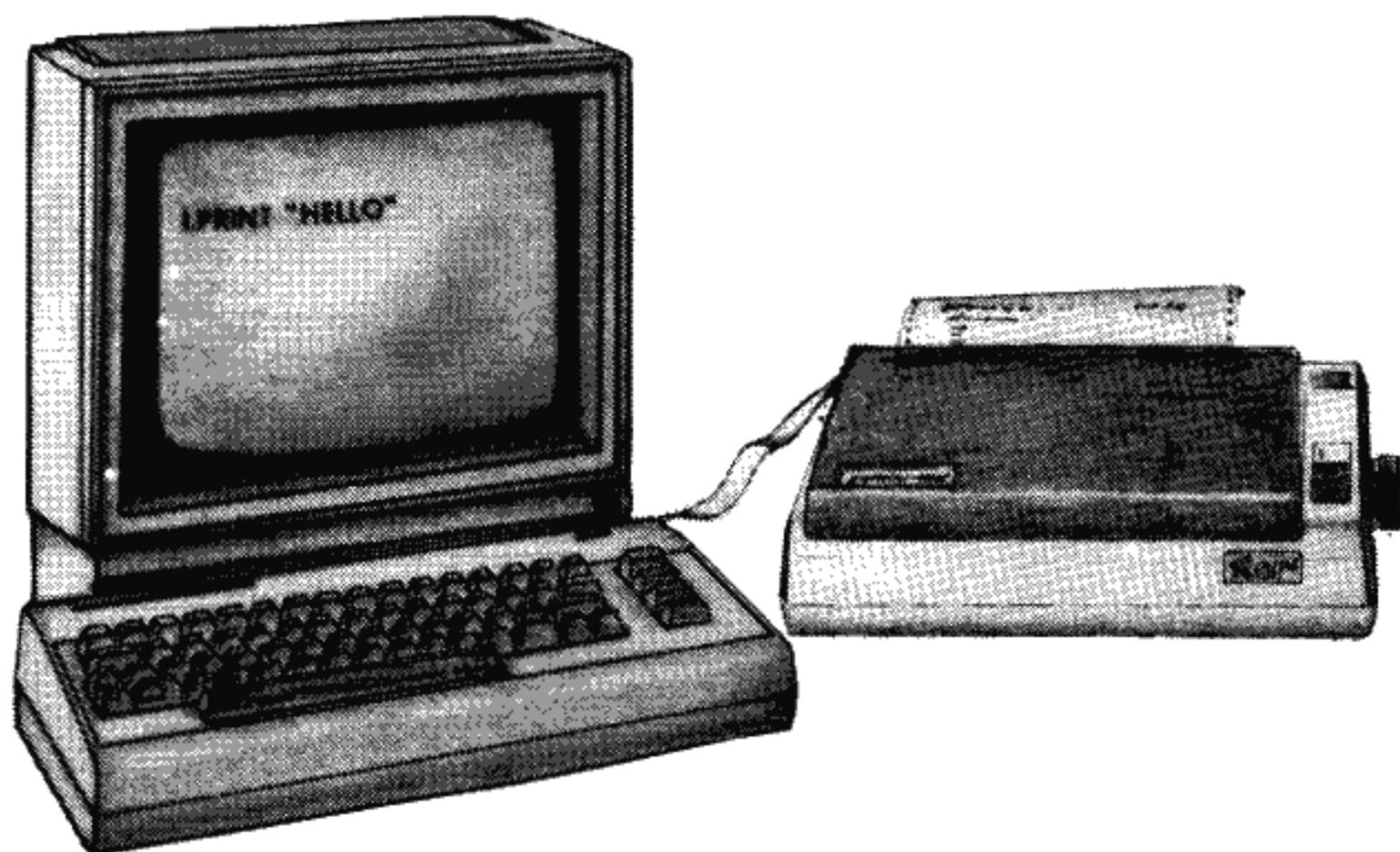
INSTRUCTION MANUAL

TABLE OF CONTENTS

INTRODUCTION

Xetec's Serial Printer Interfaces (SPI and SPI+) are designed to allow the Commodore VIC 20 or C-64 to print with any parallel printer that is Centronics compatible. They are primarily intended as a substitute for the Commodore 1525 printer, and therefore can be configured to make your printer work similarly. They have special functions, however, which the 1525 cannot perform. If you plan to make use of these features, a thorough reading of this manual is suggested. Otherwise, a skim will probably get you up and running.

NOTE: The SPI+ comes equipped with a 2048 byte buffer which allows your computer to keep computing instead of waiting for your printer to finish.

If you run into any problems, call us at (913) 827-0685 Monday through Friday, 7:00 to 5:30 CST.

SETTING IT UP

1) Turn off the computer and printer.

2) Make sure interface switches 1 thru 4 are
    OFF, OFF, ON, and ON respectively.

3) Plug the round six-pin connector into the
    serial port at the back of the computer.
    (If you are using a disk drive, plug it
    into the unused port on the back of the
    drive.)

4) Plug the large connector at the end of the
    flat cable into the parallel input port on
    your printer.

5)   Plug the connector on the end of the
    single wire into the cassette port with
    the white connector on the bottom and the
    silver side of the circuit board on top
    (viewed from the back of the computer,
    the wire should be closer to the left
    side). To use the cassette drive, plug it
    onto the board you just plugged in.

6) Turn on the printer   first, then the
    computer.

7) Make it print:

       Type: OPEN 4,4
       Then: PRINT#4,"IT WORKS!"
             CLOSE 4

    If it does, you have successfully setup
    your interface. If not, refer to appendix
    A, troubleshooting.

CONFIGURING YOUR INTERFACE

The four switches visible through the hole in the upper-right corner of the interface select certain options for printing. Depending on your application, you may have to change them.

SWITCH	OFF	ON
1	Device 4	Device 5
2	Standard printer	Okidata printer
3	No auto LF's	Software selectable
4	Transparent	Special features

Switch number 1 determines the printer's device number. If set to 'off', the interface will respond as device 4. If 'on', device 5 is selected. If you are connecting only one printer to your computer, you will most likely need to select device 4. When adding a second printer, you will normally set it as device number 5. See page 5 for a further discussion of device numbers.

Switch 2 selects the type of printer you are using. If using an Okidata, turn switch 2 'on'; otherwise leave it 'off'.

Switch 3 controls the sending of automatic line feeds. In most cases, this switch needs to be 'on'. For a few particular printers, it needs to be shut 'off'. The section titled 'Auto line feeds' on page 9 will help you determine where to set it. For now, leave it 'on'.

Switch 4 is normally set to the 'on' position, which allows you to use all the special features of your interface. If set to 'off,' the interface becomes 'transparent.' For a discussion of the benefits of being transparent, see page 11. For now, leave this

one 'on' also.

NOTE: The switch settings are checked when the computer is first turned on.   Any changes you may make in the settings will therefore not affect the operation of the interface until the power is shut off then turned on again.

MAKING IT PRINT

If you plan to use the printer from your own programs, you'll need to learn Commodore's format for printing in BASIC. Type this:

```
OPEN 10,4
PRINT#10,"HELLO"
CLOSE10
```

The first line tells the computer that we want to open a file (10) to the printer, similar to the way you must call a person on the phone before you can talk.   The number 4 is the printer's   'phone number' (device   number) because it tells the computer which device you want to talk to.

Now that you have opened file 10, the second line above will send the word "HELLO" through it (and therefore to the attached printer).

The last line tells the computer that we are through sending information, so it may close file 10 (similar to hanging up the phone).

If you understand that, you know enough to start writing BASIC programs that print. If, however, you plan to use the special features of your printer or interface, a little more in-depth discussion is required.

OPEN and CLOSE

The format for the OPEN statement is:

OPEN <file>,<device>
            or
OPEN <file>,<device>,<mode>

The 'file' number can be any from 1 to 255, although only ten can be open at once. Also, file numbers 128 to 255 stick an extra line feed in for each carriage return, so your printing will probably become double-spaced for those numbers.

The device number must be the one which was selected by the interface switch (normally '4'). If you have more than one printer attached, this number will determine which printer to direct its data to.

The optional 'mode' is used to send commands to your printer interface which tell it how to handle the data that will be coming to it through the file. If 'mode' is not included, '0' will be assumed. See pages 7-8 for a listing of all the modes and their functions.

Once a certain file is opened, it cannot be opened again until closed. Notice that if you alter a BASIC program, all open files are closed. Once a file has been opened, you may talk all you want to the printer until you CLOSE (hang up the phone).

The format for the CLOSE statement is:

CLOSE <file>

where 'file' is the number of the file to be closed.

PRINT#

Once a file has been opened to the printer, data may be sent to it the same way you would print data to the screen - with the PRINT statement. The only exception is that when you PRINT to the printer, you must tell the computer which file to talk through. The format is therefore:

PRINT#<file>,

To get an idea of how to use this statement, try this program:

```
10 OPEN 1,4
20 PRINT#1,"CHARACTER SET"
30 PRINT#1
40 FOR I=32 TO 90
50 PRINT#1,I;"=";CHR$(I)
60 NEXT I
70 CLOSE1
```

A few things about the format: there must be no space between 'PRINT' and the '#', and you cannot abbreviate it with '?'. In other words, 'PRINT #1,' and '?#1,' will not work for 'PRINT#1,'.

CMD

There is another way to send data to the printer. Once a file has been properly opened to a printer,

CMD<file>

will direct all output to that file instead of the screen. Once you have done this, a normal

PRINT or LIST statement will go to the printer. To return the output to the screen, use:

PRINT#<file>

You must do this before you try to close the file or the computer will do strange things. This only applies if you have diverted the output with the CMD command. Try this example to become familiar with 'CMD.'

```
10 PRINT "CMD EXAMPLE"
20 OPEN 1,4
30 PRINT#1,"THIS IS PRINTED"
40 CMD 1:REM DIVERT TO PRINTER
50 PRINT "THIS IS PRINTED TOO"
60 PRINT#1:REM DIRECT BACK TO SCREEN
70 CLOSE 1
```

PRINTING MODES

The following is a list of the valid modes that you may specify in an OPEN statement (see page 5).

MODE   FUNCTION

0   Normal printing, upper case only
            with line feed

1   Normal printing, upper case only
            without line feed

2   Normal printing, upper case only
            daisy wheel emulation

3   Hexadecimal output

4   Transparent, with line feed

5   Transparent, without line feed

6   Transparent, daisy wheel emulation

7   Normal printing, upper/lower case
           with line feed

8   Normal printing, upper/lower case
           without line feed

9   Normal printing, upper/lower case
           daisy wheel emulation

Add 20 to Locks the interface in any of the
any above above modes.

15   Command channel

COMMAND CHANNEL

     If you open a file in mode 15 (eg. OPEN
1,4,15) you have gained access to the command
channel.   When you send data through this
channel, it gets intercepted.   Instead of
being sent to the printer, it is interpreted
as a command.   The one-digit commands are as
follows:

COMMAND   FUNCTION

   A   Convert graphics characters to their
           ASCII values. (p. 13)

   K   *Convert graphics characters to their
           keystrokes. (p. 13)

   U   Unlock the interface (undo a 20+
           lock). (p. 14)

S   *Using Standard printer. (pp. 12,18)

O   Using Okidata printer. (pp. 12,18)

N   Make no double-width conversions.
         (pp. 12, 18)

6   *6 Lines per inch in daisy wheel
         emulation (p. 14)

8   8 Lines per inch in daisy wheel
         emulation (p. 14)

R   Reset interface (and read switches)

I   Identification. Prints the model
         and revision number of the interface

* Default modes on power-up

As an example, if you are using an Okidata
printer, and you want to print eight lines-
per-inch, you might do this:

                OPEN 1,4,15
                PRINT#1,"O8"
                CLOSE 4

     Any number of the commands may be sent in
any order with no syntax to follow (the "O8"
above could have been "O 8", "8O", "8 O", or
even   "WOW! 78" - illegal characters are
ignored).

AUTO LINE FEED
     When a carriage return code is sent at
the end of a line, some printers just move the
head to the left margin (a carriage return),

while others also move down a line (do an auto line feed).

As you can see from the table on page 7, the mode you use will determine whether or not to send your own auto line feeds. This decision will usually hinge on whether your printer does it for you. To check your printer, try this program:

```
10 OPEN 4,4,1
20 PRINT#4,"IT DOES    SEND A LINE FEED"
30 PRINT#4,CHR$(13)CHR$(13)CHR$(13)CHR$(13)
40 PRINT#4,""TAB(7)"N'T"
50 PRINT#4,CHR$(10)CHR$(10)
60 CLOSE 4
```

If it told you that it doesn't send one, you're in luck. You will usually want to use the modes which send an auto line feed (because you just discovered that your printer doesn't do it for you). Even though modes 1, 5, and 8 do not send line feeds, they might be useful. With them, you can double strike words or superimpose letters because the paper is never advanced until you manually send a line feed (CHR$(10)).

If the program above indicated that your printer does send an auto line feed, check your printer manual to see if you can shut it off (on most you can). If so, do it and all the options stated in the preceding paragraph will apply. If you can't shut the line feed off (did you remember to turn the printer off and on?), then you will want to disable line feeds in your interface (you don't want both the printer and interface sending one, or everything will be doublespaced). To do this, flip switch 3 to the off position. Remember that you will have to turn your computer off and on to recognize the change.

Modes 2, 3, 6, and 9 all send auto line feeds and have no equivalent modes that do not send them. If you followed the above steps correctly, they should work properly with your printer.

TRANSPARENT MODES

Modes 4, 5, and 6 make the interface act 'transparent,' which means that as codes are received from the computer, they are relayed to the printer with no modification. The interface may be locked 'transparent' by shutting off switch 4 (ASCII correction). Transparent modes are useful with word processors if you select them to handle all ASCII conversions. Also, if you are sending special codes to your printer (for example, to change print pitch or start underlining), it is a good idea to send the codes through a transparent channel (so that the interface cannot interfere with it).

HEXADECIMAL OUTPUT

Mode 3 is the hexadecimal output mode. Used for debugging your print routines, this mode will print every code it receives as its hexadecimal value. For example, if you OPEN file 1 to the printer (in mode 3) and type PRINT#1,"HI" you will see -48-49-0D which are the hexadecimal ASCII values for 'H', 'I', and carriage return. A new line of ASCII values is started whenever a carriage return is received.

NORMAL PRINTING

Normal printing modes are the ones that you will probably use most often. They are 0, 1, 2, 7, 8, and 9. The differences between the six modes include auto line feed or not, dot-matrix or daisy wheel emulation (see page 13), and upper/lower or upper case only.

In the upper-case-only modes, the codes sent from the computer are interpreted to be text, control characters, or graphic characters. In the upper/lower case modes, however, some of the graphic characters are now interpreted as the lower case alphabet. In other words, if you want to print in upper and lower cases, you will have to sacrifice some of the graphics characters normally available.

In the normal printing modes, the interface makes some modification to the incoming data. See appendix B for a full description, but for most purposes you need only concern yourself with four special codes.

Sending CHR$(17) (or the cursor down key) will put the interface into the upper/lower case mode. CHR$(145) (or the cursor up key) puts it into the upper-case-only mode. Both of these codes are compatible with the VIC printer.

Switch 2 selects how codes concerning the double-width function are converted. If this switch is set to the 'standard printer' position, code 15 is converted to 20 and 20 to 15. This will make double width characters on most printers compatible with the VIC printer. If set to 'Okidata', codes 14 and 30 are swapped, as are 15 and 31. You can override switch 2 by sending 'S' or 'O' through the command channel. Sending 'N' will cause none of these conversions to take place.

LISTING PROGRAMS

To LIST a program to the printer all you need to do is open a file, use 'CMD n' to divert output to the printer, and type LIST. Since your interface cannot do Commodore graphics, representing certain special codes in a BASIC listing becomes a problem. For example, a clear screen command in a PRINT statement appears as an inverse heart. How can that be represented with just text? The answer is to use symbolic abbreviations. For our example, the line might be printed

10 PRINT "[CLR]"

When listing a program to the printer, all of the special codes (such as cursor keys, color keys, etc.) will be converted to symbols (see appendix C). Graphics characters will be represented in one of two ways. The default method is to print the keystroke which produced it. For example, O is represented as [SW] meaning SHIFT-W, and ◀ is printed as [C*] meaning C= * (the commodore key and '*'). Through the command channel (see page 8) you can select the second method of printing these codes. In this mode, they are just converted to their ASCII value (e.g. [213]).

See appendix F, sample program 1 for a demonstration of how control and graphics characters can be listed.

DAISY WHEEL EMULATION

Modes 2, 6, and 9 are identical to 0, 4, and 7 except that the characters are printed in near-letter quality on GEMINI, DELTA, EPSON, and PANASONIC printers only. Due to

the increased print quality, printing speed in these modes is significantly slower. Text in this mode will be spaced at 6 lines-per-inch. You can select 8 LPI through the command channel (page 8).

Underlining is also supported in this mode. It is turned on and off the same way as underlining in the printer:

```
  Underlining on: ESC - 1
                  [CHR$(27)CHR$(45)CHR$(1)]
 Underlining off: ESC - 0
                  [CHR$(27)CHR$(45)CHR$(0)]
```

If you have the proper printer, try this simple program:

```
10 OPEN 1,4,9:OPEN 2,4,7
20 UB$=CHR$(27)+"-"+CHR$(1)
30 UE$=CHR$(27)+"-"+CHR$(0)
40 PRINT#1,"Daisy wheel quality"
50 PRINT#1,UB$"Underlining"UE$" too!"
60 PRINT#2,"Dot-matrix quality"
70 CLOSE 1:CLOSE 2
```

See appendix D for a printout of the full character set in this mode.

LOCKING THE INTERFACE

By printing in modes 20 through 29, you can lock the interface in modes 0 through 9. For example, to lock in the upper/lower case mode (7), use mode 27 (20+7) and print something. The interface is then locked in that mode, even if you close the file and try to re-open it in a different mode. Once locked in this manner, the only way to unlock it is to power down or by using the command

channel. It may be necessary to lock the interface in a certain mode before starting a word processor.

```
10 REM Demo: how to lock the interface
20 OPEN 1,4,29:REM Ready to lock in mode 9
30 PRINT#1:REM You must print something to
40 REM complete the lock.
50 CLOSE 1:END
```

USE WITH WORD PROCESSORS

Most word processors available for the Commodore computers do not send secondary addresses (the 'mode' in the OPEN statement). This means that you will always be stuck in the default mode, which is 0. Consequently, you need to do one of two things: first, you can lock your interface in the desired mode before starting the word processor (p. 14). If you use this method, you cannot change the mode until you exit the WP. Alternately, you may go ahead and start the word processor and then send the following data to change to the desired mode

CHR$(27)CHR$(46)CHR$(mode)

If you use this method, the interface is not locked. You could alternate between modes 7 and 9, for example, to switch between dot-matrix and daisy-wheel emulation.

USING TABS

The TAB function works a little different with a printer than you are used to. First of all, PRINT#1,TAB(5) will give a ?SYNTAX ERROR.

This seems to be a bug in the Commodore computers.   The only way it will work is

          PRINT#1,""TAB(5)

Notice the similar statement in the sample program on page 10.
     If your printer automatically sends its own line feeds, you are out of luck. The only way for you to simulate tabs is to print a computed number of spaces to go from the current to the next column.
     If your printer doesn't send its own line feeds, you can use the TAB function by opening the file without the line feed option. The only difference is that to advance to the next line, you must send a CHR$(10).

```
10 REM TAB EXAMPLE
20 OPEN 1,4,1
30 PRINT#1,""TAB(5);"5TH"
40 PRINT#1,""TAB(20);"20TH"
50 PRINT#1,""TAB(10);"10TH";CHR$(10)
60 CLOSE 1
```

APPENDIX A
Troubleshooting

SYMPTOM	POSSIBLE CAUSES
Computer locks up when trying to print	- Printer not connected - Printer deselected - Printer off
Doesn't lock up but nothing prints	- Wrong device number for switch #1
Device not present error	- Serial cable disconn. - Wrong device number for switch #1
File not open error	- Wrong file number - BASIC program altered
Line of data writes over previous line	- Using mode 1,5,8 - Switch 3 (LF) off
Spaces between each line	- Using mode 0,2,3,4,6,7 or 9 with switch 3 on, and a fixed line feed printer - Using file #'s 128-255
Getting garbage or graphic characters	- Listing program in mode 4,5, or 6 or with switch 4 off
Interface not paying attention to mode in OPEN	- Interface locked, try 'U' in command channel
Double width on & off not working	- Switch 2 set wrong - Wrong printer type set in command channel - 'N' sent to com. chan.

APPENDIX B
ASCII conversions

I.   Standard printer (Switch 2 off or 'S')

    CHR$(15) becomes CHR$(20)
    CHR$(20) becomes CHR$(15)

II.   Okidata printer (Switch 2 on or 'O')

    CHR$(14) becomes CHR$(31)
    CHR$(31) becomes CHR$(14)
    CHR$(15) becomes CHR$(30)
    CHR$(30) becomes CHR$(15)

III.   No double-width changes ('N')

    No changes

A.   Upper case only

    CHR$(17) changes to upper/lower case
    No other codes affected unless listing
     a BASIC program

B.   Upper/lower case

    CHR$(145) changes to upper-case-only
    CHR$(65 to 90) become CHR$(97 to 122)
    CHR$(193 to 218) become CHR$(65 to 90)
    CHR$(97 to 122) become CHR$(65 to 90)
    No other codes affected unless listing
     a BASIC program

APPENDIX C
Listing abbreviations

ASCII	SYMBOL	KEY
5	WHT	White
8	DISH	Disable shift
9	ENSH	Enable shift
14	SWLC	Switch to lower
17	DOWN	Cursor down
18	RVON	RVS on
19	HOME	Home
20	DEL	Delete
28	RED	Red
29	RGHT	Cursor right
30	GRN	Green
31	BLU	Blue
129	ORNG	Orange
133	F1	Function F1
134	F3	Function F3
135	F5	Function F5
136	F7	Function F7
137	F2	Function F2
138	F4	Function F4
139	F6	Function F6
140	F8	Function F8
141	SHRT	Shift Return
142	SWUC	Switch to upper only
144	BLK	Black
145	UP	Cursor up
146	RVOF	RVS off
147	CLR	Clear
148	INST	Insert
149	BRWN	Brown
150	LRED	Light red
151	GRY1	Gray 1
152	GRY2	Gray 2
153	LGRN	Light Green
154	LBLU	Light blue
155	GRY3	Gray 3

ASCII	SYMBOL	KEY
156	PUR	Purple
157	LEFT	Cursor left
158	YEL	Yellow
159	CYN	Cyan

APPENDIX D
Daisy wheel quality character set

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

APPENDIX E
Parallel cable pinout

PIN#	FUNCTION
1	Data strobe (active low)
2	Data bit 0
3	Data bit 1
4	Data bit 2
5	Data bit 3
6	Data bit 4
7	Data bit 5
8	Data bit 6
9	Data bit 7
10	Acknowledge (active low)
31	Printer reset (active low)
16, 19-30, 33	Grounded
All others	No connection

```
10 REM Graphic character listing demo
20 OPEN 1,4,15:REM Command channel
30 OPEN 4,4,1:REM Data channel
40 PRINT#4,"ASCII      KEYSTROKE      ASCII MODE"
50 PRINT#4,CHR$(10)
60 FOR CODE=160 TO 223
70 PRINT#4,CODE:PRINT#1,"K":REM KEYSTROKE MODE
80 PRINT#4,CHR$(34);TAB(10);CHR$(CODE)
90 PRINT#1,"A":REM ASCII MODE
100 PRINT#4,CHR$(34);TAB(23);CHR$(CODE)
110 PRINT#4,CHR$(10)
120 NEXT CODE
130 CLOSE 1:CLOSE 4:END
```

Subroutine for VIC to print graphic character
(Put screen code in GC)

```
60000 REM Graphic for Vic-20 on Gemini, Epson
60010 OPEN 99,4,5:PRINT#99,CHR$(27)CHR$(76)
      CHR$(16)CHR$(0);
60020 BASE=32768+(PEEK(36869)AND15)*1024+8*GC
60030 FOR PB=7 TO 0 STEP-1:DB=0
60040 ZZ=7:FOR ZA=BASE TO BASE+7
60050 IF PEEK(ZA)AND(2^PB)>0 THEN DB=DB+2^ZZ
60060 ZZ=ZZ-1:NEXTZA:PRINT#99,CHR$(DB)CHR$(DB);
60070 NEXT PB
60080 CLOSE 99:RETURN
```

Changes for Prowriter, C-Itoh

```
60010 OPEN 99,4,5:PRINT#99,CHR$(27)CHR$(83)
      "0008"
60040 ZZ=0:FOR ZA=BASE TO BASE+7
60060 ZZ=ZZ+1:NEXT ZA:PRINT#99,CHR$(DB);
```

Changes for the C-64 to Gemini or Epson

```
60020 BASE=(PEEK(53272)AND 14)*1024+GC*8
60023 FL=0:IFBASE>=8192 OR BASE<4096THEN60030
60025 FL=1:POKE56334,PEEK(56334)AND254:POKE
      1,PEEK(1)AND251:BASE=BASE+49152
60060 ZZ=ZZ-1:NEXTZA:DD(PB)=DB
60075 IFFL>0THENPOKE1,PEEK(1)OR4:POKE56334,
      PEEK(56334)OR 1
60077 FOR PB=7TO0STEP-1:PRINT#99,CHR$(DD(PB));
      CHR$(DD(PB));:NEXT PB
```

Changes for the C-64 to Prowriter, C-Itoh

```
60010 OPEN 99,4,5:PRINT#99,CHR$(27)CHR$(83)
      "0008"
60020 BASE=(PEEK(53272)AND 14)*1024+GC*8
60023 FL=0:IFBASE>=8192 OR BASE<4096THEN60030
60025 FL=1:POKE56334,PEEK(56334)AND254:POKE
      1,PEEK(1)AND251:BASE=BASE+49152
60040 ZZ=0:FOR ZA=BASE TO BASE+7
60060 ZZ=ZZ+1:NEXTZA:DD(PB)=DB
60075 IFFL>0THENPOKE1,PEEK(1)OR4:POKE56334,
      PEEK(56334)OR 1
60077 FORPB=7TO0STEP-1:PRINT#99,CHR$(DD(PB));:
      NEXT PB
```


>>> Preliminary manual <<<

Coming soon:

Hi-res screen dumps for C-64 and VIC to various printers. Write or call Xetec (see back page) for your copy when available.

INDEX

NOTES

- 24 -

NOTES

A PERSONAL WORD

Thank you for purchasing one of our products. We at Xetec have made every effort to deliver to you a quality product that will be a real asset to your computing. You are a real asset to us, too. Please do not hesitate to contact us with your questions or suggestions, either at the address below or by calling us at (913) 827-0685 during normal business hours (central time).

NOTICE OF LIMITED WARRANTY

Xetec, Inc. warrants this product against defects in material and/or workmanship for a period of one year from the date of purchase and will, during this warranty period, repair or replace any defective components, or, at our option, the entire product, at no charge, provided that the product is returned shipping prepaid with proof of original purchase to:

Xetec, Inc.
3010 Arnold Rd.
Salina, KS. 67401

This warranty does not apply if, in Xetec's opinion, the product has been damaged by accident, misapplication, misuse, or as a result of service, modification, or adjustment, by other than the manufacturer, Xetec, Inc. No other warranties are expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Xetec, Inc. is not responsible for consequential damages of any kind arising from or connected with the use of this product.

VIC-20 and C-64 are trademarks of Commodore Business Machines, Inc.