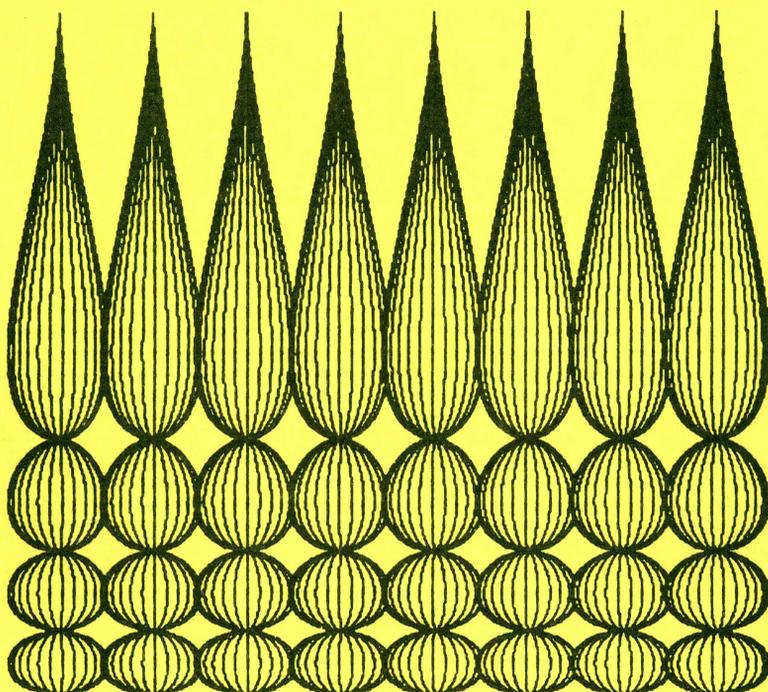


# PKASO™ ///

Printer Driver  
for  
Apple ///  
and  
Matrix Printers

## Users Manual



INTERACTIVE STRUCTURES INC.  
146 Montgomery Avenue  
P.O. Box 404  
Bala Cynwyd, PA 19004



PKASO(tm) ///  
welcomes  
the Apple Dot Matrix Printer

November 1982  
Software revision 3.1

PKASO /// has been updated to support the PKASO AF12 for the Apple Dot Matrix Printer (DMP) providing all of the features of the PKASO ///. The Apple DMP is almost identical to the NEC PC-8023 and C.itech Pro/Writer printers, and has been implemented as an enhanced variety of this species. Familiarize yourself with the PKASO /// manual and refer to the sections pertaining to these printers for specific PKASO features and commands available on the DMP.

#### System Configuration Changes

To install your PKASO driver for use with the Apple DMP, follow the instructions as described in Appendix D of your PKASO /// manual. The following changes have been made to the Driver Configuration Block to support the Apple DMP:

Byte 0 Printer type.  
Set to --> 07 <-- for NEC, C.itech or Apple DMP.

Byte A Printer Sub-type.  
Set to --> 04 <-- for Apple DMP.

The definition of all other bytes remains the same.

#### Supported Character Sizes

PKASO /// for the Apple DMP supports three character sizes and their expanded sizes. These sizes are 10, 12, and 17 characters per inch. Also the 160 dots per inch proportional size and its expanded size are supported using the ^E3, and ^E11, commands.

**PKASO (M)**  
**PRINTER INTERFACE AND DRIVER**

**For Apple III and  
Matrix Printers**

**Operating Manual**

**PKASO Driver Revision 3  
October, 1982**

**IS Interactive Structures Inc.  
146 Montgomery Avenue, Bala, PA 19004**

## PKASO and PFS Graph ///

Yes, you can print PFS Graphs with PKASO!

These are graphs from PFS Graph for Apple /// done with PKASO.

As follows:

**Method A:** (You need the Apple /// Pascal system) Save the pictures, then use our PRINTPIX or PRINTCLR routines.

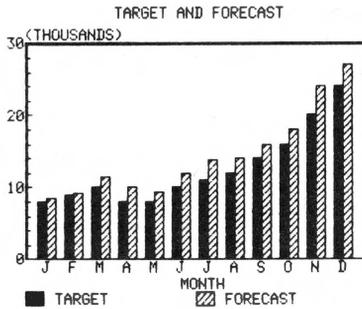
1. Save the pictures from PFS (option 4 get/save, then option 3 save picture).
2. Boot your Pascal /// system.
3. X) execute PRINTPIX or PRINCLR if you have a color printer. These programs are on our PKASO /// disk.
4. Print with any of our options.

**Method B:** Use the print routines built into PFS Graph (good for most printers).

1. Install our MINI DRIVER on your PFS disk following the installation instructions in your PKASO manual.
2. Use PFS' print features.

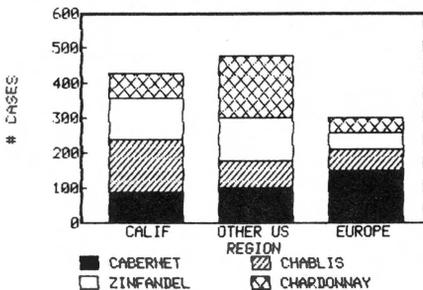
Unfortunately, PFS uses bidirectional print and offers only 2 sizes. You can correct the bidirectional problem in most printers. Use the appropriate codes to select unidirectional printing in a PFS set-up string.

Using PRINTPIX from the PKASO disk



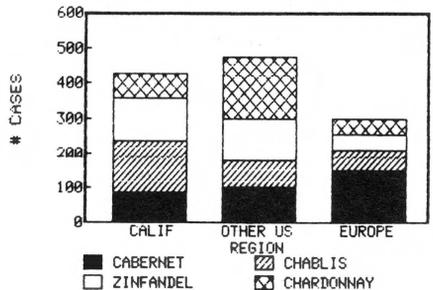
Using PFS' print routine with  
Bi-Directional "jitter"

MAY SHIPMENTS BY REGION



After defeating bidirectional on the  
NEC with ESCAPE <

MAY SHIPMENTS BY REGION



**COPYRIGHT NOTICE**

**THE SOFTWARE AND DOCUMENTATION CONNECTED WITH THIS PRODUCT ARE COPYRIGHTED AND ALL RIGHTS ARE RESERVED. No portion of the product may be copied, translated, or reproduced in any fashion for use by or sale to any individual or firm other than the original purchaser. Due to the proprietary nature of the techniques used, any unauthorized attempt to copy any portion of the product will be treated under the Copyright Law of the United States, and may result in both civil and criminal penalties and imprisonment.**

**PKASO (™)**  
**PRINTER INTERFACE and DRIVER**  
**Apple /// Operating Manual**

Table of Contents

Copyright Notice . . . . .	1
Requirements & Quick Startup Procedure . . . . .	3
Features of the PKASO Interface . . . . .	4
Installing the system . . . . .	5
Driver Configuration Block (DCB) . . . . .	6
Use with Software Packages . . . . .	7
Use with Business Basic . . . . .	8
Use with Pascal . . . . .	9
Status of the Printer . . . . .	10
The PKASO /// Command language . . . . .	11
Size of the printed characters . . . . .	15
Line Width and Margins . . . . .	17
TABS . . . . .	19
Vertical Spacing . . . . .	20
Form-feed and Line-feed . . . . .	21
Changing Character Sets with SoftFont . . . . .	22
Text Screen Snapshot Print . . . . .	23
HiRes Screen Print . . . . .	24
Gray Scale Printing . . . . .	27
Dot Graphics with the SuperRes system . . . . .	29
Direct Binary Output with BYTE Commands . . . . .	31
Sending ESCAPE Sequences to the Printer . . . . .	34
Soft Reset and Printer Selection . . . . .	37
Memory Usage & Assembly Programming . . . . .	37
SOFTWARE COMPATIBILITY HINTS . . . . .	39
Appendix A - Command Quick Reference . . . . .	41
Appendix B - Status & Control Calls . . . . .	42
Appendix C - Mini Driver . . . . .	46
Appendix D - Installation Details . . . . .	47
Appendix E - Driver Configuration Block . . . . .	49
Appendix F - Printers Supported . . . . .	50
Warranty & Liabilities Disclaimer . . . . .	51
Command Charts . . . . .	52

**PKASO (TM)  
PRINTER INTERFACE AND DRIVER  
Apple /// Operating Manual**

## INTRODUCTION

You are now the proud owner of a PKASO (TM) Printer Interface System, the first and most powerful interface system available for matrix printers and the Apple /// microcomputer.

The PKASO Apple /// software package includes this manual and an Apple /// software diskette. It works in conjunction with the PKASO interface card, and provides complete support for Apple /// in both Native and Emulation modes.

This manual includes a QUICK REFERENCE summarizing the operation of the PKASO system. (Appendix A)

A fold-out command chart at the end of this manual includes the details, sizes, and eccentricities of your printer.

## WHAT YOU NEED . . .

A matrix printer: C. Itoh 8510,  
(See Appendix F) Centronics 739 or 122,  
Epson MX70, 80, or 100,  
IDS Prism, 560, or 460,  
NEC PC8023A-C,  
Okidata 82A, 83A.

The PKASO /// system is designed for Apple /// with at least 128K of memory.

## GETTING STARTED . . .

If you're experienced with the Apple ///, here is a quick startup procedure: With all power OFF, plug the interface card into slot 1 and connect the cable to the printer. Use SCP to install PKASO.DRIVER (from the PKASO Diskette) onto your bootable diskette as .PRINTER. Set byte 0 of its DCB to the proper number for your printer. Then RUN PRINTPIX (from the PKASO Diskette). PRINTPIX lets you load FOTO files and print them.

## **FEATURES OF THE PKASO SYSTEM**

The PKASO Interface System is designed to smooth out the rough spots in using a printer and a computer together. It will make routine operations easier, and it will add extra capabilities to the system.

- Ⓢ A simple language for easily controlling printer features from any program.
- Ⓢ HiRes Screen Print with your choice of size, rotation, and style.
- Ⓢ Choice of 2 HiRes print modes for: (A) graphing, or (B) shaded images.
- Ⓢ SoftFont (M) Character Mode lets you print all text using the present screen character style.
- Ⓢ Text screen snapshot print, 40 or 80 columns, can print with the same character style you see on the screen.
- Ⓢ HalfTone Gray Scale mode lets you print with 16 levels of gray.
- Ⓢ SuperRes (M) Dot graphics, lets you print dot graphics directly from any program.
- Ⓢ Visible signal on screen when the printer is off line or out of paper.
- Ⓢ Easy-to-use Tabs for indentation and column printing.

## INSTALLATION

### A. THE INTERFACE CARD . . .

First, **TURN OFF** your Apple and your printer!

The PKASO interface system includes an interface electronics card and a cable. The gold edge connector of the card plugs into one of the I/O Expansion Slots located inside the case of the Apple, on the rear portion of the processor board. (Your Apple /// manual gives all the details on this.)

The customary location for a printer is expansion Slot 1, the leftmost slot as you look from the front of the machine. The PKASO Driver and the examples in this manual are designed for a PKASO card in Slot 1.

Plug the interface card firmly into the Slot, with the cable extending out one of the openings in the back of the Apple cabinet. Plug the PKASO cable connector into the matching connector at the rear of the printer.

Turn the printer power on. (Consult your printer manual for the proper way to do this, how to make sure it worked, and what to do if it didn't.) The printer is now ready to print and is awaiting data from the Apple.

## B. THE SOS DRIVER...

Part of the PKASO system when running in Apple /// native mode is a software module called a Driver. The driver is in charge of accepting data from the language interpreter or application program you are using, and transmitting it properly to the printer.

The driver must first be installed (the software equivalent of plugging it in) on your boot diskette. It then becomes part of your SOS configuration and is ready to perform whenever printing is requested. Apple gives you very complete instructions for doing the installation in the Standard Device Drivers manual. APPENDIX D OF THIS MANUAL GIVES THE FULL PROCEDURE.

Briefly, you run the System Configuration Program (SCP) and follow its instructions. First set the printer slot to 1. Delete any other printer drivers (.PRINTER, .QUME, or .SILENTYPE.) When SCP asks for a driver file, insert the PKASO Apple /// disk in drive 1 and enter the name PKASO.DRIVER.

### DCB

#### THE DRIVER CONFIGURATION BLOCK

The driver contains a Driver Configuration Block, (DCB) which specifies how the printer options will be set when you OPEN the printer for the first time or whenever you send a PKASO SOFT RESET command to the printer. The full description of the DCB is in Appendix E. These settings are default conditions and can of course be changed later by PKASO commands.

The first time through the startup procedure, just make 2 changes to the DCB:

1. Set the printer type, byte 0, to the proper number for your printer. (See Appendix E)
2. Set the printer sub-type, byte 10, to 0 for 8.5" wide paper, or to 1 for 13.9" paper.

As you become familiar with the options in the DCB, you may want to re-adjust them (using the SCP program) so that the printer always comes up with your most convenient settings for the software in use.

## PRINTING FROM A PACKAGE

The PKASO Driver follows Apple /// printer conventions and allows your matrix printer to be used by any software which requests printing. All you need to do is install the PKASO system, and you have a 10 character-per-inch printing capability. If you have a package which needs only rudimentary printing capability without commands or graphics, you can install MINI.DRIVER instead of the full PKASO.DRIVER. (See Appendix C.)

This manual includes a section on Hints for use with popular languages and software, as well as a complete dump procedure for Apple Business Graphics.

## BUSINESS BASIC

If you want to enter printing commands at the keyboard or write a Business Basic program which prints, it's simple. Just OPEN the printer, PRINT, then CLOSE it.

### OPENING...

The OPEN step informs Basic that you wish to print and (if it hasn't been done yet) initializes the printer driver. Here is an example of how this is done:

```
OPEN #1, .PRINTER from command level, or  
10 OPEN #1, ".PRINTER" from a BASIC program.
```

Refer to the sections on FILES and FILE I/O in your Business Basic manual for specifics on the OPEN command.

### PRINTING...

Once the printer has been OPENed it is ready for use. There are two ways to access the printer: PRINT# and OUTPUT#.

Method one is to use the PRINT# statement:

```
PRINT #1;"PKASO" from command level, or  
20 PRINT #1;"PKASO" from a program.
```

The second method uses the OUTPUT# statement. Once this statement has been used, all output from PRINT, TRACE, and LIST statements and the SOS CATALOG command will be printed on the printer instead of the video screen.

```
30 OUTPUT #1
40 PRINT "THIS GOES TO THE PRINTER"
```

To switch output back to the video screen,

```
50 OUTPUT #0
60 PRINT "THIS GOES TO THE SCREEN"
```

Printer output is switched off whenever the program terminates abnormally.

CLOSING...

Once you have finished using the printer you must CLOSE it. This is done as follows:

```
70 CLOSE #1
```

## PASCAL

The printer appears to the PASCAL System to be the same as a text file. The PASCAL documentation, particularly the Programmer's Manual, contains a good discussion on the use of printers. To set up the file:

```
VAR PR:TEXT;
```

To open the file:

```
REWRITE (PR, '.PRINTER');
```

Then use WRITE or WRITELN to PR to print.

## PRINTER STATUS

If you try to print with the printer in local mode or out of paper, a blinking message will appear in the lower right corner of the video screen. The PKASO Driver will then wait until printing is possible.

This is a "non-destructive" display, and anything printed in that location will be replaced when the condition is remedied. If you have switched to a full-screen graphics display, you will not see the message, since it is displayed on the text screen only.

This, however, brings processing to a standstill since nothing will happen until the printer is ready. To prevent this, PKASO lets you check the status of the printer before trying to print. You can then prompt the user, come back later and try again, etc.

Appendix B gives the details of Status Calls. Briefly, OPEN the printer, then make a Status Call to the PKASO Driver, as follows:

```
INVOKE REQUEST.INV (on Bus.Basic 1.1 disk)
PERFORM STATUS(%3,@STATUSLIST$)".PRINTER"
PRSTATUS = ASC(MID$(STATUSLIST$,5,1))
```

Performing STATUS fills your string (STATUSLIST) with information about the printer. The 5th character is the ready indicator. If PRSTATUS is 0, the printer is ready. (1 is off-line, 2 is paper-out, 3 is both.)

For the corresponding procedure in Pascal, see Appendix H of the Standard Device Drivers Manual.

## PKASO COMMANDS

There are two types of information you need to send to a printer:

1. PRINTABLE information, and
2. CONTROL information, which does not produce any printing, but instructs the printer to change sizes & options.

To send CONTROL information to the printer and the PKASO system, a miniature language has been designed called the PKASO /// Command Language. It is an enhancement to, and in some cases a replacement for the standard ASCII control character set which includes Carriage Return, Form Feed, etc.

PKASO Command =

`<~> <Command Char> <Option, ...Option,>`

The ~ Lead-in character is how the PKASO system identifies a command. The ~ is completely printable and can be sent from within any program.

The Command Characters are ordinary printable characters as given in the description of each command.

Options will usually be just decimal numbers, but may also be given in Literal or Hexadecimal form. The description of each command gives the type and number of options required. You should FOLLOW ALL OPTIONS, INCLUDING THE LAST ONE, WITH A SINGLE COMMA, as shown above. (This insures that PKASO sees the end of each option and does not attach the last one to any numbers you may be printing next.)

## OPTION FORMAT

In sending the Options part of a command, you have your choice of three ways to express yourself:

(a.) Decimal Numbers, 0 thru 255, using the digits <0123456789>,

(b.) Hexadecimal Numbers, \$0 thru \$FF, using the digits <012345689ABCDEF> preceeded by a Dollar Sign, \$.

(c.) Literal Characters, any single ASCII character preceeded by a "Grave Accent", `.

For example, some valid Option strings are:

1,2,3,      22,\$7F,      `X`,`y,2,      255,

The following are not valid:

1,2,3      22\$7F,      `Xy,2,      256

(No disaster will occur, but you may get surprising results.)

The PKASO Command interpreter does not require that you use the comma as a separator for Options. You can use any character which unambiguously separates the items. You may prefer to use a space, asterisk, or any other character which PKASO will not interpret as another Option. Our convention is the comma, and examples in this manual will use commas.

For example, to select compressed (16.5 char. per inch) printing, print "~E2,". This style of command can be used in the middle of word processing text, or even in a label field in Visicalc, for bold titles, etc.

## ALTERNATE COMMAND FORMS

## A. CHANGING ~ TO SOMETHING ELSE...

The ~ lead-in character was chosen to denote PKASO Commands because it is printable, yet uncommon in ordinary printing. To change the lead-in to another character instead, you can (a.) Change DCB byte 9 to the code for your desired lead-in character, such as \$B3 for a #, or (b.) Use the following PKASO Command:

LEAD-IN Cmd:	Changes Command Lead-in.
Format:	~O<NewCode> ,
Ctrl Char Form:	^RC~O
From Basic:	PRINT "~O";NEWCODE;",";
From Pascal:	WRITE (PR,'~O',NEWCODE,',');

For example, you could use ~O\$B3, or ~O`# to change the lead-in to #, then #O\$FE, or #O`~ to change it back to ~.

The Option NewCode should be one of the ASCII code numbers 32 thru 127. A NewCode of zero disables the Lead-in Character processing of PKASO, and only the Control Character form of commands will be recognized. (See the next section.)

## B. CONTROL CHARACTER FORM...

An alternate command format is also included which is more concise, but uses ASCII Control Characters for Commands and Options.

PKASO Command = <ASCII CTRL Char> <Options>

Options in this case are supplied in only one way: as ASCII character values, codes 0 thru 255, which may or may not be printable.

For example, the ~O\$B3 command becomes <sup>Ctrl</sup>O#, an option of 5 becomes CHR\$(5), etc.

This mode of supplying commands to PKASO is always active IN ADDITION TO the ~ format, and provides a second method for supplying commands which may be preferable for some applications.

It is ONLY suited for software which adheres strictly to Apple ///'s convention of allowing all Control Characters to be printed. In Pascal, for example you can use UNITWRITE to send most commands in Ctrl Char Form, however a \$00 can not be sent.

To select this format exclusively, set DCB byte 9 to a 00, or use the PKASO Lead-in Command above with a NewCode of 0.

## INITIALIZATION OF THE SETTINGS

The first time the PKASO system is opened after a boot, or whenever a PKASO SOFT RESET command is done, the printer is set up as described by the Driver Configuration Block. These are the "Default" settings. After this, the PKASO system does no more resetting or initializing, so printer settings will generally stay the same unless you change them.

**SIZE COMMAND**

The **SIZE** command can request any of the printing widths provided by your printer.

SIZE Command:	Selects character width
Cmd Format:	~E <SizeCode> ,
Ctrl Char Form:	^TRC E
From Basic:	PRINT "~E";SIZE;",";
From Pascal:	WRITE (PR, '~E', SIZE, ', ,')

Selecting a character size is relatively simple. The command may be included anywhere in your text, and the character size will indeed change in the middle of the line where the command is.

The sizes are selected by a size code. This table shows the codes and the widths expressed in characters per inch.

Size	Code	Command	Size	Code	Command
5 CPI	8	~E8,	10	0	~E0,
6	9	~E9,	12	1	~E1,
8.25	10	~E10,	16.5	2	~E2,

You also may control any proportional spacing and emphasize modes using **ESC** and **BYTE** commands.

For example, to select 16.5 CPI (compressed) printing, use ~E2.

**NOTE:** Use the **SIZE** Command, not the printer's direct compress or expand ASCII codes, or you may find that the **PKASO** driver overrides your settings.

**SIZE COMMAND (Cont'd.)**

The following shows a sample of the range of print sizes from 5 cpi to 16.5 cpi.

**FIXED SPACING**

The Quick	5.0 CPI
The Quick Brown	6.0 CPI
The Quick Brown Fox	8.25 CPI
The Quick Brown Fox Jumped	10 CPI
The Quick Brown Fox Jumped Over	12 CPI
The Quick Brown Fox Jumped Over The Lazy Dog.	16.5 CPI

**PROPORTIONAL SPACING**

The Quick	5.0 CPI
The Quick Brown	6.0 CPI
The Quick Brown Fox	8.25 CPI
The Quick Brown Fox Jumped	10 CPI
The Quick Brown Fox Jumped Over	12 CPI
The Quick Brown Fox Jumped Over The Lazy Dog.	16.5 CPI

Consult your Command Chart for the sizes available on your printer.

**WIDTH COMMAND**

The **WIDTH** command sets the maximum number of characters on a line. If the margins are all set to zero and you start printing characters at the left edge of the paper, this is how many you can print before the PKASO system inserts a Carriage Return and a Linefeed and takes you to the next line.

<b>WIDTH Cmd:</b>	Sets line width in chars.
<b>Cmd Format:</b>	~D <Width in Characters> ,
<b>Ctrl Char Form:</b>	^TRC D
<b>From Basic:</b>	PRINT "~D";WIDTH;" , "
<b>From Pascal:</b>	WRITE<PR, '~D', WIDTH, ', ' >;

**NOTE:** Most Apple /// interpreters such as Basic and Appleswriter /// handle the end of each line themselves, in which case the PKASO width can be left at 255, which switches off PKASO line width checking.

Some languages may need line width to be enforced by the PKASO system, so the Width parameter should be set to match the printer size, the paper width, and the character size in use. For example, for standard 8.5" paper and 10 characters per inch, use a width of 80 characters: PRINT "~D80,";

**MARGIN COMMAND**

Left and right margins are supported by PKASO. They are measured in characters, inward from the edges of the page, as set by the Width command. Note that this means the **WIDTH** and **MARGIN** commands interact, and the effective number of characters you can print on a line is Width minus Left Margin minus Right Margin.

A third type of margin is also provided. This is called the Wrap Margin. If printing exceeds the right margin, it "wraps" around the page to the next line and starts not at the left margin, but at the Wrap Margin. This is useful for setting up paragraph indentation.

MARGIN Cmd.:	Sets all margins.
Cmd Format:	~C<LM>,<RM>,<WM>,
Ctrl Char Form:	^RL^C
From Basic:	PRINT "~C"; LM;"",";RM;"",";WM;"",
From Pascal:	WRITE (PR, '~C', LM, ', ', RM, ', ', WM, ', ', );

The Margin command takes effect as it is encountered in printing. This means it affects the right but not the left margin of the present line. Follow the command with a Carriage Return to start a new line at the new left margin. (Omit the final ";" in Basic, or use WRITELN in Pascal.)

Since margins are calculated with respect to the present page width, you should avoid setting things up so that there is no room on the page. (Width minus margins is too small.)

For example, lets say you want to set up a centered paragraph format with 60-column lines and the first line indented 5 spaces. Using the standard 80-column Width, this would be done as follows:

```
PRINT "~C";15;"",";10;"",";10;"","
```

## TAB COMMANDS

The PKASO system includes its own flexible tab feature. The operation is similar to tabs on a standard typewriter: you can SET tabs at up to 16 different columns and you can CLEAR all tabs. The HT command is used during printing to space over to the desired column.

TABSET Cmd.:	Adds N stops to tab list.
Cmd Format:	~F<N>,<T1>,... <TN>,
Ctrl Char Form:	<sup>Ctrl</sup> F
From Basic:	PRINT "~F";n;"," T1;",";T2;...;TN",";
From Pascal:	WRITE(PR,'~F',N,',', T1,',',T2,...,TN,',');

TABCLR Cmd.:	Clears all tab settings.
Cmd Format:	~F0,
Ctrl Char Form:	<sup>Ctrl</sup> F
From Basic:	PRINT "~F0,";
From Pascal:	WRITE (PR,'~F0,');

HT Command:	Spaces to next tab stop.
Cmd Format:	~I
Ctrl Char Form:	<sup>Ctrl</sup> I
From Basic:	PRINT "~I,";
From Pascal:	WRITE (PR,'~I,');

The system also has an auto-tab mode: if you use the HT command without setting any tab stops, or if you are beyond the last stop on your tab list, this takes you TO THE NEXT EVEN MULTIPLE OF 8 COLUMNS.

Tabs are used in the order in which you set them. If the next tab stop is to the left of the present print position, THE SYSTEM DOES A CARRIAGE RETURN AND A LINEFEED, then goes to the requested column. Thus one set of tab stops can completely format a multi-line mailing label or payroll check printout.

## TAB COMMAND (Cont'd)

For example, to set up a mailing label printing format,

```
Name
Address
City                State  Zip
```

you could use the following TABSET command:

```
~F4,1,1,20,27,
```

Then an entire label can be printed as a simple string with interspersed HT commands.

```
100 HT$ = "~I"
110 PRINT#1; NAME$ + HT$ + ADDR$ + HT$ +
    CITY$ + HT$ + STATE$ + HT$ + ZIP$
```

## VERTICAL SPACING COMMAND

This command adjusts the function of Line-feed for graphics or text spacing.

VS Command:	Sets Vertical Spacing
Cmd Format:	~T<V> ,
Ctrl Char Form:	^T
From Basic:	PRINT "~T";V;" ,";
From Pascal:	WRITE (PR, '~T', ' ', ' ', V, ' ', ' ');

The number V selects the spacing as follows:

- 0 - Standard text spacing (6 LPI)
- 1 - HiRes print spacing for your printer.  
(See the Command Chart.)
- 2 - Gray Scale or graphics 6 dot spacing.
- 3 - Bit graphics 7-dot spacing.

**FORM FEED COMMAND**

This is the customary ASCII new page command. It puts you at the top line of the new page, at the left margin. Your printer's form length setting (usually 11" or 66 lines) determines the position of the top of the new page.

FF Command:	Skips to the next page
Cmd Format:	~L
Ctrl Char Form:	Ctrl-L
From Basic:	PRINT "~L";
From Pascal:	WRITE (PR, '~L');

**LINE-FEED COMMAND**

LF Command:	Scrolls paper up 1 line.
Cmd Format:	~J
Ctrl Char Form:	Ctrl-J
Use from Basic or Pascal is Automatic.	

Line Feed moves to the left margin of the next line.

The PKASO system has three vertical spacing modes: text, dot graphics, and gray scale. Line-feed spaces the paper up by the proper number of dot spaces for the present spacing mode.

The convention in Apple /// software is that the software, not the printer, furnishes the necessary line-feed at the end of each line of text. The PKASO system merely passes these line-feeds through to the printer.

If you want to use your printer's auto line-feed after carriage return feature instead, then you can, using SCP, set Driver Configuration Block byte 7 to a 0. PKASO will then block line-feed characters from reaching the printer and causing double spacing.

## SOFT-FONT COMMAND

Your printer has a built-in character set or Font which it uses for standard text printing. Whenever a printer receives a printable character from the computer, its character set table is automatically consulted to get the dot-pattern which forms the symbol on the paper. This is convenient and fast, since it all happens in the hardware and firmware of the printer.

If you want a different style of characters or a symbol which is not included on your printer, however, you have a problem. PKASO gives you the solution by providing a choice of the standard printer characters OR SoftFont, the Apple /// screen characters! The SoftFont system uses the printer's dot graphics capability, and Apple ///'s present character table to produce characters.

FONT Command:	Selects Print Style
Cmd Format:	~R<F> ,
Ctrl Char Form:	^~R
From Basic:	PRINT "~R";F;",";";
From Pascal:	WRITE (PR,'~R',F,',',');

**F** is the Font Code, used as follows:

Standard Printer Font	Code 0
Apple /// Screen Font	Code 1
Compressed Screen Font	Code 3
Inverse Screen Font	Code 5
Inverse & Compressed	Code 7

**NOTE:** The FONT Command and its options depend on the printer. (See you Command Chart). The selection of character SIZE does not have the normal effect on SoftFont printing. SoftFont characters are a fixed width and height.

## TEXT PRINT SCREEN DUMP COMMAND

This command prints a snapshot copy of the text screen at the present left margin.

TPRINT Cmd.:	Prints the Text Screen
Format:	~P<Mode>, .
Ctrl Char Form:	Ctrl-P
From Basic:	PRINT "~P";MODE;";";
From Pascal:	WRITE (PR, '~P', mode, ', ');

Four modes are provided to match Apple /// screen text formats:

- 0 - 40 column standard screen 1
  - 1 - 40 column screen 2 (\*)
  - 2 - 80 column standard screen
  - 3 - 80 column reversed interlace (\*)
- (\* - for custom software)

If it is possible on your printer, the PKASO system prints the text screen using the Apple /// screen character set and line spacing. After the print, line spacing returns to 6 LPI.

See your Command Chart for the standard TPRINT mode, text font or screen font, for your printer. If screen font is standard, then:

1. To print in inverse mode (white on black) add 4 to the mode number.
2. To select text font instead (using the character set selected by the FONT command, and the present printer vertical line spacing, usually 6 lines per inch) add 8 to the mode number.

## HIRES PRINT SCREEN DUMP

HiRes screen printing is done by an Invokable Module named PKASODMP.INV. The module has entries BPKASODMP for Basic, and PPKASODMP for Pascal. Copy PKASODMP.INV onto your working software diskette for easiest use. From Basic:

```
INVOKE PKASODMP.INV
PERFORM BPKASODMP(@PRNAME$, %MODE)
```

From Pascal: The module must be linked directly from PKASODMP.INV. Define it and use it as follows:

```
PROCEDURE PPKASODMP(VAR PRNAME:STRING;
MODE:INTEGER); EXTERNAL;
```

```
PPKASODMP(PRNAME, MODE);
```

PRNAME is a string containing the name of the printer device (e.g. .PRINTER.)

## COLORS PRINTED IN GRAY...

Screen colors are printed using gray levels composed of dot patterns. The number of distinguishable levels (patterns) depends on the plotting size selected, and on your printer's graphic dot format. To experiment with the sizes and styles on your printer, RUN PRINTPIX on the PKASO /// disk.

The smaller formats of HiRes print do not have sufficient resolution to directly print all the information on the HiRes screen. These are printed with a reduced number of gray levels, combining groups of color codes. A statistical option is provided which may give a more complete gray scale using texturing. Experiment with these formats to find the best one for your graphic application.

There are a large number of options to choose from in requesting a HiRes screen print. Here is how to construct the Mode number which describes to the PKASO system the type of print you want.

### HOW IT WAS PLOTTED . . .

First of all, a HiRes picture on Apple III is created in one of 4 formats. It must be printed in the same format. Choose this format number first in constructing Mode:

Format	Description
0	Black & White, 280 x 192 points
1	16 Limited Colors, 280 x 192 points
2	Black & White, 560 x 192 points
3	16 Colors, 140 x 192 points

### HOW YOU WANT TO PRINT IT . . .

Secondly, there are other choices you may make which determine how the print will appear on the page. The default or normal print is a Small, Right-Side-Up print of HiRes page 1. Each of the options is assigned a numeric factor. To select one or more options, ADD ITS FACTOR to the Format number chosen above:

- Add 4 for a rotated picture
- Add 8 for a medium size picture
- Add 16 for a large size picture
- Add 32 to select HiRes screen 2.
- Add 64 to invert blacks, whites & grays.
- Add 128 for statistical gray levels.

The result of this is a number which is used for the Mode in the HiRes command.

For example, to print a 560-point graphics screen in small rotated format,

1. Choose format 2 for 560 mode.
2. Add 4 for a rotated print.
3. The sum is 6 -- use this for MODE

## HORIZONTAL POSITIONING . . .

The HiRes print always starts at the present Left Margin. To change the position of a print to anywhere on the page, simply set the margin accordingly.

## VERTICAL SPACING

If you want space vertically between HiRes prints, then print blank lines. Since no space is automatically inserted, prints may be produced contiguously on the page. This can be used to print a continuous "strip chart" format.

## CANCELLING THE PRINT . . .

To cancel, press ALPHA LOCK once. The system will beep and stop printing. Then push ALPHA LOCK again to toggle it back the way it was.

## GRAY COMMAND

The PKASO system has a built-in capability to print in gray levels. This can be used in conjunction with selection of ribbon color to control color saturation for an unlimited mixing capability. The GRAY command puts the system into HalfTone mode, then each character you send to it will be printed as a Pixel, in one of 16 levels of gray using a random dot pattern.

GRAY Command:	Prints in gray levels
Format:	~U<N>,<G1>,...,<GN>,
Ctrl Char Form:	^RC-U
From Basic:	PRINT "~U";N;",";"; G1;",";...;GN;",";";
From Pascal:	WRITE (PR,'~U',N,' ','', G1,' ','',...;GN,' ','');

The GRAY command sets the line spacing to 6 dots per line and prepares the system to print the next N characters (G1...GN) as gray scale (or color) pixels. N may be any number from 1 through 192. After N characters, the system returns to text mode, but stays in 6-dot spacing, ready for more GRAY commands.

Each character code produces a gray pixel, 5 dots wide by 6 dots high. Values may be between 0 (white) and 15 (black). A good set of character codes for the set of 16 gray levels is:

WHITE	Lt.Gray	Med.Gray	Dk.Gray	BLACK
0	1 2 3 4 5 6 7 8 9	J K L M N	O	

For example, to print a gray scale in Basic:

```
PRINT "~U16,0,1,2,3,4,5,6,7,8,9,J,K,L,M,N,O,"
```

For example, to print a gray scale in BASIC:

```
PRINT CHR$(21);CHR$(16);"0123456789JKLMN/"
```

Epson MX-80 and MX-100 print pixel pairs. Each character code produces two pixels, one above the other. For Epson, the character code is constructed as follows:

$$G = G_{lo} * 16 + G_{hi}$$

To print a gray scale on an Epson printer:

```
PRINT CHR$(21);CHR$(16);:
FOR I = 0 TO 15 : PRINT CHR$(I*16+I) : NEXT
```

## TEXT COMMAND

To clear all graphic modes and return to text spacing, use the following:

TEXT Command:	Returns to text mode, spacing
Format:	~U0,
Ctrl Char Form:	<del>Ctrl</del> U <del>Ctrl</del> @
From Basic:	PRINT "~U0,";
From Pascal:	WRITE (PR, '~U0,');

## **SUPER-RES GRAPHICS**

The SuperRes DOT command lets you send bit graphics information directly to the printer. It prints any text still in the buffer, sets the printer into graphics mode and prepares to print character codes as dot graphics patterns.

Each character is printed as a vertical dot pattern that is one dot wide. Each dot in the pattern is controlled by one bit in the character code. Bit 0 always controls the bottom dot, bit 1 controls the one above it, etc, to conform to normal graphing conventions. The PKASO system takes care of shifting and reversing of bits necessary to produce this format, and also the quirks in handling ETX codes, so this paragraph may supercede the explanation and examples in your printer manual under the graphics option. See the Command Chart for dot formats for your printer.

In designing printer-independent software, you must be aware of the fact that printers from different manufacturers may have 6, 7, or 8-dot patterns for each SuperRes character. If you write software which produces 6-bit graphics, it will work identically on all printer types. You can also use Status Requests (Appendix B) to determine and adjust to the printer's dot format.

This command does not change vertical spacing. Use VS commands to switch in and out of dot graphics spacing if desired.

**DOT COMMAND**

DOT Command:	Enter SuperRes Graphics Mode
Format:	~Z<Nlo>, <Nhi> ,
Ctrl Char Form:	^Z
From Basic:	PRINT "~Z";NLO;",";NHI;",";
From Pascal:	WRITE (PR, '~Z', NLO, ', ', NHI, ', ');

**N** is the length specification, and may be any number between 1 and 65535. **Nlo** and **Nhi** are the two halves of **N**. These are created as follows:

$$NHI = N/256; NLO = N - 256*INT(NHI)$$

After **N** characters, the system switches out of graphics mode. Your printer may also switch out of graphics mode under certain conditions, such as exceeding the right margin. See your Command Chart.

For example, to print a pyramid in Basic, use the following command:

```
PRINT "~Z11,0, ";chr$(1);CHR$(2);CHR$(4);
CHR$(8);CHR$(16);CHR$(32);CHR$(16);
CHR$(8);CHR$(4);CHR$(2);CHR$(1)
```

This prints: ^

## BYTE COMMANDS

These commands create a momentarily "dumb" printing system by allowing you to send characters directly to the printer, as-is, in their binary form. No margin control is performed, font selection is bypassed, and the PKASO Driver does not even watch for commands. All characters printed are sent unmodified to the printer.

### N BYTES . . .

This command instructs the system to allow the next N characters to pass to the printer as-is.

NBYTE Command:	Prints N bytes as-is
Format:	~Y<Nlo>, <Nhi>
Ctrl Char Form:	Ctrl-Y
From Basic:	PRINT "~Y";NLO;",";NHI;",";
From Pascal:	WRITE (PR, '~Y', NLO, ', ', NHI, ', ', '');

N is the length specification, and may be any number between 1 and 65535. Nlo and Nhi are the two halves of N. These are created as follows:

$$NHI = N/256; NLO = N - 256*INT(NHI)$$

## THE BUFFER AS BYTES . . .

When you print a string or string expression in BASIC, all the characters go into a "Buffer" area in Apple /// memory, and then on to the printer. The BBYTE command can be inserted in the string or string expression, and all characters AFTER the command will be printed directly without modification or command processing.

BBYTE Cmd.:	Rest of buffer as-is
Format:	~B
Ctrl Char Form:	^T~B
From Basic:	PRINT "~B"+"Data"
From Pascal:	Not useful

This command is useful in printing command parameters and graphics dot sequences. For example, to send an ESC command followed by control characters to the printer, set the command up as a string with CHR\$(27) as the first character and CHR\$(2) as the second. Then add the control codes, and they will be sent to the printer unmodified.

Remember that PRINT A\$+B\$ prints a single string expression (buffer) but PRINT A\$;B\$ actually prints two buffers. (Pascal breaks all printing into 1-character "buffers" so this command is not applicable.)

## STRING OF BYTES...

This lets you set up bytes using the Options format, then sends the bytes directly to the printer without processing. NOTE that this is a variable length format, in which you use the Options format for as many bytes as you want to send, with a "J" after the last byte to be transmitted.

BYTE Command:	Sends characters AS-IS
Format:	~S<Options>J
Ctrl Char Form:	<sup>CTRL</sup> S (See below)
From Basic:	PRINT "~S";<Options .....>;"J";
From Pascal:	WRITE (PR, '~S', <Options .....>, 'J');

A single ~S command can be used to send from 1 to 254 bytes to the printer.

NOTE: For the Ctrl Char Form of this command, only one character can be transmitted. For example <sup>CTRL</sup>S<sup>CTRL</sup>A.

## ESCAPE COMMAND

The printable version of the escape command now includes the ability to send a variable-length string of <Options>. The ESC ("Escape") command is included to directly operate the modes and settings built into the printer, such as page positioning and enhanced printing.

While most printer control, such as character size, etc., is handled by standard PKASO commands, you may have an application which uses the printer in an unusual way. These are generally accomplished using "Escape Sequences."

ESC Command:	Sends ESC for printer control
Format:	~[<Printer Options>]
Ctrl Char Form:	<sup>Ctrl</sup> [
From Basic:	PRINT "~["; <Options ...>; "]"
From Pascal:	WRITE (PR, '~['; <Options ...>; ']');

You can send Option characters which may not be printable characters as Decimal, Hex, or Literal codes inside the brackets. For example, ~[7] sends the sequence ESC <sup>Ctrl</sup>G.

If the Options are printable, they can be sent after the J. For example, the Basic statement PRINT "~[JG,120,\$" sends an ASCII sequence ESC G,120,\$ which positions an IDS Prism 1" from the left edge of the page.

The <Options> of the ~[ command function like the <Options> of the ~S Command: A command ~[1,2,3] is equivalent to ~[]~S1,2,3].

The Ctrl Char Form, <sup>Ctrl</sup>~[, has no option processing and merely sends an ESC, as in Revision 2.

IT'S NOT NICE TO FOOL MOTHER PKASO...

NOTE: Do not use ESC sequences where you could use PKASO commands instead. For example, if you use the direct ESC command to change a printer's character size, the PKASO driver will change it back again.

In your Command Chart is a summary of the ESC commands you will find most useful. More details about their operation can be found in your printer manual.

## PRINTER TYPE COMMANDS

If you write software to run on more than one variety of printer, these commands let your software ask the user for his printer type, then re-configure the driver "on-the-fly" to match.

PTYPE Command:	Changes Printer Type
Format:	~Q<P>
Ctrl Char Form:	Ctrl-Q
From Basic:	PRINT "~Q";P;",";
From Pascal:	WRITE (PR, '~Q', P, ',', ',');

P is the printer type code, as described in the DCB section.

**NOTE:** This command sets the character size to 0.1" You should make sure other settings are consistent with the printer chosen.

To change the SubType byte, which holds information on printer size and color capability, use the STYPE Command:

STYPE Command:	Changes Printer SubType
Format:	~\<S>
Ctrl Char Form:	Ctrl-\
From Basic:	PRINT "~\";S;",";
From Pascal:	WRITE (PR, '~\'', s, ',', ',');

S is used as follows:

0 - 80 col. B&W	2 - 80 col. Color
1 - 132 col. B&W	3 - 132 col. Color

**SOFTWARE RESET COMMAND**

At any point you have the option to reset all parameters back to the settings given in the DCB.

RESET Command:	Software reset
Format:	~W
Ctrl Char Form:	^TR~W
From Basic:	PRINT "~W";
From Pascal:	WRITE (PR, '~W');

Unlike the OPEN operation which initializes printer settings only the first time it is done, SOFTWARE RESET always puts the driver and interface in the starting state described by the DCB.

**MEMORY USAGE**

The complete PKASO Driver occupies approximately 5000 bytes of memory. The invocable module for HiRes screen prints requires approximately 4000 bytes.

**ASSEMBLY LANGUAGE PROGRAMMING INTERFACE**

The following briefly describes the PKASO direct I/O interface for diagnostic and development purposes. Direct programming of I/O devices requires an extensive understanding of the SOS environment, I/O drivers, and Apple /// hardware. All accesses to the PKASO interface must be done in 1 MHz mode. All guidelines regarding use of the I/O provisions of Apple /// must be observed.

## STATUS INPUT FROM THE PRINTER...

The printer status register is located at hexadecimal address

**\$C080 + NO** (N is the PKASO slot number)

A byte read from this address is arranged as follows:

Bit:	7	6	5	4
"1" =	Int. Ready	Paper Out	On Line	
"0" =	No Int. Busy	Paper OK	Local	

Bits 3,2,1,0 are not used.

Bit 7 is used to denote an active interrupt coming from the PKASO, and may be used by an interrupt service routine to identify a PKASO interrupt. Bits 6 and 4 indicate the status of the printer. If bits 7 thru 4 are "0101", the printer is ready to accept data.

## DATA OUTPUT TO THE PRINTER...

The locations associated with output of a data byte are the following:

**\$C800 Strobe Down, Interrupts off, Clear IRQ**  
**\$C801 Strobe Up, Interrupts off, Clear IRQ**  
**\$C802 Strobe Down, Enable IRQ on Ack.**  
**\$C803 Strobe Up, Enable IRQ on Ack.**

A byte is sent by writing it to these addresses so as to toggle the strobe in "up-down-up" sequence.

## HINTS

### USE WITH POPULAR LANGUAGES AND SOFTWARE

**APPLE II EMULATION MODE** - Normal operation as Apple II printer system. See your Apple II PKASO manual.

**APPLEWRITER ///** - Full use of PKASO Commands from within the text. Use the ~ commands to set character size, etc.

**APPLE Business Graphics** - The PKASO system can provide full printing capability for this package. See the next page.

**APPLE Business Basic** - Follow the instructions and examples in this manual. RUN PRINTPIX to dump HiRes images.

**APPLE PASCAL** - Follow the instructions and examples in this manual. X(ecute PRINTPIX to dump HiRes images

**VISICALC ///** - Send PKASO commands in the printing setup string, or include them within labels on your sheet. Example: Use "~E2," to select compressed printing.

**WORD PROCESSORS IN GENERAL** - Consult the manual for the proper procedure for installing a driver, then include ~ commands directly in the text. If the package is inextricably attached to its own direct printer control software, install MINI.DRIVER.

## APPLE BUSINESS GRAPHICS GRAPHIC PRINT PROCEDURE

With Apple ///, any program which needs to use the printer must have the PKASO Driver installed or its program diskette. So you must go through the procedure which we describe in great detail in Appendix D to put the PKASO Driver on your Business Graphics Diskette. Then you are ready to print any picture you create with Business Graphics.

1. Boot your Business Graphics diskette (with PKASO Driver installed).
2. Create the plot on the screen.
3. Save it on disk.
4. Quit.
5. Insert the PKASO /// diskette in the built-in drive.
6. Type "X". Pascal will ask "What File?".
7. Type "/PKASODRIVER/PRINTPIX".
8. Follow the program's instructions to load and print images.
9. When PRINTPIX asks you for the name of your picture file, enter the COMPLETE form of the filename, including the device name (.D1, etc.), and the filename with the suffix .S.DATA added.

EXAMPLE: .D2/graph1.S.DATA

10. If you created the image with Business Graphics mode 2, print it with PRINTPIX mode 3; BG mode 1 becomes PRINTPIX mode 2; and mode 0 stays the same.

## APPENDIX A

### PKASO /// QUICK COMMAND REFERENCE

NAME	PKASO CMD.	ARGS	FUNCTION
----	----	----	-----
			None.
			None.
BBYTE	~B		Buffer BYTE mode.
MARGIN	~C	<LM,RM,WM>	Set margins.
WIDTH	~D	<Width>	Set line width
SIZE	~E	<Size Code>	Set char size.
TABSET	~F	<N,T1..Tn>	Set tab stops
BELL	~G		Ring printer bell
BS	~H		Backspace
HT	~I		Go to next tab stop
LF	~J		Line feed
			None.
FF	~L		Form feed
CR	~M		Carriage return
COLOR	~N	<ClrCode>	Select Ribbon
LEAD-IN	~O	<NewCode>	Change ~
TPRINT	~P	<Mode>	Text screen dump.
PTYPE	~Q	<Prtype>	Printer type
FONT	~R	<Chmode>	Character set mode
BYTE	~S	<Options>	String of BYTEs
VS	~T	<VSmode>	Vertical Spacing
GRAY	~U	<N,G1..Gn>	Gray Scale mode
BDOT	~V	<Buffer>	Single Buffer Graphics
RESET	~W		Software Reset
			None.
NBYTE	~Y	<Nlo,Nhi..>	N Char. BYTE mode.
NDOT	~Z	<Nlo,Nhi..>	N Char. Bit Graphics.
ESC	~[	<Options>	Send ESC Sequences
STYPE	~\	<SubType>	Width, Color, etc.
			None.
			None.
			None.

HiRes Print: INVOKE PKASODMP.INV  
PERFORM (@PR\$,%MODE)

## APPENDIX B STATUS & CONTROL REQUESTS

### A. STATUS REQUESTS

The PKASO Driver provides status requests which allow the user to determine the characteristics and condition of the printer driver and printer.

The Status Request is a standard SOS operation, however different interpreters provide different means of performing status requests. Business basic uses the invocable REQUEST.INV to get the status information. Pascal programs use the intrinsic procedure UNITSTATUS.

FROM BUSINESS BASIC . . .

```

INVOKE "REQUEST.INV"
OPEN #1,".PRINTER"
PERFORM STATUS (%STATCODE,@STATLST$)" .PRINTER"

```

STATCODE contains your status request code.  
STATLST\$ gets loaded with status information.  
To get the N'th status byte:

```
STAT = ASC (MID$ (STATLST$,N,1) )
```

FROM PASCAL . . .

```
UNITSTATUS (UNITNUM, STATLST, REQCODE);
```

For the printer, UNITNUM should be 6. STATLST is a variable whose structure is dependant on the status requested. (See below.) REQCODE is a 16 bit value which determines the type of status call requested. For instructions see Appendix H of the Standard Device Drivers manual.

**STATUS REQUEST NUMBERS . . .**

**3 - Get printer status.**

This call returns 5 bytes:

- 0 - Printer type (See the DCB)
- 1 - Printer mode byte 1
- 2 - Printer mode byte 2
- 3 - Printer vertical spacing:
  - 0 = Standard text (6 LPI)
  - 6 = Graphics spacing (6 dots)
  - 7 = Graphics spacing (7 dots)
  - 8 = Graphics spacing (8 dots)
- 4 - Printer on-line & paper status:
  - 0 = Ready to print
  - 1 = Off line
  - 2 = Paper out
  - 3 = Off line and paper out

**4 - Margins and sizes**

This call returns 5 bytes:

- 0 - Width of page in characters
- 1 - Left margin
- 2 - Right margin
- 3 - Wrap margin
- 4 - Character size code.

**5 - Tab information**

This call returns 18 bytes:

- 0 - The number of tab stops now set
- 1...16 - The tab stops
- 17 - The next tab stop to be used

**6 - The Command Lead-in Byte**

Returns present lead-in (~, etc.)

**7 - The Printer Sub-type Byte**

- 0 - 80 col, B&W            2 - 80 col, Color
- 1 - 132 col, B&W        3 - 132 col, Color

## B. CONTROL REQUESTS

The PKASO driver provides a selection of control request functions to set driver parameters.

### FROM BUSINESS BASIC . . .

Perform the Control Procedure in the REQUEST.INV invokable:

```
CONTROL (%CTRLCODE, @CTRLLIST$) ".PRINTER"
```

%CTRLCODE contains the number of the control function requested. CTRLLIST\$ contains the control information to be set.

### FROM PASCAL . . .

Use the UNITSTATUS function as described under Status Calls, above, setting the STAT-OR-CTRL field to 1 instead of 0. See Appendix H of the Standard device drivers manual.

## CONTROL REQUEST NUMBERS . . .

### REVISION 3

#### 3 Flush Buffer.

(no control list) Causes any printing in progress to finish.  
(For example, to prepare for a switch to some graphics mode.)  
This dumps the Apple screen font buffer if active. Should be called before doing your own graphics.

#### 4 Restore Size

(no control list) Sets the printer character size to that given by the last SIZE command.

#### 5 Set vertical line-feed spacing.

The control list is a 1-byte size code:

- 0 - Text spacing, 6 LPI
- 1 - HiRes print spacing for your printer. (6 or 8 dots)
- 2 - Gray Scale spacing, 6 dots.
- 3 - 7 dot spacing

#### 6 Turn off dot graphics mode.

(no control list) Also restores char size.

#### 7 Turn on dot graphics mode.

(no control list)

#### 8 Send graphics length in chars.

The control list is the 16-bit length, low byte first.

#### 9 Set margins and sizes.

The control list is 5 bytes long:

- 0 - Line Width    2 - Right Margin    4 - Character Size Code
- 1 - Left Margin    3 - Wrap Margin

#### 10 Set tabs.

The control list is 18 bytes long: Byte 0 - Number of tabs set, Bytes 1...16 - Up to 16 tab stops, Byte 17 - Next stop to be used

#### 11 Set Lead-in Character.

The control list is 1 byte - the new lead-in character.

#### 12 Set Printer Sub-Type Byte.

The Control list is 1 byte -    0 = 80 col B&W    2 = 80 col Color  
   1 = 132 col B&W    3 = 132 col Color

## **APPENDIX C THE MINI DRIVER**

In addition to the PKASO.DRIVER described in this manual, the PKASO /// diskette also contains a mini-driver which occupies only 512 bytes of storage. It can be installed using the same procedures given in Appendix D, with the file name MINI.DRIVER. The printer Slot Number is the only settable option for this driver.

It has one and only one function - to transfer bytes absolutely unaltered to the printer. There are no commands, no options, no print sizes, no margins. Everything printed will be transmitted to the printer without processing.

MINI.DRIVER does not support the PKASODMP.INV or PKASOCLR.INV graphics invokables.

MINI.DRIVER is intended for the system application where programming has already been done to handle a printer, or where special requirements exist. For example, the full PKASO driver will not fit on a standard System Utilities diskette, so MINI.DRIVER can be used instead to provide printing of simple text.

## APPENDIX D INSTALLATION of the PKASO Driver

Each software package or language system which will directly use the printer must have the PKASO Driver "installed" on its diskette. We include the procedure here with appropriate filenames and terminology for the PKASO Driver.

It is always best to use a COPY of your diskette for the driver installation. The PKASO interface card should be installed in SLOT 1.

1. Boot your Apple /// SOS Utilities Diskette.
2. Select the System Configuration Program (SCP)
3. Select the READ A DRIVER FILE option, and use SOS.DRIVER as a file name.
4. Place your software diskette (the one which is to get the new the PKASO Driver) in the built-in drive and hit 'RETURN'. This loads your diskette's present drivers into Apple ///'s memory.

Note: Press RETURN only after entering the driver filename! As soon as you see the list of drivers, press ESCAPE to return to the SCP menu. (If you pressed RETURN here, extra copies of the driver file would get loaded.)

5. You will see a list of the present drivers. If you see:

.PRINTER .QUME ,or .SILENTYPE

then select the DELETE A DRIVER option to remove them.

6. Select READ A DRIVER FILE again to pull the PKASO Driver into memory.

7. Insert your PKASO /// Driver diskette in the built-in drive. Use the filename PKASO.DRIVER. Remember, press RETURN only once after the file name, then press ESCAPE to go back to the main menu.
8. If SCP alerts you that there is insufficient memory for the driver at this point, then the problem is actually that SCP hasn't released the space yet which was occupied by the printer drivers you deleted. If this happens, go to Step 11, below. You will generate a system with no printer drivers, then exit SCP. THEN GO BACK TO STEP 1. This time through, SCP will have plenty of space to work with. (If not, then MINI.DRIVER must be used.)
9. Select EDIT DRIVER PARAMETERS, then select the CONFIGURATION BLOCK option. Referring to your PKASO /// manual, determine the code for your type of printer. Select byte 0, the printer type byte of the Driver Configuration Block, and change it to 05, the code for your printer. Next select byte A, the printer Subtype, and change it to the proper value as specified by byte A in the DCB. (See Appendix E.)
10. Returning to the SCP menu, select the CHANGE SYSTEM PARAMETERS option, then select PERIPHERAL SLOT ASSIGNMENTS. The new driver will appear as .PRINTER with a slot number of "?". Change this to slot 1. Return to the SCP menu.
11. Select the GENERATE NEW SYSTEM option. Insert your software diskette again in the built-in drive and write your new driver file onto the diskette as SOS.DRIVER. SCP may ask you to verify that you want to unlock and overwrite the old SOS.DRIVER file. Type "Y".
12. Return to the SCP menu and QUIT. Then return to the main menu and QUIT. Your diskette is now ready for use.

**APPENDIX E - THE DCB  
THE DRIVER CONFIGURATION BLOCK**

BYTE	DESCRIPTION / INITIALIZED VALUE
0	Printer type / set this for your printer 00 - Epson MX-82, Type-II, Old MX-100 01 - Epson Type III, Graftrax, Graftrax-Plus 02 - Epson, MX-70 03 - Centronics 739 04 - Okidata, test models 05 - IDS Prism 06 - Okidata w/Okigraph 07 - NEC PC8023, C.Itoh 8510
1	Status display flash rate / F8 (Hex)
2	Line width in characters / 255 (max)
3	Left margin in characters / 0 (none)
4	Right margin in characters / 0 (none)
5	Paragraph margin in chars / 0 (none)
6	Printing Size Code / 0 (10 CPI) 08 - 5 CPI      00 - 10 CPI 09 - 6 CPI      01 - 12 CPI 0A - 8.2 CPI    02 - 14.5 CPI 03 - Proportional   0B - Prop. wide
7	Linefeed mode / 1 (software linefeed) 00 - Printer produces all linefeeds. 01 - Software produces linefeeds.
8	Printing Character Set / 0 (standard) 00 - Printer's standard characters 01 - Apple screen character font 05 - Screen font, inverse mode
9	Command Lead-in Byte/ 7E (~)
A	Printer Sub-type / 0 (80 Col., B&W) 00 - 80 col. B&W      02 - 80 col. Color 01 - 132 col. B&W    03 - 132 col. Color

**APPENDIX F  
PRINTERS SUPPORTED**

Parallel version of the following printers:  
(Switch the printer's auto-line feed switch OFF)

Mfgr Name	Graphics Models	Text Only Models	PKASO Card
C. Itoh	8510 Pro/Writer		NE12
Centronics	122, 739 SoftFont feature not supported.		CE12
Epson	MX-70, 80, 100 For MX80, specify Type III or Graftrax Plus. Graftrax-80 and Type II (old MX-100) also supported.		EP12
IDS	Prism Specify Graphics and Color options with Process color ribbon.		ID12
NEC	PC8023A-C		NE12
Okidata	82A/83A Specify the Okigraph I graphics option.	80,82,83,84	OK12

## **LIMITED WARRANTY**

Software and documentation connected with this product are guaranteed to reach the purchaser in readable condition, excepting those failures which, in the sole opinion of Interactive Structures, Inc., were the result of misuse or abuse of the product. Defective products will be promptly replaced if Interactive Structures is notified within 90 days of purchase of this product.

## **LIABILITIES DISCLAIMER**

NO OTHER WARRANTIES, EXPRESS OR IMPLIED, ARE EXTENDED BY INTERACTIVE STRUCTURES IN CONNECTION WITH THIS PRODUCT AND ITS DOCUMENTATION. INTERACTIVE STRUCTURES IS NOT AND WILL NOT BE RESPONSIBLE OR LIABLE FOR DAMAGES WITH RESPECT TO THE PRODUCT DESCRIBED IN THIS MANUAL, ITS PERFORMANCE, MERCHANTABILITY, OR FITNESS FOR ANY PARTICULAR PURPOSE.

In using this product, the purchaser assumes the entire responsibility and risk as to its use and any incidental or consequential damages which might result.

## **IMPROVEMENTS**

Interactive Structures reserves the right to make improvements in the product described in this manual at any time and without notice.

# COMMAND CHART

Centronics 122, 739

Printer Type 3

## USEFUL ESCAPE COMMANDS

- ~[1] Space left 1 dot space.
- ~[6] Space left 6 dot spaces.
- ~[10] Full line-feed, reverse direction
- ~[28] One-half line-feed forward.
- ~[30] One-half line-feed reverse.

----

Command codes are not printable so they should be placed inside the brackets.

See your Centronics manual for functional details.

# COMMAND CHART

Centronics 122, 739

Printer Type 3

**DOT GRAPHICS (LF = .0833")**

Dot format: 6 dots per character

Dot spacing: 72 dots/" vert.  
84 dots/" horiz.

Page format: 8" x 10" = 672 x 720 dots

**GRAY SCALE (LF = .0833", 6 dots)**

Pixel size: 5 dots wide x 6 dots high

Coding: Each byte prints 1 pixel

## SOFT-FONT

The 6-dot graphic format is not amenable to the SoftFont system.

**TEXT PRINTING (LF = .1666", 6 LPI)**

Sizes: .1", .2" (.06, .12 on 739)

Size Command: Takes effect immediately.

## TEXT SCREEN PRINT

Standard: Printer font & spacing

Options: Set char. size to control screen print width.

# COMMAND CHART

EPSON MX-70  
Printer Type 2

## USEFUL ESCAPE COMMANDS

- ~[] 2 Set 6 lines per inch vertical spacing.
- ~[] A \* Set line spacing in 1/72" increments.
- ~[] C \* Set form length in 1/6" increments.
- ~[] K \* 60 dots per inch graphics lead-in.

----

Command codes are printable: e.g. "~[]A" or "~[]`A]"

\* - These are followed by parameters, usually not printable. For these commands place the function codes and options inside the brackets:  
e.g. PRINT "~[]`C,12]"

See your Epson manual for function details.

# COMMAND CHART

EPSON MX-70  
Printer Type 2

**DOT GRAPHICS (LF = .1065", 8 dots)**

Dot format: 8 dots per character

Dot spacing: 72 dots/" printhead spacing  
60 dots/" horiz.

Page format: 8" x 10" = 480 x 720 dots  
13" x 10" = 780 x 720 dots

**GRAY SCALE (LF = .0787", 6 dots)**

Pixel size: 5 dots wide x 6 dots high

Coding: Each byte prints 1 pixel

**SOFT-FONT (LF = .1666", 6 LPI)**

Options: Inverse

Char. Width: .1333"

**TEXT PRINTING (LF = .1666", 6 LPI)**

Sizes: .1", .2

Size Command: Takes effect immediately.

**TEXT SCREEN PRINT**

Standard: Screen font & spacing

Options: Inverse, Text font & spacing

**COMMAND CHART**  
**EPSON with GRAFTRAX**  
**Printer Type 1**

**USEFUL ESCAPE COMMANDS**

- ~[] 0 Set 8 lines per inch vertical spacing.
- ~[] 2 Set 6 lines per inch vertical spacing.
- ~[] 3 \* Set line spacing in 1/216" increments.
- ~[] 4 Turn on Italics character set.
- ~[] 5 Turn off Italics character set.
- ~[] @ Reset printer to the power up state.
- ~[] A \* Set line spacing in 1/72" increments.
- ~[] C \* Set form length in 1/6" increments.
- ~[] D \* Set internal printer tabs.
- ~[] E Turn on emphasized mode.
- ~[] F Turn off emphasized mode.
- ~[] G Turn on double strike mode.
- ~[] H Turn off double strike mode.
- ~[] K \* 60 dots per inch graphics lead-in.
- ~[] L \* 120 dots per inch graphics lead-in.
- ~[] N \* Set auto skip over top of form mode.
- ~[] O Turn off auto skip over top of form mode.
- ~[] Q \* Set the printer width in columns.

----

Command codes are printable: e.g. "~[]A" or "~[]A]

\* - These are followed by parameters, usually not printable. For these commands place your command code and options inside the brackets:  
e.g. PRINT "~[]C,12]"

See your GRAFTRAX manual for function details.

**COMMAND CHART**  
**EPSON with GRAFTRAX**  
**Printer Type 1**

**DOT GRAPHICS (LF = .1065", 8 dots)**

**Dot format: 8 dots per character**

**Dot spacing: 72 dots/" printhead spacing**  
**216 dots/" vert. addressing**  
**60 dots/" horiz.**  
**120 dots/" horiz. compressed**

**Page format: 8" x 10" = 960 x 2160 dots**  
**13" x 10" = 1560 x 2160 dots**

**GRAY SCALE (LF = .0787", 6 dots)**

**Pixel size: 5 dots wide x 3 dots high**

**Coding: Each byte prints 2 pixels**  
**(5 wide x 6 high rectangle)**

**SOFT-FONT (LF = .1666", 6 LPI)**

**Options: Compressed, Inverse**

**Char. Width: .1333" (.0666" compressed)**

**TEXT PRINTING (LF = .1666", 6 LPI)**

**Sizes: .06", .1, .12, .2**

**Size Command: Takes effect immediately**

**Emphasize enlarges .06" and .12" chars.**

**TEXT SCREEN PRINT**

**Standard: Screen font & spacing**

**Options: Inverse, Text font & spacing**

**COMMAND CHART**  
EPSON Type II, 82, MX-100  
Printer Type 0

**USEFUL ESCAPE COMMANDS**

- ~[] 0 Set 8 lines per inch vertical spacing.
- ~[] 2 Set 6 lines per inch vertical spacing.
- ~[] A \* Set line spacing in 1/72" increments.
- ~[] C \* Set form length in 1/6" increments.
- ~[] D \* Set internal printer tabs.
- ~[] E Turn on emphasized mode.
- ~[] F Turn off emphasized mode.
- ~[] G Turn on double strike mode.
- ~[] H Turn off double strike mode.
- ~[] K \* 60 dots per inch graphics lead-in.
- ~[] L \* 120 dots per inch graphics lead-in.
- ~[] N \* Set auto skip over top of form mode.
- ~[] O Turn off auto skip over top of form mode.
- ~[] Q \* Set the printer width in columns.

----

Command codes are printable:

e.g. PRINT "~[]A" or PRINT "~[]\A"

\* - These are followed by parameters, usually not printable. For these commands place the command code and options inside the brackets:

e.g. PRINT "~[]\C,12]"

See your Epson manual for function details.

## COMMAND CHART

EPSON Type II, 82, MX-100

(Without GRAFTRAX)

Printer Type 0

**DOT GRAPHICS (LF = .1065", 8 dots)**

Dot format: 8 dots per character

Dot spacing: 72 dots/" vert.  
60 dots/" horiz.  
120 dots/" horiz. compressed

Page format: 8" x 10" = 960 x 720 dots  
13" x 10" = 1560 x 720 dots

**GRAY SCALE (LF = .0787", 6 dots)**

Pixel size: 5 dots wide x 3 dots high

Coding: Each byte prints 2 pixels  
(5 wide x 6 high rectangle)

**SOFT-FONT (LF = .1666", 6 LPI)**

Options: Compressed, Inverse

Char. Width: .1333" (.0666" compressed)

**TEXT PRINTING (LF = .1666", 6 LPI)**

Sizes: .06", .082, .1, .12, .16, .2

Size Command: Takes effect immediately if just  
a switch in or out of expanded, e.g. .1" to .2".  
Takes effect BEFORE the line otherwise.

**TEXT SCREEN PRINT**

Standard: Screen font & spacing

Options: Inverse, Text font & spacing

# COMMAND CHART

IDS Prism Printers  
Printer Type 5

## USEFUL ESCAPE COMMANDS

- ~[] B Set Vertical Adv-1
- ~[] C Set Vertical Adv-2
- ~[] D Set Vertical Adv-3
- ~[] E Set printer's vertical tabs
- ~[] G Set absolute Horizontal Position
- ~[] H Set absolute Vertical Position
- ~[] L Set form size
- ~[] M Set absolute line position
- ~[] N Set horizontal character position
- ~[] P Set intercharacter spacing
- ~[] Q Set ribbon color on color Prisms
- ~[] R Set character font, Sprint Mode only

Command codes are printable. Most commands require parameters. e.g. "~[]B,8,\$" to select 6 lines per inch. See your IDS manual for formats & details

## USEFUL BYTE COMMANDS

- ~S4] Turn on Text Justify mode
- ~S5] Turn off Text Justify mode
- ~S16] Turn on Proportional Spacing mode
- ~S6] Turn off Proportional Spacing mode

# COMMAND CHART

IDS Prism Printers  
Printer Type 5

## DOT GRAPHICS (LF = .0625")

Dot format: 7 dots printed per character  
HiRes print uses 6 dot spacing

Dot spacing: 84 dots/" vert.  
84 dots/" horiz.

Page format: 8" x 10" = 672 x 840 dots  
13" x 10" = 1092 x 840 dots

## GRAY SCALE (LF = .0625", 6 dots)

Pixel size: 5 dots wide x 6 dots high

Coding: Each byte prints 1 pixel.

## SOFT-FONT (LF = .1666", 6 LPI)

Options: Inverse

Char. Width: .0952"

## TEXT PRINTING (LF = .1666", 6 LPI)

Sizes: 0.6", .1, .12, .2, Proportional

Size Command: Takes effect immediately when used.

## TEXT SCREEN PRINT

Standard: Screen font & spacing

Options: SoftFont with 7-dot character  
height.

# COMMAND CHART

NEC PC8023 & C.Itoh 8510

Printer Type 7

## USEFUL ESCAPE COMMANDS

- ~[ f      Select forward line-feed direction
- ~[ r      Select Reverse line-feed direction
- ~[ \$      Select alphanumeric character set
- ~[ #      Select internal graphics symbols
- ~[ !      Select enhanced mode, text & graphics
- ~[ "      Clear enhanced mode
- ~[ S \*    Enter bit graphics mode
- ~[ A      Set vertical spacing to 6 LPI
- ~[ B      Set vertical spacing to 8 LPI
- ~[ T <N> Set vertical spacing to N/144"
- ~[ X      Turn on Underline Mode
- ~[ Y      Turn off Underline Mode

----

Most command codes are printable: e.g.

"~[A" to select 6 LPI spacing.

<N> will usually be a printable code.

For example to set 12/144" vertical spacing:

PRINT "~[T12"

\* - Followed by parameters and data, not necessarily printable. For these commands place the function code and options inside the brackets:

e.g. "~[`S,`0,`0,`0,`5,1,2,4,8,16]"...

Hint: for bit graphics use the PKASO SuperRes command instead.

See your printer manual for function details.

**COMMAND CHART**  
NEC PC-8023 & C.Itoh 8510  
Printer Type 7

**DOT GRAPHICS (LF = .1111", 8 dots)**

**Dot format: 8 dots per character**

**Dot spacing: 72 dots/" printhead spacing  
144 dots/" vert. addressing  
81 dots/" horiz.  
133 dots/" horiz. compressed**

**Page format: 8" x 10" = 1064 x 1440 dots**

**GRAY SCALE (LF = .1042", 6 dots)**

**Pixel size: 5 dots wide x 6 dots high**

**Coding: Each byte prints 1 pixel**

**SOFT-FONT (LF = .1666", 6 LPI)**

**Options: Compressed, Inverse**

**Char. Width: .1" (.06" compressed)**

**TEXT PRINTING (LF = .1666", 6 LPI)**

**Sizes: 0.06", .1, .12, .2**

**Size Command: Takes effect immediately**

**TEXT SCREEN PRINT**

**Standard: Screen font & spacing**

**Options: Inverse, Text font & spacing**

**COMMAND CHART**  
Okidata 82A and 83A Printers with  
Dot Addressable Graphics  
Printer Type 6

**USEFUL ESCAPE COMMANDS**

- ~[] M \* Direct line skip
- ~[] F \* Set number of lines per page
- ~[] 5 Reset top of form
- ~[] 6 Select 6 lines per inch text
- ~[] 8 Select 8 lines per inch text
- ~[] % 9 <N> Move paper up N/144"  
(Use in text mode only)

----  
Command codes are printable:  
e.g. "~[]6" or "~[]'6]"

\* - These are followed by parameters,  
e.g. "~[]F22" sets 22 lines per page.

See your Okidata manuals for more details.

**COMMAND CHART**  
Okidata 82A and 83A Printers with  
Dot Addressable Graphics  
Printer Type 6

**DOT GRAPHICS (LF = .0833")**

Dot format: 7 dots printed per character  
HiRes print uses 6 dot spacing

Dot spacing: 66 dots/" printhead spacing  
144 dots/" vert. addressing  
60 dots/" horiz.

(Note: 99 dots/" horiz. in 16.5 CPI Size.  
Use this to change graphics, SoftFont width.)

Page format: 8" x 10" = 480 x 1440 dots  
13" x 10" = 816 x 1440 dots

**GRAY SCALE (LF = .0833", 6 dots)**

Pixel size: 5 dots wide x 6 dots high

Coding: Each byte prints 1 pixel

**SOFT-FONT (LF = .1666", 6 LPI)**

Options: Inverse, also use the 16.5 CPI  
character Size for a 40%  
compression effect.

Char. Width: .1333" (.0808" at 16.5 CPI)

**TEXT PRINTING (LF = .1666", 6 LPI)**

Sizes: .06", .1, .12, .2

Size Command takes effect: Start of  
next line if just a switch in or out of  
expanded mode. Immediately otherwise

**TEXT SCREEN PRINT**

Standard: Present text font & spacing

Options: SoftFont with 7-dot character height.  
(Use 16.5 CPI char size to  
compress print)

