PowerComm

User Manual

By Logitek Systems

*"Bringing Innovative Solutions
to the Pick Community"*

PowerComm™ User Manual

By Logitek Systems

Copyright Notice & Trademarks

Logitek Systems, Carlsbad, CA 92009 USA

Office	☎	(760) 607-6000
Fax	🖹	(760) 607-6001
E-Mail	✉	info1@logitek.com
Web	💻	www.logitek.com

Disclaimer & Limitation of Liability

Logitek Systems make no representations or warranties with respect to this product or the contents hereof. This product (including both software and the documentation) is licensed **as is** and without any express warranty of any nature. **ANY IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE IS HEREBY DISCLAIMED.**

In no event shall Logitek Systems be liable for consequential damages, even if Logitek Systems has been advised of the possibility of such damages. Further, Logitek Systems reserves the right to revise this publication and to make changes from time to time in the contents hereof without obligation of Logitek Systems to notify any person or organization of such revision or changes.

Manual Information

Version 2.1		February 4, 1995
Version 2.1a	Revision 1	July 21, 1995
Version 2.2		April 5, 1996
Version 2.3		January 3, 1997
Version 2.3	Reprint	April 6, 1998
Version 2.3	Reprint	May 1, 2001

Table of Contents Summary

Table of Contents

Appendices

About This Manual

This *User Manual* covers the installation and use of PowerComm. In addition, reference information about commands, API's and menu selections are also documented.

New users of PowerComm are reminded to complete the product registration card enclosed with the diskettes and return it to Logitek Systems. This will enable us to provide you with technical support and keep you informed of further development and releases.

In addition to this manual, please refer to the *Release Notes* that came with PowerComm as they contain last minute or other important information regarding PowerComm.

What is PowerComm?

PowerComm is a high performance data communications software package for systems running the Pick Operating System. PowerComm provides data communications services by providing full TCL command and user mode interfaces for maximum flexibility.

TCL Command Mode

For ease of use and flexibility, all of the PowerComm file transfer commands are available at TCL. Using the standard TCL command line interface, all options and parameters are easily incorporated into programs.

- Full TCL command line functionality using the standard TCL interface convention makes it quick and easy to use and remember.
- On-line help for all TCL commands provides quick reference when you need it.

Direct Terminal Mode

PowerComm includes a direct terminal mode that allows you to communicate to another system as a terminal on that system.

- User-definable alternate break key changeable at runtime for flexible break key handling.
- User-definable alternate escape key at runtime also provides for a flexible direct mode exit key.
- Capture mode provides storage and playback of your entire direct terminal mode.
- Script mode can automate log on sequences for quick and easy dial ups.

BASIC Applications Program Interface

The BASIC API is an application programming interface to the communications software. This interface is fully documented and is accessed by calling subroutines or user exits.

- TCL command interface is used with the BASIC execute statement and all TCL commands return error message numbers for complete testing in BASIC.
- User mode interface provides programmers maximum flexibility for unique end-user requirements.
- Fully documented examples can be used as a template to start using the API routines
- A new PowerComm Library (PCL) module has been added with full source code. Here is an excerpt from the PowerComm Library document:

> The PowerComm Communications Library (PCL) is a separate collection of BASIC programs layered on top of PowerComm that may be used with your code to make it easier to interface with the communications functions of PowerComm.
>
> Pick to Pick using the Remote Server
>
> If your intention is to communicate with another Pick system in a fully automated configuration, you should look at the remote server application program interface (RSAPI) to accomplish this. By defining some values in some control items and writing a subroutine or two, you can have a reliable method of automating Pick to Pick data communications. There are quite a few examples of using the RSAPI you can look at to get started.
>
> Other Systems
>
> If you are going to be communicating to another type of system, you can still take advantage of the low level communications functions that the RSAPI relies on. You will want to look at PCL.PORT.IO.SUB for handling the port i/o, PCL.MODEM.IO.SUB for a complete modem interface routine and some of the other support routines like PCL.CP.UTY.SUB for interfaces to the Start, Attach, Detach and Kill port routines.
>
> Also, take a look at PCL.SCRIPT.SUB for common routines you may need to help automate getting 'logged on' after a successful modem connection.

Manual Contents

The PowerComm Documentation is organized into five parts. Each part documents various aspects of PowerComm as summarized below:

Part 1 Installation Guide

This part contains several chapters that pertain to initial product installation and configuration.

- Chapter 1, *Before You Begin*, briefly describes some of the terminology and concepts you will need to install and use PowerComm. It also describes how to use the menus and entry screens.
- Chapter 2, *Product Installation*, documents the installation of PowerComm by release type.
- Chapter 3, *Initial Configuration*, covers the initial configuration of PowerComm parameters.

Part 2 User Guide

This part contains several chapters that describe how to use PowerComm.

- Chapter 4, *Using TCL Commands*, provides information and examples on how to use some of the TCL commands with various options.
- Chapter 5, *Technical Information*, provides more detailed information about PowerComm TCL commands. It also contains information about communications ports and more tips for using PowerComm.
- Chapter 6, *Troubleshooting*, contains information helpful in finding the cause and solution to problems while using PowerComm.
- Chapter 7, *Questions and Answers*, contains many common questions and answers about PowerComm.

Part 3 User Reference

This part includes several chapters that contain reference information about PowerComm including TCL commands, menus and screens.

- Chapter 8, *TCL Commands Reference*, provides the description, syntax and parameters for all PowerComm TCL commands.
- Chapter 9, *Menu Selections Reference*, documents the menus and entry screens in PowerComm.

Part 4 Programmer Reference

This part includes several chapters that pertain to application programming with PowerComm.

- Chapter 10, *Using PowerComm in Applications*, describes how to use PowerComm in your application programs.
- Chapter 11, *User Mode API Reference*, documents the programs for accessing the user mode interface capabilities of PowerComm.

Appendices

This part contains the following appendices:

- Appendix A, *PowerComm's Script Language*, documents PowerComm's built in script language commands and how they are used.
- Appendix B, *TCL Command Message Control Table*, documents the additional options that may be used with PowerComm TCL commands to suppress or add information to the execution of these commands.
- Appendix C, *PowerComm Specifications*, documents the PowerComm specifications.
- Appendix D, *Removing PowerComm*, documents how to remove PowerComm from your system by release type.
- Appendix E, *Error Messages*, contains a numerical listing of all of PowerComm error messages.
- Appendix F, *File Contents*, documents some of the software's data files and what they contain or how they are used.

Documentation Conventions

This section documents the conventions, special symbols, and typefaces used throughout the documentation.

General Conventions

The following text formatting in the documentation represent different types of information.

Formatting	Description
Prompt fields	These are screen prompts or fields in entry screens.

`screen display`
`monospaced font`
	These shaded boxes represent what is displayed on the screen.
	In screens, this bold and underlined typeface represents what you must type in.

`program listing`
`text`
`text`
	These shaded boxes represent computer program listings.

`command parameters (options)`	This typeface represents text that you type for examples.

Keyboard Conventions

The following is a list of keyboard conventions used:

- Key names match the names shown on most keyboards and appear in uppercase. For example, the shift key appears as SHIFT, the control key as CTRL.
- The return key and the enter key on Pick perform the same action. In this documentation, "Press <ENTER**>**" means that you can press either ENTER or RETURN.
- A plus sign "+" used between two key names indicates that you must press both keys at the same time. For example, "Press SHIFT+F1" means that you must press the SHIFT key and hold it down while you press the F1 key.

Technical Support

Before Calling Logitek Systems for Technical Support...

▸ **Follow these steps:**

1. Check with the printed documentation, especially the *Troubleshooting, Questions and Answers,* and *Technical Information* chapters. You may also find where to look for your answer by looking in the table of contents or index.
2. If you don't find a solution by reading in the manual, check the *Release Notes* that came with PowerComm.
3. Check with your software dealer first. If the software product was not purchased directly from Logitek Systems, your first line of support is always the dealer from whom the software was purchased.
4. If you still need assistance, please obtain the following information along with a detailed description of the problem before calling our technical support department. The following information can be obtained by typing CP-VERSION (D,W) at TCL
   - PowerComm version information.
   - PowerComm serial number.
   - Pick O/S release information.

You Can Reach Logitek Systems in the Following Ways...

Technical Support Phone **(760) 607-6000**	You can reach Logitek Systems' Technical Support department, Monday through Friday between 9 AM and 5 PM Pacific Standard Time.
24 Hour Fax **(760) 607-6001**	You may also reach us by fax, 24 hours a day. We will try to respond within 1 business day of your fax if a response is requested.
Internet E-Mail **support@logitek.com**	You may reach us through internet email at this address.
Internet World Wide Web **www.logitek.com**	You can check our web page on the internet at this URL address.

PART 1
Installation Guide

CHAPTER 1
Before You Begin

This chapter briefly describes some of the terminology and concepts you will need to install and use PowerComm. It also describes how to use the menus and entry screens.

Terminology

This section briefly describes terms that are used throughout this manual.

General Pick Terms

"CC" item pointer
On R83, BASIC object code is stored in contiguous frames in which the beginning frame is stored in an item that is in the dictionary of the source code file. This item pointer contains a "CC" on attribute 1.

"CL" item pointer
On R83, list items, such as items in the POINTER-FILE, are stored in non-contigous frames in which the beginning frame is stored in an item that is in the source code file. This item pointer contains a "CL" on attribute 1.

"D" pointer
Also known as file defining items. These items are pointers to the next file level and are found in the SYSTEM or MDS file, the master dictionary (MD), and dictionaries.

"DC" file type
On R83, the "DC" files are a special "D" pointer type that permits the storing of "CC" or "CL" items.

"Q" pointer
The "Q" pointers is an alternate way of referencing a file in the current account and also the way through which you can reference a file located in another account.

ABS Area
A special area of the Pick system allocated for the operating system and assembler user modes.

ABS Frame
A frame in the ABS Area.

AP
This abbreviation stands for Pick System's version of Pick known as Advanced Pick.

API
Application Programming Interface.

BASIC
Pick's structured programming language, an extended version of Dartmouth BASIC.

Boot ABS
The ABS area of Advanced Pick that is reserved for the original product ABS diskettes. The currently executing ABS can be changed by the Advanced Pick TCL command "EXEC".

Hex Data
Characters from 0 through 9, and A through F. In PowerComm, all hex data must be in units of 2 characters.

Macro
Pick's macro facility available only on Advanced Pick.

MD
Master Dictionary.

Proc
Pick's somewhat crippled batch language facility available on R83 and AP.

R83
This abbreviation stands for Pick System's version of Pick known as R83.

TCL
Pick's command language know as Terminal Control Level.

User Modes
Assembly Language written routines that are installed into the ABS area of the operating system. These may be called from within BASIC programs.

Special Pick Characters

AM	Attribute mark, CHAR(254)
SM	Segment mark, CHAR(255)
SVM	Sub-value mark, CHAR(252)
VM	Value mark, CHAR(253)

PowerComm Communications Terms

Comm Port	An abbreviation for Communications Port.
Communications Port	Refers to the port currently being used for data communications. This would typically be the modem port.
Host	This is also known as the local, current, or originating system you are running a command on.
Me Mode	The mode of communications that sends and receives data on the same port the command was issued on.
Remote	Commonly refers to the other computer at the other end of the data communications session.

Software Operation Conventions

Menus

Menu options may be selected by number or keyword. Keywords may be shown on the menus by using the "K" option with the CP-MENU command. Or, select the "K" option from a menu to toggle the keyword display on or off. The following apply to any menu:

- To exit from a menu to the prior menu, press ENTER.
- To log off from a menu, choose "OFF".
- To exit to TCL from a menu, choose "TCL".

Entry Screens

When you are in a entry screen you are in one of three modes:

- Typing in a new value at a field
- Editing a value at a field
- Selecting a command at the command line

Typing in a New Value at a Field

When you first enter a screen, the cursor is at a field. When at a field, you may perform a command if the first character entered is one of the following characters followed by the ENTER key as shown:

This Character	Followed by ENTER Performs this Function
.	(Dot) Brings up the command line at the bottom of the screen where the record may be filed, deleted or exited. A dot "." used at the first field in a screen exits the screen.
,	(Comma) Replaces the current field with the contents of this field from the last record edited.
/	(Slash) Backs up to the previous prompt.
SPACE	(Space) Clears the contents of the current field.
;	(Semicolon) Runs a line oriented editor on the current field.

If you are at the number 1 field in a entry screen, you may exit the screen by pressing ENTER without entering data.

Editing a Value at a Field

To enter a new value at a field, just type in the new value. Or, if you would like to edit the existing value without having to re-enter the data you may use the line editing mode. This mode appears and works similar to the Pick ED verb but with more features.

To enter this mode at a field, you enter a semicolon ";" followed by ENTER. The line editor is run on the contents of the current field. All value marks are expanded into attribute marks, and sub-value marks are converted to value marks.

- You can press "?" for help once in this mode
- Use the "F" command to prompt for save options.

Once editing is done, all value marks are changed to sub-value marks, and attribute marks are changed back to value marks.

Note: Cursor keys are **not** supported for any editing function. In the line editing mode, the backspace key is defined as CTRL+H.

Tip: For making a change to a small amount of data, it is faster to type the whole data in again then it is to use this editing mode.

Selecting a Command at the Command Line

The following is a description of the available commands when you are at the following command line prompt in entry screens:

Command (n=field#/Pn=Page#/F=File/D=Delete/E=Exit/C=Copy/Q=Quit) :

This Command	Performs This Function
n	Goes to field number "n". You cannot go to field number '1' directly. You must file, delete or exit the record to return to field 1.
Pn	Goes to page "n" in a multi-page screen.
F	Files, or saves, the contents of the current record to the data file.
D	Deletes the current record from the data file.
E	Exits, or discards, any changes made to the current record.
C	Copy function. This command provides an easy way to copy a record from the same file to the current record. All fields are overlaid from the source record.
Q	Quit. Exits the current record, and exits the entry screen.

CHAPTER 2
Product Installation

The following installation instructions are by system type. Refer to the appropriate section for your system type.

Refer to *Terminology* starting on page 15 if you are unfamiliar with any terms used in this section.

PowerComm for Advanced Pick Installation Instructions

▸ Follow these steps before loading the product diskettes:

1. Logon to the DM account.

   This product needs to be installed from the DM account on Advanced Pick.

2. Check for available disk space.

   ```
   WHAT
   ```

   This product requires anywhere from 700 to 1000 frames of available space. Use the "WHAT" command from TCL and look for the number of frames under the "Available" heading.

3. Check available ABS space.

   ```
   ABS.FID ABS
   ```

   Before loading the product you can check to see how much ABS space is used on Advanced Pick by using the ABS.FID command from the DM account. Type "ABS.FID filename", where "filename" is the ABS file you wish to check. Make sure there is at least 20 frames free before loading the software if it has not already been loaded before.

4. Verify the system.

   ```
   VERIFY-SYSTEM
   ```

   It is always a good idea to verify the system before loading any new software. Use the "VERIFY-SYSTEM" verb from TCL to make sure it verifies.

5. Check that no users are logged on. Use "LISTU" from TCL to check for users logged on.

   ```
   LISTU
   ```

   As a precaution, when loading software into the ABS area, no users should be logged on. As an added safety margin, you may use the "MAXUSERS (S" command from TCL to run in single user mode.

   ```
   MAXUSERS (S
   ```

6. Have a set of formatted diskettes ready for the ABS dump.

   On Advanced Pick, if you load assembler modes into the boot ABS, you will need to make a new set of ABS diskettes. This is because both the ABS area and the ABS data files are modified during the product installation procedure and the ABS data file will no longer match the original ABS boot diskettes. Should you need to reload the ABS area you will have to use these new boot ABS diskettes that correspond to the modified ABS data file.

Note: This requirement to make boot ABS diskettes only apply if you are loading into the boot ABS. Refer to the Advanced Pick documentation for further information if needed.

▸ Follow these steps to load and install PowerComm:

1. Logon to the DM account and get to TCL if you are not logged on to DM and at TCL already.

   This product needs to be installed from the DM account on Advanced Pick.

2. Select the appropriate installation drive and options.

```
SET-FLOPPY (A,H           or            SET-DEVICE
    or
SET-FLOPPY (B,H
```

   PowerComm comes on either high density 1.44MB or 1.2MB diskettes. Use the SET-FLOPPY command with the "A" and "H" options, or to select drive B, use the "B" and "H" options. For AP/Unix systems, you may want to use the SET-DEVICE command instead of SET-FLOPPY.

3. Load the loader proc from the media.

```
T-LOAD MD (O
```

4. Run the product loader proc.

```
LOAD-POWERCOMM
```

   The load procedure will automatically load a loader program into the dictionary of BP, compile and run it.

5. Enter the account name to restore the product to.

   You may enter an account name or accept the default of "POWERCOMM" by pressing ENTER at the prompt. If you decide to quit, enter "Q".

   The account specified will be created from the account on the media. Then, the installation program, "INSTALL-POWERCOMM" will automatically be run when the account has been restored.

Tip: You may stop the following installation program at any point by entering "Q" at any prompt. If you need to re-run the installation utility, enter "INSTALL-POWERCOMM" from TCL.

6. Specify installation options for the installation program. Usually "*" for all.

   The following installation options are available:

Option	Comments
*****	All options below.
L	Load modes. This option gives you an opportunity to load the assembler code into the ABS area.
U	Update coldstart proc/macro. This option will automatically update the coldstart proc or macro to run the PowerComm coldstart routine during the system coldstart.
C	Compile included sample programs. This option will compile a few sample BASIC programs included with PowerComm.

7. Load the ABS modes by entering "Y" at the prompt.
8. If you are loading into an ABS file other than "ABS", enter "N" and enter your ABS file name, otherwise enter "Y" to load into the system ABS file.
9. Confirm with "Y" if the ABS load was successful.
10. Enter "Y" to automatically update the coldstart macro.
11. Enter "Y" to compile the included sample programs.

You may now remove the product media from the drive.

If you used the "MAXUSERS" command earlier, you may want to bring back multi-user mode by entering "MAXUSERS (M)" from TCL.

```
MAXUSERS (M
```

To complete the installation of PowerComm, refer to the next chapter, *Initial Configuration.*

If you loaded the ABS frames into the boot ABS, you will need to make a new set of ABS boot diskettes. This is required by Advanced Pick. If you did not load into the boot ABS, you can skip this step.

▸ If you loaded into the boot ABS, follow these steps to make new boot ABS diskettes:

1. Logon to the DM account if you are not logged on to DM already.

2. Make a set of boot ABS diskettes.

   ```
   ABS-DUMP ABS (A,H
   ```

   Use the "ABS-DUMP" command from TCL to make a set of matching bootable ABS diskettes. Use the appropriate options, typically "A" and "H" for drive A: and high density.

3. Check the ABS diskettes.

   ```
   T-CHK (A
   ```

   You can use the TCL command "T-CHK (A" to check the ABS diskettes.

4. You may also want to save the abs data area as well with:

   ```
   T-DUMP ABS
   ```

   Use the same command as step 3 to check them.

Warning: If you fail to verify these diskettes and they fail when you need them, you will have to go through a lengthy process of re-establishing a matching ABS area and ABS data file. This process is performed by doing a virgin boot from the original ABS diskettes, a clear-file of ABS and a select restore of the original ABS data file from the Pick data diskettes. You will then need to load the ABS modes from this product and make boot ABS diskettes. This is an Advanced Pick requirement, not PowerComm.

PowerComm Supplemental SCO Unix Disk Installation

If you are running Advanced Pick on SCO/Unix, we have provided a supplemental diskette. This diskette contains a program for Unix that allows you to send a break with PowerComm.

▸ Follow these steps to load the supplemental diskette:

1. On Unix, log in as root.

2. From Unix, change to the directory you want to load the 'sndbrk' executable program into. You can load it anywhere that all users will have access to running this program. You might want to load it where Pick is loaded for locality.

   ```
   cd /usr/lib/pick
   ```

3. Load the diskette using the Unix tar command. This is a 720K diskette. You may need to substitute the device name if you are loading from the "B" floppy drive.

   ```
   tar xvf /dev/fd0135ds9      or      tar xvf /dev/fd1135ds9
   ```

4. Update the permissions on the program to be executable by everyone.

   ```
   chmod +r+x sndbrk
   ```

5. Configure PowerComm to use the Unix program you just loaded by using the *Global Parameters Definition* screen described in the next chapter under *Global Parameters Configuration*.

PowerComm for R83 Installation Instructions

▸ Follow these steps before loading the product diskettes:

1. Logon to the SYSPROG account.

   This product needs to be installed from the SYSPROG account on Pick R83.

2. Check for available disk space.

   ```
   WHAT
   ```

   This product requires anywhere from 700 to 1000 frames of available disk space. Use the "WHAT" command from TCL and look for the number of frames under the "Avail. Overflow " heading.

3. Verify the system.

   ```
   VERIFY-SYSTEM
   ```

   It is always a good idea to verify the system before loading any new software. If you have problems with the verify, refer to page 84 for more information.

4. Decide which ABS frames you want to load PowerComm into. PowerComm now requires at least 18 ABS frames on R83.

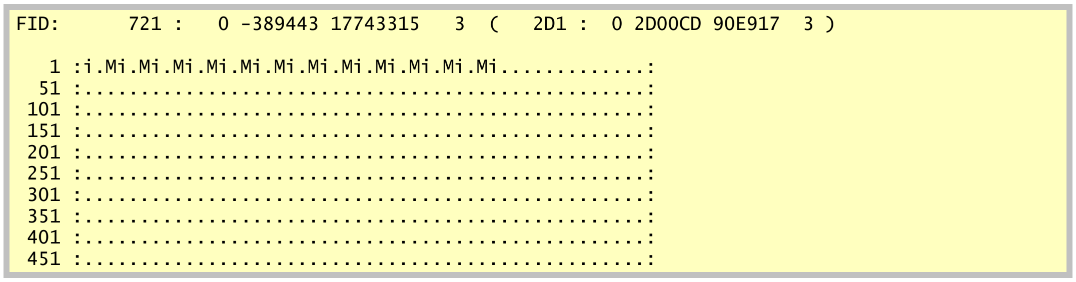
   You have 3 ranges to choose from:

Range	Frames	Comments
1	621-638	Possibly used by Keyword.
2	721-738	Possibly used by Via Systems products.
3	821-838	

   Using the command "DUMP" at TCL, check each frame by entering the command "DUMP 721-729". An empty frame usually has a repeating pattern in it of U's or dots ".". Also, check any other product you may have installed to see if it may be using these same frames.

   ```
   DUMP n1-n2
   ```

   An empty frame might look like the following:



```
FID:      721 :   0 -389443 17743315   3  (   2D1 :  0 2D00CD 90E917  3 )

   1 :i.Mi.Mi.Mi.Mi.Mi.Mi.Mi.Mi.Mi.Mi.Mi.Mi.............:
  51 :..................................................:
 101 :..................................................:
 151 :..................................................:
 201 :..................................................:
 251 :..................................................:
 301 :..................................................:
 351 :..................................................:
 401 :..................................................:
 451 :..................................................:
```

An example empty ABS frame

5. Check that no users are logged on.

   ```
   LISTU
   ```

   As a precaution, when loading software into the ABS area, no users should be logged on.

Warning: If you load the PowerComm frames over other ABS frames that have been assigned to another product you are using, you could crash your system.

▸ Follow these steps to load and install PowerComm:

1. Logon to the SYSPROG account if you are not logged on to SYSPROG already.

This product needs to be installed from the SYSPROG account on Pick R83.

2. Select the appropriate installation drive and options.

```
SET-FLOPPY (A,H
    or
SET-FLOPPY (B,H
```

   PowerComm comes on either high density 1.44MB or 1.2MB diskettes. Use the SET-FLOPPY command with the "A" and "H" options, or to select drive B, use the "B" and "H" options.

3. Load the loader proc from the media.

```
T-LOAD MD (O
```

4. Run the product loader proc.

```
LOAD-POWERCOMM
```

   The load procedure will automatically load a loader program into the dictionary of BP, compile and run it.

5. Enter the account name to restore the product to.

   You may enter an account name or accept the default of "POWERCOMM" by pressing <ENTER> at the prompt. If you decide to quit, enter "Q".

   The account specified will be created from the account on the media. Then, the installation program, "INSTALL-POWERCOMM" will automatically be run when the account has been restored.

Tip: You may stop the following installation program at any point by entering "Q" at any prompt. If you need to re-run the installation utility, enter "INSTALL-POWERCOMM" from TCL.

6. Specify installation options for the installation program. Usually "*" for all.

   The following installation options are available:

Option	**Comments**
*****	All options below.
L	Load modes.
	This option gives you an opportunity to load the assembler code into the ABS area.
U	Update coldstart proc.
	This option will automatically update the coldstart proc to run the PowerComm coldstart routine during the system coldstart.
C	Compile included sample programs.
	This option will compile a few sample BASIC programs included with PowerComm.

7. Select the range of ABS frames, from 1 to 3, you want to use.
8. Confirm your choices by dumping the frame contents.
9. If you are sure you want to use these frames, enter "Y" otherwise choose "Q" to quit.
10. Enter "Y" to automatically update the coldstart proc.
11. Enter "Y" to compile the included sample programs.

To complete the installation of PowerComm, refer to the next chapter, *Initial Configuration.*

You may now remove the product media from the drive.

CHAPTER 3 Initial Configuration

This chapter documents the initial configuration of PowerComm.

Note: The PowerComm software account name is "POWERCOMM" unless you changed the default when PowerComm was installed.

Refer to *Terminology* starting on page 15 if you are unfamiliar with any terms used in this section.

Refer to *Software Operation Conventions* starting on page 16 for information on using the PowerComm menus and entry screens.

Coldstart Configuration

The coldstart routine should be modified to clear the communications port at coldstart time. This will clear out any communications port table entries that may have been active at the time of system shutdown.

In addition, on R83, the ABS modes should be loaded in case an ABS load was performed. On, AP, you should already have a set of new boot ABS diskettes if you loaded PowerComm into the boot ABS.

PowerComm attempts to make these modifications during installation, but you should check to make sure it has been done correctly.

The following modifications are by system type. Refer to the appropriate section for your system type.

Advanced Pick Coldstart Modifications

You need to make sure the coldstart macro or proc has been modified to contain the following lines.

On AP, this would typically be the USER-COLDSTART item in the MD of the DM account.

```
RUN LSCP,BP LSCP.COLDSTART]LOAD:ABS
EXEC ABS
RUN LSCP,BP LSCP.COLDSTART]CLEAR
```

If the coldstart procedure is a macro

Note that the brackets "]" above are value marks, entered by pressing CTRL+] .

```
HRUN LSCP,BP LSCP.COLDSTART
STON
HLOAD:ABS<
P
HEXEC ABS
P
HRUN LSCP,BP LSCP.COLDSTART
STON
HCLEAR<
P
```

If the coldstart procedure is a proc

These modifications to the coldstart procedure will automatically load the ABS after a coldstart and clear the communications tables.

If you have specified an alternate ABS file, change the filename "ABS" to what you have specified during the installation procedure.

Warning: If you have not loaded the ABS modes, you should not run **ANY** command of PowerComm except LSCP.COLDSTART or INSTALL-POWERCOMM. **FAILURE TO LOAD THE ABS MODES BEFORE RUNNING THE SOFTWARE COULD CRASH YOUR SYSTEM!**

Advanced Pick Optional Coldstart Modifications

Once the PowerComm ABS modes have been loaded, it is **not** necessary to load the modes for each coldstart. If you want to save time during the coldstart procedure you may remove the lines from the coldstart proc or macro that loads the modes. However, you should leave the lines in the proc or macro that clear the communication ports. If you reload the original ABS diskettes or create a new ABS file, you can run the LSCP.COLDSTART command to reload the PowerComm ABS modes.

Advanced Pick Special Account Logon

PowerComm uses an account logon that is defined in the MDS file as well as an item in the USERS file. This account is automatically set up during the install procedure.

Account Logon Name	Purpose
CP:COMM	This logon is automatically invoked by the CP-START command which logs on the comm port to this logon which runs a proc then runs a program that starts the assembly code server program. ***You should never log on to this account!***

The "CP:COMM" logon must exist for PowerComm to work properly on all AP systems.

R83 Coldstart Modifications

You need to make sure your coldstart proc has been modified to contain the following lines.

On R83, this would typically be the USER-COLDSTART item in the SYSPROG-PL file.

```
HRUN LSCP,BP LSCP.COLDSTART
STON
HLOAD CLEAR<
P
```

These modifications to the coldstart procedure will automatically load the ABS after a coldstart and clear the communications tables.

Warning: If you have not loaded the ABS modes, you should not run **ANY** command of PowerComm except LSCP.COLDSTART or INSTALL-POWERCOMM. **FAILURE TO LOAD THE ABS MODES BEFORE RUNNING THE SOFTWARE COULD CRASH YOUR SYSTEM!**

Modifying the Logon Procedure

If you loaded the PowerComm product into a different ABS then the boot ABS, you will have to make sure you "EXEC" the ABS file you loaded the product into. This EXEC has to be in the logon proc, or the users file, for the PowerComm account you installed.

In this example the ABS file is "ABS1". There should also be a "Q" pointer to the account the "ABS1" file is in.

Example:

```
POWERCOMM
001 PQ
002 C DEFAULT LOGON PROC FOR POWERCOMM
003 C (C) COPYRIGHT 1994, LOGITEK SYSTEMS
004 C
005 C MODIFY AS NECESSARY FOR YOUR SYSTEM
006 C
007 HTERM-TYPE
008 P
009 HEXEC ABS1
010 P
```

Modified PowerComm logon proc

Important Note: Any accounts that use PowerComm commands must also EXEC the proper ABS file.

Global Parameters Configuration

The *Global Parameters Definition* screen is accessed using CP-MENU and then selecting the *Global Parameters Definition* option.

Fast User Exits Configuration

The fast user exits feature is supported on the 6.1.x version of AP. If you have this version, you can configure PowerComm to pre-load the user exits for faster runtime execution.

To enable this feature, enter "T" for test mode. When you exit and re-run the menu with CP-MENU, the screen will display a message indicating a test mode has started. If it succeeds, a message indicating success will be displayed and this option will automatically be changed from "T" to "Y". If it does not succeed, it may abort, and subsequent running of CP-MENU will clear the flag. See the *Global Parameters Definition* screen documentation starting on page 127, for additional information.

Unix Break Program Configuration

If you are installing PowerComm on to a Unix system, we have supplied a Unix utility which enables PowerComm to output a BREAK through a comm port. You need to enter the full Unix path to the command in the *Global Parameters Definition* screen, field *Unix break program location*.

If you had loaded the diskette into /usr/lib/pick, then the correct entry in this screen would be "/usr/lib/pick/sndbrk".

If you have not loaded the Unix utility, refer to page 18 for installation instructions.

Port Definition Setup

The *Port Definition* screen is accessed using CP-MENU and then selecting the *Port Definition* option.

The *Port Definition* screen allows you to configure parameters either globally for all ports using "G" as the record ID, or you may enter an individual port number. When using a port number, any field that is filled in overrides the global parameter for that field.

For more details about the *Port Definition* screen, refer to page 129.

Screen Displays Setup

The PowerComm screen displays are designed to use the dim (half intensity) cursor control attribute for the static text and normal intensity for the field values. This normally has been associated with @(-7) and @(-8) in the terminal device control tables. On more recent versions of Pick, there is a @(-100) and a @(-101) definition that corresponds to half and normal intensity as well. You may have to play around with these numbers to obtain the correct appearance of the screens. You may disable the feature by removing the 2 parameters from the global fields or cancel the global default by entering a backslash "\" into these 2 fields for a particular port.

If all of your terminals are the same and the "-7" and "-8" definitions display half intensity, you can leave the settings in the global port definition alone. You may override the default on a port by port basis by entering the port number, then changing the field values for the fields, *Dim on @ cursor control*, and *Dim off @ cursor control*.

Break Key Setup

In order to use the BREAK key in PowerComm, you must configure the appropriate parameter for your system type.

AP/Unix Systems

On AP/Unix platforms, the BREAK key feature is configured in the *Global Parameters* screen and is described in the previous section titled, *Unix Break Program Configuration.*

AP/Pro

On recent versions of AP/Pro, the 'send-break' command has been provided by Pick Systems to send a break out a port regardless of board type. This is the only way to send a break out an intelligent port. If this command is not available, you will have to use the 'Base address' field described in the 'All Others' section heading following this one.

To use the 'send-break' command on AP/Pro, fill in the 'Send break TCL command' field in the *Port Definition* screen (page 7 of 7), field *Send break TCL command*. This field may contain a percent sign "%" to indicate the current port number. This allows you to use the "G"lobal item for all ports in one place:

```
SEND-BREAK %
```

All Others

For R83, AP/Native, AP/DOS (and earlier versions of AP/Pro) you must define the port address in the *Port Definition* screen, field *Base address*. The base address is only available for dumb (non-intelligent) ports.

This field must be 4 hex digits.

The following are standard COM port base addresses:

COM 1, IRQ 4, address is 03F8
COM 2, IRQ 3, address is 02F8

Example:

A 4 port Digiboard may have the following addresses by communications port:

Port 1	01B0
Port 2	01B8
Port 3	01C0
Port 4	01C8

Refer to the PC or Pick documentation for more details about your serial board's base addresses. On recent versions of Advanced Pick, you can use the CONFIG command (in the DM account) to display this information.

Start and Kill Port Subroutine Configuration

We have provided a hook that allows you to run a subroutine before a communications port is started with optional parameters from the *Port Definition* screen. Another subroutine can be called after the port is killed.

For information about using this feature and about CP.AP.UNIX.UTY.SUB, refer the section *Using the Before Start and After Kill Port Program Parameters* starting on page 73 in the *Technical Reference* chapter.

AP/Unix Protocol Characters Configuration

On AP/Unix, you can re-map the escape (level push) and break keys to a different character. If you have re-mapped either of these keys to a character less than decimal 27, you may have to make changes to the protocol characters PowerComm uses.

These changes may be made in the *Port Definition* screen, fields starting with the sub-heading "[Protocol Characters]". Refer to page 133 for more details.

If you make changes to the protocol characters, you also have to change each system that PowerComm may be connected with. In other words, the protocol characters have to match on both ends of the communications link or it will not function properly.

Installing on Other Accounts

We recommend that you install the PowerComm product into another account before using it. That way, any files you create will be localized in your own account, making a future upgrade to a new PowerComm version easier.

To install PowerComm on to another account, you access the main PowerComm menu from TCL by using CP-MENU. Then select the *Install on Another Account* menu selection on the main menu.

If you have loaded PowerComm into another ABS file, make sure you EXEC the proper ABS file before running any PowerComm commands.

Hardware Configuration Recommendations

Cabling Requirements

PowerComm sits on top of Pick and uses whatever facilities, or lack thereof, of the version of Pick you are using provides. If your version of Pick does not support hardware flow control, PowerComm does not either.

To use PowerComm with a modem you can use the standard modem cable wired according to the instructions that came with your version of Pick with the following precautions.

- By default, PowerComm disables XON/XOFF flow control when starting a port.

  This is to eliminate noise on the phone line from inadvertently locking up the port with the XOFF character, CTRL-S. You can disable turning off XON/XOFF flow control by using the "X" option with the CP-START PowerComm TCL command. Refer to the topic, *XON/XOFF Mode*, starting on page 79 in the *Technical Information* chapter for more information about XON/XOFF mode.

- Do not use a modem cable or a null modem cable that does not properly maintain a HIGH or LOW signal to Pick's flow control pins.

If the pins are not properly connected to a signal, they can float causing unexplained port lockup problems. When ports are locked up, a complete system shutdown and reset is usually required.

Specific Notes by System Type

Keep the following points in mind when using modems and cables with your version of Pick:

Pick R83

- On Pick R83, outbound hardware flow control is controlled on DSR, pin 6 on a 25 pin cable. This is not CTS, pin 5, which is the PC standard.
- Pick R83 does not support inbound flow control of any kind.

AP/Native, AP/Pro, AP/DOS

- Some outbound hardware flow control has been implemented on certain versions and intelligent ports. Refer to your Pick release notes documentation for details.
- These versions do not support inbound flow control of any kind.

AP/Unix

- On AP/Unix, all flow control, including inbound, outbound, XON/XOFF and hardware flow control is dependent upon the TTY device associated with the port. Many of the parameters are changeable using the STTY command.

For all versions, refer to your Pick system or Unix documentation for more information about specific cabling requirements.

Modem Configuration

Use 8 Data Bits

For best results, make sure you are using 8 data bits instead of 7. This will speed up data communications by not having to have both the sending and receiving computers mask the high order bits on each character. Refer to the topic, *Binary 7 bit mode,* starting on page 79 in the *Technical Information* chapter for more information about 7 bit mode.

Reset Modem to No Echo and No Result Codes

We recommend that you reset your modem switches to NO ECHO, and NO RESULT CODES. When a modem has ECHO and RESULT CODES enabled, it is possible to lock up the communications port because of how Pick's logon message and the modem can interact. The logon message is echoed back to Pick which is then echoed back to the modem and so on. Usually Pick will lock up, often times a reboot is required to correct this problem. To avoid this, reset the dip switches on the modem to NO ECHO, NO RESULT CODES.

When you need to use the modem, it is simple enough to enable echo and result codes before usage with the Hayes "AT" command of "ATE1Q". Also, when you have completed your work over the modem, get into a habit of typing "ATZ" to reset the modem back to the dip switch settings of NO ECHO, and NO RESULT CODES. The dip switch information should be documented in your modem's documentation.

Newer modems without dip switches work principally the same way, but store the power-on settings in non-volatile memory. These modem commands usually involve selecting the settings you want, and saving them with another modem command to one of several configuration banks in the modem. Refer to your modem documentation for more details.

Set Modems to Dumb Mode on Remotes

Unless you need dial-out capabilities on the remote end, you can reduce the likelihood of modem wars by setting the remote modems to DUMB mode so that they will never recognize any commands.

Other Software Configuration Recommendations

Use LOGON-LOCK on AP

Pick R83 version 3.1 uses a technique to reduce modem feedback wars with the logon prompt. This works by making the user press a key, other than the ENTER key, two times in a row to get the logon prompt. Now, on AP version 6.1, there is a command "LOGON-LOCK" which provides the same functionality for AP on a port-by-port basis. Earlier versions do not have this feature and will continually scroll the logon message to the modem and back to the system until the port usually locks up.

Add the "LOGON-LOCK" to your logon procs or coldstart procedure:

For all ports:

```
    LOGON-LOCK (A,N
```

Or, to enable the feature for ports 'n' through 'm'.

```
    LOGON-LOCK (n-m,N
```

Remove "AT" Characters from Logon Screens and Error Messages

To reduce the likelihood of the modem locking up or the port locking up, refer to the previous recommendation about setting NO ECHO and NO RESULT CODES. Another step you can do is to remove any sequence of characters in your logon screen that could be interpreted as a command attention sequence by your modem. This would include any occurrence of the letters "AT" or "at" anywhere in the screen. This includes the error message "logged off **at** ..." in the *messages* (ERRMSG on R83) file, item 336. The other message to check is 340 "connected for …". The following shows the Advanced Pick messages.

```
:print-err messages 336 340
< logged off at 15:37:44      on 09 Apr 1996           >
< Connect time= A Mins.; CPU= B Units; LPTR pages= C       >
```

PART 2
User Guide

CHAPTER 4
Using TCL Commands

This chapter provides information and examples on how to use some of the TCL commands with various options. PowerComm TCL commands are documented in detail in the *TCL Commands Reference* chapter starting on page 94.

Introduction

Setting up a Communications Port

The following commands demonstrate the basic operations required to use PowerComm from TCL.

This Command	Performs This Function
CP-START	Starts a communications server on the port usually connected to the modem or another computer
CP-ATT	Reserves a communications port for your use.
CP-DET	Releases the communications port for use by another user.
CP-KILL	Stops the communications server and returns the port to logon.

Using a Communications Port

Once a communications port has been started and attached, you may use any of the following TCL commands to perform communications work.

This Command	Performs This Function
CP-SEND	Send a file to another system running PowerComm.
CP-RECV	Receive a file from another system running PowerComm.
CP-ABORT	Aborts a CP-SEND or CP-RECV data transfer on the remote system.
CP-SEND-ASCII	Sends data to a remote computer in ASCII text format.
CP-RECV-ASCII	Receives data from a remote computer in ASCII text format.
CP-SEND-PICK	Sends a file to a remote Pick computer in dumb mode.
CP-RECV-PICK	Receives a file from a remote Pick computer in dumb mode.
CP-DIR	Enter direct mode with currently attached communications port.
CP-CAPTURE	Enter direct mode with currently attached communications port and capture all data to the capture file specified.

Using Me Mode

Normally, most commands use a communications port to communicate with another system. However, the "ME" mode of operation does not use a communications port. This mode involves using the same port as the command was entered from. The "ME" mode of operation is available on some commands. These commands always use the "M" option for this mode.

Figure 1.

In figure 1, user "L" on port "L1" is attached to a comm port running on port "L2" which is connected to the remote system port "R3". Entering TCL commands on the local system does not require the "M" option since the data is moving through the comm port. However, when receiving data on the remote system from the local system, port "R3" is used for communications and not a comm port. If you were to type CP-RECV on the remote system without the "M" option, the command would fail with the message, "None attached." This is because, by default, a comm port is used for data communications. The "M" option overrides this default and causes the data communications to occur on the same port.

Example:

Local System		**Remote System**
User L on port L1		**dialed in on port R3**
		type CP-RECV BP (M exit direct mode
type CP-SEND BP *		
	data movement →	

Using the ME mode on the remote system to receive a file

In this example, data is sent to the remote system through local comm port "L2", out the local modem, into the remote modem, and to the remote system through remote port "R3". Since port "R3" is the same port I was on, I used the "M" option to receive the data.

The same option is used with sending data from the remote system to the local system. I would use the "M" option on the CP-SEND command.

Example:

Local System		**Remote System**
User L on port L1		**dialed in on port R3**
		type CP-SEND BP * (M exit direct mode
type CP-RECV BP		
	← data movement	

Using the ME mode on the remote system to send a file

In this example, data is sent out the remote system through remote port "R3", to the remote modem, into the local modem, through local comm port "L2". Since port "R3" is the same port I was on, I used the "M" option when sending the data.

Using Port Numbers or Names

PowerComm allows you to reference comm ports that have been started by numeric port number or by the name you used when you started the port. If you did not specify the name when using CP-START, the default name is "P" and the port number.

Example:

```
:CP-START 15,19200
[CP.008] Communications port number 15 started, named P15.

:CP-LIST

      Port Name........... Mode..... Status Input..(size) Output.(size) Entry#
        15 P15                       Idle         0( 20k)       0( 20k)    1

[CP.006] 1 communications ports listed.
```

The next two commands are equivalent because the comm port can be referenced by either 15 or P15.

Example:

```
:CP-ATT 15
[CP.020] Communications port P15 attached.
:CP-ATT P15
[CP.020] Communications port P15 attached.
```

Using a port number or name to reference a communications port

Note: Most messages generated by PowerComm reference the port name even though you may have referenced the port number with the command.

Omitting the Port Number or Name

To save time and typing, PowerComm allows you to omit the port number from many commands when only 1 comm port is running.

Example:

```
:CP-ATT
[CP.020] Communications port P15 attached.

:CP-LIST

      Port Name........... Mode..... Status Input..(size) Output.(size) Entry#
*   1 => 15 P15                      Idle         0( 20k)       0( 20k)    1

[CP.006] 1 communications ports listed.
```

Attaching a port when 1 communications port is running

Likewise with only 1 comm port running, you may kill the port without specifying the port number.

Example:

```
:CP-KILL
[CP.025] Communications port number 15 killed.
```

Using CP-START to Start a Communications Port

In order to use PowerComm you need to set up a port for communications. This involves starting a server process on the comm port with the CP-START command. In addition to starting the server, this command may also set up the ports' baud rate, and other communications parameters.

The CP-START command may take several parameters, each separated by a comma.

The first and only required parameter is the port number. It must not be logged on to a user or already in use by PowerComm. If the port is unavailable, an appropriate message will be displayed.

Port communications parameters are specified in the second parameter and are separated by colons ":". Any of the parameters may be omitted and the current port settings will be used instead.

An optional port name may be specified in the third parameter. The port started may be named for the convenience of referring to the port by its name in subsequent commands. If you do not specify a name, PowerComm will default the name as "P" and the port number.

The fourth and fifth parameters are the input and output buffer sizes in bytes. The defaults when these are not specified are configured in the *Port Definition* screen.

Once the comm port is started, it may be attached and reserved by using the CP-ATT command.

The following example starts a communications port on port 15 at 19200 baud. If the port is already at the speed you want, you may omit the baud rate.

Example:

```
:CP-START 15,19200
[CP.008] Communications port number 15 started, named P15.
```

The following example starts a communications port on port 15 at 19200 baud, with a throughput rate of 9600 baud. In addition, ODD parity, 1 stop bits and 7 data bits is chosen. The port is named "Modem" and the input buffer size is selected at 1000 bytes and the output buffer size of 4000 bytes. Because this example is running on a 2K frame size system, the input buffer size is rounded up to 2K bytes. The "$$$" options invokes the CP-LIST command showing details about the port just started and the "A" option automatically attaches the port after the start command completes.

Example:

```
:CP-START 15,19200:9600:ODD:1:7,Modem,1000,4000 (A,$$$
[CP.008] Communications port number 15 started, named Modem.
[CP.142] Baud rate: 19200 (9600), parity: ODD, stop bits: 1, word length: 7.
         Input buffer: 2,000 bytes, output buffer: 4,000 bytes, entry: 0.

      Port Name........... Mode..... Status Input..(size) Output.(size) Entry#
        15 Modem                     Idle          0(  2k)       0(  4k)    0
                           Baud.( Link) P S W  (Original settings)
                           19200( 9600) O 1 7   19200 N 1 8
                           Original Statuses   Start
                           TA  FC  XCS DCD     Options
                           ON  OFF ON  OFF     B
                           Started......... Attached........ Mode............
                           03-04 10:10:45pm

[CP.006] 1 communications ports listed.
[CP.020] Communications port Modem attached.
```

Using more parameters with the CP-START command

Using Quick Start, Attach and Enter Direct Mode

To save time when starting a comm port, PowerComm provides two option with the CP-START command that allows you to attach the port and go into direct mode:

- To attach the port, use the "A" option.
- To enter direct mode, use the "D" option.

```
:CP-START 15,19200 (A,D
[CP.008] Communications port number 15 started, named P15.
[CP.020] Communications port P15 attached.
[CP.031] Direct connected to communications port P15.
```

Using the "A" and "D" options to attach and enter direct mode with the CP-START command

For more options and information on using CP-START refer to page 120.

Using CP-ATT to Attach a Communications Port

The purpose of attaching a port is to allocate the port to your port so that all of the communications ports input and output is reserved for your port only. Starting the port simply starts a server program that allows any port to attach and use it. The CP-ATT command attaches the comm port specified and reserves it for your port and prevents others from allocating the same resource. PowerComm allows you to have multiple ports attached at the same time. This feature allows you to have several comm ports reserved for your port. However, the most recently attached port is the port that will be used for any communications work.

Example:

```
:CP-ATT 15
[CP.020] Communications port P15 attached.
```

Using CP-ATT with More than One Port

It is possible with PowerComm to have more than one port attached to your port at the same time. However, only the most recently attached port is used for communications work. There are a couple of methods to find out which comm port is the current port.

You may use CP-LIST to view the ports.

Example:

```
:CP-START 2,19200 (A
[CP.008] Communications port number 2 started, named P2.
[CP.020] Communications port P2 attached.
:CP-START 3,19200 (A
[CP.008] Communications port number 3 started, named P3.
[CP.020] Communications port P3 attached.
:CP-LIST

      Port Name........... Mode..... Status Input..(size) Output.(size) Entry#
*  1 ->  2 P2                        Idle       227( 20k)       0( 20k)    0
*  1 =>  3 P3                        Idle         0( 20k)       0( 20k)    1

[CP.006] 2 communications ports listed.
```

Using CP-LIST to determine currently attached port

Special characters in the CP-LIST command display

The "*" in the very left most position is shown when the port number is your port. The "->" and "=>" indicate which comm ports that port 1 has attached. The current attached port is shown with "=>". In this example, any subsequent communications work will occur with port 3.

You may use CP-ATT with the "C" option to check which is the currently attached port. The "C" option will also clear the mode that may be left from aborting certain commands.

Example:

```
:CP-ATT (C
[CP.020] Communications port P3 attached.
```

Using CP-ATT to determine currently attached port

When you have more than 1 comm port attached, detaching 1 comm port automatically reattaches one of the remaining ports making it the current attached port.

Example:

```
:CP-DET
[CP.029] Communications port P3 detached.

:CP-LIST (T

      Port Name........... Mode..... Status Input..(size) Output.(size) Entry#
*  1 =>  2 P2                        Idle       227( 20k)       0( 20k)    0
         3 P3                        Idle         0( 20k)       0( 20k)    1

[CP.006] 2 communications ports listed.
```

Using CP-LIST to View Communications Ports

The CP-LIST command is used to show the current communications ports that are running. In addition, you can determine which ports have which communications ports attached.

If another port has the communications port attached, it is shown to the left of the communications port with either a single arrow "->" or a double arrow "=>". The double arrow represents the current port and the single arrow shows up only when a port has additional communication ports attached. The "*" to the left of the port number is shown to indicate it is your own port number.

Each of the column headings is explained in detail starting on page 65 in the *Technical Information* chapter.

In the following example port 12 has 2 communications ports attached and port 14 is the current port. Communications port 13 is reserved by port 12 but not currently in use.

Example:

```
:CP-LIST

      Port Name............ Mode..... Status Input..(size) Output.(size) Entry#
  18 =>  2 P2              RECV      Active   3,351( 20k)       0( 20k)    0
        11 Office                    Idle         0(  7k)       0(  2k)    5
  12 -> 13 Laptop                    Idle       102(  2k)       0(  7k)    2
  12 => 14 P14             ASCII     Active       0( 20k)   2,954( 20k)    3
  19 => 15 P15             DIR       Idle         0( 20k)       0( 20k)    1

[CP.006] 5 communications ports listed.
```

Using CP-LIST to view current port attachments and statuses

This next example uses some additional options. The "M" option lists only my ports. The "*" next to the port number indicates that port 12 is my port. The "C" option displays communications parameters and the "T" option shows communications port start time and last attachment times.

Example:

```
:CP-LIST (C,M,T

      Port Name............ Mode..... Status Input..(size) Output.(size) Entry#
* 12 -> 13 Laptop                    Idle       102(  2k)       0(  7k)    2
                           Baud.( Link) P S W  (Original settings)
                            4800( 1200) S 2 7   19200 N 1 8
                           Started......... Attached........ Mode............
                           03-01 07:46:56pm 03-01 07:46:57pm

      Port Name............ Mode..... Status Input..(size) Output.(size) Entry#
* 12 => 14 P14                       Idle         0( 20k)       0( 20k)    3
                           Baud.( Link) P S W  (Original settings)
                           19200        N 1 8   19200 N 1 8
                           Started......... Attached........ Mode............
                           03-01 09:49:41pm 03-01 09:49:41pm

[CP.006] 2 communications ports listed.
```

Using CP-LIST with several options

Using CP-DET to Detach a Communications Port

When you have completed communications work with the comm port you may release it for use by others using the CP-DET command.

Example:

```
:CP-DET 15
[CP.029] Communications port P15 detached.
```

Tip: It is not necessary to detach the port in order to kill it using the CP-KILL command.

Detaching a Port from Another User

Occasionally it may be necessary to detach a communications port from another user. This may occur when the communications port has been left in direct mode by another user that is no longer using the port. Using the CP-DET command, you can detach the port from the other user by using the "U" option.

This function is only supported if the communications port is in one of the following modes:

Mode	Meaning
ASCII	ASCII send or receive mode from using the CP-SEND-ASCII, CP-RECV-ASCII, CP-SEND-PICK, CP-RECV-PICK commands.
DIR	Direct mode from using the CP-DIR or CP-CAPTURE commands
SCRIPT	Script mode from using the CP-SCRIPT command.

The current mode can be shown using the CP-LIST command.

The following example shows port 19 which is currently in direct mode with communications port 15. It is detached from port 15 by this port using the CP-DET command with the communications port number and the "U" option. Port 19, will automatically exit direct mode when this occurs with a detached message.

Example:

```
:CP-LIST

      Port Name........... Mode..... Status Input..(size) Output.(size) Entry#
  19 => 15 P15             DIR       Idle          0( 20k)       0( 20k)    1

[CP.006] 1 communications ports listed.

:CP-DET 15 (U
[CP.037] Port number 19 detached from communications port P15

:CP-LIST

      Port Name........... Mode..... Status Input..(size) Output.(size) Entry#
        15 P15                       Idle          0( 20k)       0( 20k)    1

[CP.006] 1 communications ports listed.
```

Using the "U" option with CP-DET to detach a port from another user

Using CP-KILL to Stop Communications Ports

When the comm port is no longer needed for any other communications work, you can use the CP-KILL command to halt the server program and reset the ports' communication parameters back to their original values. If you skip the CP-DET command before killing one of your own ports, the CP-KILL command will automatically detach the port first.

You may not kill a port that is attached to another port without using the CP-DET command with the "U" option or the CP-KILL command with the "U" option.

Example:

```
:CP-KILL 15
[CP.025] Communications port number 15 killed.
```

Killing All Ports Quickly

To quickly shutdown all communications ports on the system, PowerComm provides two options on the CP-KILL command. The "U" option first invokes the CP-DET command with the "U" option on each port specified. The "A" option refers to all ports possible.

In this example, the "A" and "U" options are used to quickly stop all 5 communications ports running on the system.

Example:

```
:CP-LIST

      Port Name........... Mode..... Status Input..(size) Output.(size) Entry#
  16 =>  4 P4                        ?             0( 10k)       0( 16k)    4
  16 -> 11 P11                       Idle          0( 20k)       0( 20k)    3
  16 -> 13 P13                       ?             0( 20k)       0( 20k)    1
        14 P14                       Idle          0( 20k)       0( 20k)    2
* 12 => 15 P15                       Idle          0( 20k)       0( 20k)    0

[CP.006] 5 communications ports listed.

:CP-KILL (A,U
[CP.037] Port number 16 detached from communications port P4
[CP.037] Port number 16 detached from communications port P11
[CP.037] Port number 16 detached from communications port P13
[CP.019] Communications port P14 not attached
[CP.029] Communications port P15 detached.
[CP.025] Communications port number 4 killed.
[CP.025] Communications port number 11 killed.
[CP.025] Communications port number 13 killed.
[CP.025] Communications port number 14 killed.
[CP.025] Communications port number 15 killed.
```

Using CP-KILL with the "A" and "U" options to quickly kill all communications ports

Using CP-DIR to Enter Direct Mode

Direct mode is used after attaching a port. This mode allows you to use the comm port as a pass through device that takes all of your keystrokes and sends them out the comm port. All incoming characters are displayed on the screen. You may find this mode useful when dialing up another system to execute commands and run applications.

Example:

```
:CP-DIR
[CP.031] Direct connected to communications port P15.

[CP.032] Direct connect exited from communications port P15.
```

To exit direct mode you must press an escape key. The escape key is defined by port or globally in the PowerComm menu selection, *Port Definition*, documented beginning on page 129. In addition to customizing the escape key in the menu, you may override this definition when using CP-DIR by using the "A" option.

Note that the default escape key is not the ESCAPE key. Many applications and terminals use escape keys so we have chosen the default of the accent mark "`" instead. You may change this if needed.

The following example uses the "A" option to prompt the user for a new exit and break key. It also demonstrates the "T" option to show how long direct mode was used

Example:

```
:CP-DIR (A,T
[CP.031] Direct connected to communications port P15.
[CP.104] Escape character is "`".
[CP.105] Alternate break character is "Control V".
Enter single character to represent exit option:_
Character entered was "Control A"; 1 in ASCII decimal.
Enter single character to represent break option:_
Character entered was "Control C"; 3 in ASCII decimal.
[CP.104] Escape character is "Control A".
[CP.105] Alternate break character is "Control C".
ATI7
Configuration Profile...

Product type           External
Options                HST
Clock Freq             10.0Mhz
Eprom                  64k
Ram                    32k

Supervisor date        10/02/90
IOP date               05/17/89
DSP date               09/18/89

Supervisor rev         2.4
IOP rev                1.0
DSP rev                2

OK

[CP.032] Direct connect exited from communications port P15.
[CP.121] Direct mode active for 2.20 minutes.
```

Using CP-DIR with the "A" option to ask for new exit and break character keys

For More Information

For more options and information on using CP-DIR refer to page 103.

Note: In order to use the alternate break key, you must have set up the break key definitions in the PowerComm menu. Refer to page 25 for configuration information. When using the "A" option, if you press ENTER for either question, the key will remain unchanged. Any other key will take that key as the key to use. In the rare instance you want to assign the ENTER key as either the exit or alternate break key, you may do so using stacked input with hex notation as shown in the examples for CP-DIR starting on page 103.

Using CP-CAPTURE to Enter Direct Mode with Capture

Direct mode can also be used with a capture option to capture all data received to a data file.

Note: During direct mode with capture, the captured data is stored in a buffer until the exit or break key is pressed. Depending upon how long you have been capturing data, there may be a slight delay after using the exit or break key while the captured data is written to the data file.

The following example invokes the PowerComm CP-CAPTURE command to capture the session to a file.

Example:

```
:CP-CAPTURE CAPTURE.FILE MODEM.CONFIG
[CP.046] Capture item "MODEM.CONFIG" in file "CAPTURE.FILE" created.
[CP.031] Direct connected to communications port P15.
[CP.036] Warning: No <BREAK> key option
ATI7
Configuration Profile...

Product type           External
.
.
.
OK

[CP.032] Direct connect exited from communications port P15.
[CP.035] Capture item "MODEM.CONFIG" in file "CAPTURE.FILE" updated.
         Added 375 bytes for a total of 375 bytes.
```

Using CP-CAPTURE to record a direct mode session

The PowerComm CP-PLAYBACK command may be used to play back the capture file to the screen or printer.

```
:CP-PLAYBACK CAPTURE.FILE MODEM.CONFIG
ATI7
Configuration Profile...

Product type           External
.
.
.
OK

[CP.047] Playback of "MODEM.CONFIG" in file "CAPTURE.FILE" completed.
```

Using CP-PLAYBACK to view the captured data

For More Information

For more options and information on using CP-PLAYBACK refer to page 109.

Using CP-DIR with Monitor Mode

Direct mode can also be used with a monitor option to view received data in both ASCII and hex at the same time.

The following example invokes the PowerComm CP-DIR command with the "M" option to enter direct mode with the monitor option. The screen display is a result of an "ATI7" command to the modem. All hex data is in brackets and shown in half intensity.

Example:

```
:CP-DIR (M
[CP.031] Direct connected to communications port P15.
[0D]]T[54]I[49]7[37]
Configuration Profile...

Product type           External
Options      [0D0A436F6E66696775726174696F6E2050726F66696C652E2E2E0D0A0D0A50726F
6475637420747970652020202020202020202020204578746572616C0D0A4F7074696F6E73202020
202020]          HST
.
.
.
DSP rev                2
[20202020202020202020204853540D0A436C6F636B20467265712020202020202020202020202020313
.
.
.
02020202020320D0A]
OK
[0D0A4F4B0D0A]
```

Using CP-DIR with the monitor mode option

The next example uses the numeric option of "10" to limit the number of ASCII characters shown at once to 10 characters at a time. This can make it far easier to make the corresponding relationship between ASCII data and its hex equivalent. The screen display is a result of an "ATI7" command to the modem.

Example:

```
:CP-DIR (M,10
[CP.031] Direct connected to communications port P15.
[0D]]T[54]I[49]7[37]
Configur[0D0A436F6E6669677572]ation Prof[6174696F6E2050726F66]ile...

[696C652E2E2E0D0A0D0A]Product ty[50726F64756374207479]pe        [706520202020202
02020]   Externa[20202045787465726E61]l
.
.
.
                      DSP rev  [0A445350207265762020]          [202020202020202020
2020]    2
[20202020320D0A]
OK
[0D0A4F4B0D0A]
```

Using CP-DIR with monitor mode options

For More Information

For more options and information on using CP-DIR refer to page 103.

Using CP-SEND and CP-RECV to Send and Receive Files

PowerComm can be used to transfer files between two systems with PowerComm installed. This section demonstrates the PowerComm TCL commands CP-SEND and CP-RECV.

The following examples assume a connection has already been established to the remote system. For clarity, the file name used on the remote system is BP.REMOTE. The local file is named BP.LOCAL.

Using CP-SEND to Send a File to the Remote System

To send a file to the remote system we use the PowerComm commands CP-SEND and CP-RECV. The CP-SEND command is used on the local system and the CP-RECV command is entered on the remote system. Note that we must use the "M" option with the command invoked on the remote system in order to receive data on the same port as the command was entered from.

The following example begins in direct mode with the remote system at TCL. The CP-RECV command is entered and the exit key is pressed to exit direct mode.

Example:

```
:CP-RECV BP.REMOTE (M

[CP.032] Direct connect exited from communications port P2.
```

The CP-RECV command entered on the remote system

The CP-SEND command is then used on the local system to send the entire file to the remote system.

```
:CP-SEND BP.LOCAL *
      1   9,146 color
      2   3,433 pick
      3   3,713 rmbi
      4   2,517 chksum
      5     157 setpibs
      6   1,441 tail
      7     709 inputwait
      8   4,258 maxusers
[CP.048] Send of BP.LOCAL completed.
         8 items, 25550 bytes, 7 blocks, 0.27 minutes, 1597 chars/sec.
```

The CP-SEND command entered on the local system

Using CP-RECV to Receive a File from the Remote System

To receive a file from the remote system we use the PowerComm commands CP-SEND and CP-RECV. The CP-SEND command is used on the remote system and the CP-RECV command is entered on the local system. Note that we must use the "M" option with the command invoked on the remote system in order to send data on the same port as the command was entered from.

The following example begins in direct mode with the remote system at TCL. The CP-SEND command is entered and the exit key is pressed to exit direct mode.

Example:

```
:CP-SEND BP.REMOTE * (M

[CP.032] Direct connect exited from communications port P2.
```

The CP-SEND command entered on the remote system

The CP-RECV command is then used on the local system to receive the entire file from the remote system.

```
:CP-RECV BP.LOCAL
      1   9,152 color
      2   3,438 pick
      3   3,718 rmbi
      4   2,524 chksum
      5     165 setpibs
      6   1,446 tail
      7     719 inputwait
      8   4,267 maxusers
[CP.049] Receive of BP.LOCAL completed.
         8 items, 25550 bytes, 7 blocks, 0.25 minutes, 1703 chars/sec.
```

The CP-RECV command entered on the local system

For More Information

For more options and information on using CP-SEND refer to page 115. For more options and information on using CP-RECV refer to page 110.

Using Send and Receive with Additional Options

The following are a few examples showing the output from using the CP-SEND and CP-RECV commands with some additional options.

The following example demonstrates the use of the "C" option to show the communications blocks processed with a send command. The "I" option was used to suppress the item ID's sent. The columns shown with the "C" option are as follows:

- the block number
- total bytes successfully sent or received so far
- current block size
- time elapsed in minutes
- average transfer rate in characters per second

Example:

```
:CP-SEND BP.LOCAL * (C,I
                        1          0     250   0.00   N/A
                        2        250   5,000   0.02   250
                        3      5,250   5,000   0.05  1750
                        4     10,250   5,000   0.10  1708
                        5     15,250   5,000   0.15  1694
                        6     20,250   5,000   0.20  1688
                        7     25,250   5,000   0.25  1683
                        8     30,250   5,000   0.30  1681
                        9     35,250   5,000   0.35  1679
                       10     40,250   5,000   0.40  1677
                       11     45,250   5,000   0.45  1676
                       12     50,250   5,000   0.50  1675
                       13     55,250   5,000   0.55  1674
                       14     60,250     974   0.60  1674
[CP.048] Send of BP.LOCAL completed.
         10 items, 61224 bytes, 14 blocks, 0.60 minutes, 1701 chars/sec.
```

Using CP-SEND with the "C" and "I" options

Example:

This example demonstrates the "C" option without suppressing ID's, the block size override of 10000 and the "$" option to show more information about transfer statistics. The item ID's are on the left as before, and the communications blocks are indented to differentiate them from the item ID's on the left.

```
:CP-SEND BP.LOCAL * (C,$,10000
      1     9,146 color
                        1          0     500   0.00   N/A
      2     3,433 pick
                        2        500  10,000   0.00   N/A
      3     2,445 shp-kill
      4       556 create-index
      5     3,713 rmbi
      6     2,429 :unlink-pibdev
                        3     10,500  10,000   0.10  1750
      7    26,180 set-imap
                        4     20,500  10,000   0.20  1708
                        5     30,500  10,000   0.28  1794
      8     2,181 restore-accounts.dev2
      9     1,496 setpib0
                        6     40,500  10,000   0.38  1761
     10     9,330 buf-map
                        7     50,500  10,000   0.48  1741
                        8     60,500     724   0.57  1779
[CP.048] Send of BP.LOCAL completed.
         10 items, 61224 bytes, 8 blocks, 0.58 minutes, 1749 chars/sec.
[CP.055] 0 overruns, 0 timeouts, 0 resends, 0 sync losses.
```

Using CP-SEND with other options

For More Information

For more options and information on using CP-SEND refer to page 115. For more options and information on using CP-RECV refer to page 110.

Aborting File Transfers with CP-ABORT

There may be times during file transfers that you need to abort the current send or receive in process. To do this the CP-ABORT command is used.

The following example shows the CP-ABORT command being used after exiting direct mode.

Example:

```
:CP-SEND BP.REMOTE * (M

[CP.032] Direct connect exited from communications port P2.
:CP-ABORT
[CP.050] Transfer aborted
```

The following example shows the CP-ABORT command used after the file transfer is underway. The "C" option on the CP-ABORT command clears the sent data in the input buffer. And the "D" option enters direct mode after the abort sequence is sent.

Example:

```
:CP-SEND BP.REMOTE * (M

[CP.032] Direct connect exited from communications port P2.
:CP-RECV BP.LOCAL
      1   9,152 color
      2   3,438 pick

I LSCP.READ.TO.CH.TO:130
!END

:CP-ABORT (C,D
<5012 characters lost>
[CP.050] Transfer aborted
[CP.031] Direct connected to communications port P2.
[CP.100] Remote aborted with the following error:
         [CP.101] The abort command was used.
[CP.050] Transfer aborted
```

Aborting a file transfer in progress

Tip: You can also press ENTER 10 to 15 times, depending upon the number of retries configured, to abort the remote send or receive while you are in direct mode.

For More Information

For more options and information on using CP-ABORT refer to page 101.

Using the Aborted Send and Recover Options

PowerComm provides an option for the CP-RECV command to remember the last item ID received. This is option is not invoked by default because of the additional overhead imposed by saving the item ID in a file after the item is saved.

Important Note: To remember the item ID's the "A" option must be used with the CP-RECEIVE command.

In the following example, the BP.LOCAL file is sent to the remote system at which point we abort the transfer.

Example:

```
:CP-RECV BP.REMOTE (A,M

[CP.032] Direct connect exited from communications port P2.
:CP-SEND BP.LOCAL *
      1   9,146 color
      2   3,433 pick
      3   3,713 rmbi

I LSCP.READ.TO.CH.TO:130
!END
:CP-ABORT (D
[CP.050] Transfer aborted
[CP.031] Direct connected to communications port P2.
[CP.100] Remote aborted with the following error:
         [CP.101] The abort command was used.
[CP.050] Transfer aborted
```

Aborting a file transfer in progress

In direct mode, we invoke the same CP-RECV command, and exit out of direct mode. Now, the CP-SEND command is used with the "A" option to perform the aborted recover. The "A" option asks the remote system which is the last ID it successfully received. In this example, the item "pick" was the last filed ID.

```
:CP-RECV BP.REMOTE (A,M

[CP.032] Direct connect exited from communications port P2.
:CP-SEND BP.LOCAL * (A
Last filed ID = pick
      1   3,713 rmbi
      2   2,517 chksum
      3     157 setpibs
      4   1,441 tail
      5   4,258 maxusers
      6     709 inputwait
[CP.048] Send of BP.LOCAL completed.
         6 items, 12941 bytes, 5 blocks, 0.13 minutes, 1618 chars/sec.
```

Resuming an aborted file transfer

For More Information

For more options and information on using CP-SEND refer to page 115. For more options and information on using CP-RECV refer to page 110.

Using CP-SEND-ASCII and CP-RECV-ASCII to Send and Receive Files

Two commands are provided to communicate with systems that have the ability to receive and send ASCII data. Many BBS's support the ASCII protocol.

For sending files to a remote system the CP-SEND-ASCII command is used. It may be used with optional pacing parameters which control the rate of data transmission.

For receiving ASCII files from a remote computer the CP-RECV-ASCII command is used. Like the ASCII send command, the receive command may also be used with a pacing character.

With both commands, a translation option is available to convert between Pick attribute marks and DOS carriage return and line feeds.

Both of these commands require a reliable connection since no error detection or correction is available.

Using CP-SEND-ASCII to Send Data

The following example demonstrates how you could be in an editor on a remote system or BBS and send a Pick item to the remote system. This example happens to use the AP update editor, it could very well be a BBS running a full screen editor. The "P" option prompts for pacing parameters, the "T" option translates attribute marks to carriage returns, the "V" option views the received echoed data, and the "Z" option suppresses the normal CTRL-Z at the end of the file.

Example:

```
:CP-SEND-ASCII BP.LOCAL BM.GOSUB (P,T,V,Z)
Enter pace character (hex), character delay, line delay: OD,50,1
[CP.109] ASCII upload of file "BP.LOCAL" started.
T = SYSTEM(12)
02 SUB.NAME = "BM.GOSUB"
03 FOR I = 1 TO 1000
04   GOSUB 100
05 NEXT I
06 CRT 1000/((SYSTEM(12) - T) / 1000)
07 STOP
08 100 RETURN      1       124 BM.GOSUB

[CP.107] ASCII upload of 1 items in file "BP.LOCAL" completed.
[CP.110] 124 bytes, 0.13 minutes, 16 chars/sec.
```

Using CP-SEND-ASCII to upload a file while in the editor of a remote system

This next example demonstrates how to send multiple Pick items to a remote system that is capable of receiving one long ASCII file until an EOF (CTRL-Z). The Pick system sends 8 items from the BP.LOCAL file to the remote computer using the "F" option to send it fast and the "N" option to include the Pick item ID's in the data. The "^^" is an enhanced PowerComm itemlist reference that gives you the ability to use TCL commands within the command line to generate a selected list of items. This feature is described in the *TCL Commands Reference* chapter starting on page 95.

Example:

```
:CP-SEND-ASCII BP.LOCAL ^^SELECT BP.LOCAL = "[-84" (F,N

[404] 8 items selected out of 40 items.
[CP.109] ASCII upload of file "BP.LOCAL" started.
      1     1,260 GERMAN-84
      2     1,261 ITALIAN-84
      3     1,261 SPANISH-84
      4     1,261 BRITISH-84
      5     1,285 swiss-fr-84
      6     1,260 FRENCH-84
      7     1,257 USA-84
      8     1,278 swiss-gr-84

[CP.107] ASCII upload of 8 items in file "BP.LOCAL" completed.
[CP.110] 10216 bytes, 0.10 minutes, 1703 chars/sec.
```

Using CP-SEND-ASCII to upload multiple Pick items to a remote system

Later, we can tell the remote system to initiate an ASCII download of the same file, and we use the "N" option to restore the individual Pick items to the file.

```
:CP-RECV-ASCII TEMP3 (N
[CP.124] ASCII download of file "TEMP3" started.
      1     1,260 GERMAN-84
      2     1,261 ITALIAN-84
      3     1,261 SPANISH-84
      4     1,261 BRITISH-84
      5     1,285 swiss-fr-84
      6     1,260 FRENCH-84
      7     1,257 USA-84
      8     1,278 swiss-gr-84
[CP.122] ASCII download of 8 items in file "TEMP3" completed.
[CP.125] 10216 bytes, 0.10 minutes, 1703 chars/sec.
```

Using CP-RECV-ASCII to receive the data as Pick items

For More Information

For more options and information on using CP-SEND-ASCII refer to page 101.

Runtime Options

During execution you may press a key to interrupt the data transfer with the following message displayed:

```
ASCII upload interrupted by keystroke
Select action (C=continue, A=abort, S=skip wait, V=view toggle):
```

The following actions are available:

C	Continue with the upload
A	Abort the transfer
S	Skip the current pace character wait
V	Toggle the view option on or off

Using CP-RECV-ASCII to Receive Data

The following example receives ASCII data from a remote system and uses the "S" option to split records into 5000 byte items.

Example:

```
:CP-RECV-ASCII TEMP2 DL.DATA (S
Enter maximum number of bytes per record: 5000
[CP.124] ASCII download of file "TEMP2" started.
      1     5,000 DL.DATA
      2     5,000 DL.DATA*1
      3     5,000 DL.DATA*2
      4     5,000 DL.DATA*3
      5     5,000 DL.DATA*4
      6     5,000 DL.DATA*5
      7     1,754 DL.DATA*6
[CP.122] ASCII download of 7 items in file "TEMP2" completed.
[CP.125] 31755 bytes, 0.27 minutes, 1985 chars/sec.
```

Using CP-RECV-ASCII to receive split received data into multiple Pick items

The next example demonstrates using the CP-SEND-ASCII and CP-RECV-ASCII commands to send and receive data between two systems running PowerComm. This is not recommended because there is no error detection or correction, but it is shown here to demonstrate some options.

The "*AL" is an enhanced PowerComm itemlist format that selects items in the file into ascending left-justified sequence. This feature is described in the *TCL Commands Reference* chapter starting on page 95. The "F" option sends the data fast with no pacing. The "M" option specifies Me mode. The "N" option includes item names and the "10" waits for 10 seconds before sending of data starts.

On the receiving side, the "C" option is used to clear the input buffer, the "N" option will recognize individual Pick items and the "$" option displays running dots to indicate progress of 1,000 bytes at a time.

The timing in this example is crucial. Using the 10 second delay in the send command gives us enough time to type in the receive command before data transmission starts. Without the "C" option, the receive buffer would include some extra data from the echoed TCL statement and would add extra data to the received items.

Example:

```
:CP-MSG "CP-SEND-ASCII BP.REMOTE *AL (F,M,N,10"
[CP.039] Message sent to communications port P2.

:CP-RECV-ASCII TEMP2 (C,N,$
[CP.124] ASCII download of file "TEMP2" started.

      1        63 BM.CALL

      2       124 BM.GOSUB

      3        10 BM.SUB
...
      4     3,674 BM.TEST
...
      5     2,617 CON
....
      6     3,674 HPII
...............................
      7    31,662 JP
.....
      8     4,854 LAF
....
      9     3,265 REMOTE.SEND
[CP.122] ASCII download of 9 items in file "TEMP2" completed.
[CP.125] 50013 bytes, 0.50 minutes, 1667 chars/sec.
```

Using CP-SEND-ASCII and CP-RECV-ASCII to move data between Pick systems

Runtime Options

During execution you may press a key to interrupt the data transfer with the following message displayed:

```
ASCII download interrupted by keystroke
Select action (C=continue, A=abort, F=finish, S=skip wait, V=view toggle):
```

The following actions are available:

C	Continue with the download
A	Abort the transfer
F	Finish by saving the current received data and quitting
S	Skip the current pace character wait
V	Toggle the view option on or off

Using CP-SEND-PICK and CP-RECV-PICK to Send and Receive Files

Two commands are provided to communicate with Pick systems that do not have the full PowerComm product installed. The remote system, without PowerComm installed, is referred to as the dumb end because it does not have the enhanced communications capabilities of PowerComm.

For sending files to a remote system without PowerComm the CP-SEND-PICK command is used. This command requires the Pick line editor on the remote system. It works by automatically invoking the remote system's line editor with insert mode. The local system then sends the data to the remote system and sends the editor the commands to save the data when completed.

For receiving files from a remote computer without PowerComm installed, the CP-RECV-PICK command is used. This command requires the remote computer to use a small program called "REMOTE.SEND" which is used to send data to the local system. This program can be uploaded using the CP-SEND-PICK command. This step-by-step procedure is shown starting on page 48.

The sending command, CP-SEND-PICK, requires a reliable connection since no error detection or correction is available. The receiving command, CP-RECV-PICK, has minimal error detection and correction.

Note: We plan to expand the capabilities of the *dumb end* in a future release.

Using CP-SEND-PICK to Send Files

To send files to a remote system without PowerComm installed, the CP-SEND-PICK command is used. This command uses the editor on the remote system to receive the data.

The following example demonstrates the CP-SEND-COMMAND with no extra options to move 3 items to the remote system.

Example:

```
:CP-SEND-PICK BP BM.GOSUB BM.CALL BM.SUB
[CP.109] ASCII upload of file "BP" started.
      1       124 BM.GOSUB
      2        63 BM.CALL
      3        10 BM.SUB

[CP.107] ASCII upload of 3 items in file "BP" completed.
[CP.110] 314 bytes, 0.07 minutes, 79 chars/sec.
```

Sending data using CP-SEND-PICK

This example does the same, but the "V" option views the transfer as it takes place.

Example:

```
:CP-SEND-PICK BP BM.GOSUB BM.CALL BM.SUB (V
[CP.109] ASCII upload of file "BP" started.
ED BP BM.GOSUB (SZ)
.F
..DE32000
.F
.I
+T = SYSTEM(12)
+SUB.NAME = "BM.GOSUB"
+FOR I = 1 TO 1000
+  GOSUB 100
+NEXT I
+CRT 1000/((SYSTEM(12) - T) / 1000)
+STOP
+100 RETURN
+
.FI
[221] 'BM.GOSUB' filed.

:      1       124 BM.GOSUB
ED BP BM.CALL (SZ)
.F
..DE32000
.F
.I
+P1 = "ORIGINAL P1"
+CALL BM.SUB(P1, "THIS IS ANOTHER P1")
+CRT P1
+
.FI
[221] 'BM.CALL' filed.

:      2        63 BM.CALL
ED BP BM.SUB (SZ)
.F
..DE32000
.F
.I
+SUB
+RETURN
+
.FI
[221] 'BM.SUB' filed.

:      3        10 BM.SUB

[CP.107] ASCII upload of 3 items in file "BP" completed.
[CP.110] 314 bytes, 0.08 minutes, 63 chars/sec.
```

Sending data using CP-SEND-PICK with the view option

The following example uses several options to transmit an item to the remote system. The "O" option turns off echo on the remote system. This speeds up the transfer since the comm port does not have to process both outgoing and incoming data at the same time. The "V" option views the transfer. The "F" option uses the fast mode by sending the data as one long insert. (The long insert is only available on AP versions of the editor).

The "R" option is used to rename the target file to the stacked input of "BP.REMOTE". The "Z" option makes the editor use its "S?" command to show the item size. Note that the item size reported by the editor will mismatch the item size shown transmitted because the editor includes the item ID and a few other bytes that PowerComm does not include.

Example:

```
:CP-SEND-PICK BP.LOCAL BIG (O,V,F,R,Z) {BP.REMOTE}

:ECHO (I)
:[CP.109] ASCII upload of file "BP.LOCAL" started.

.
.
.
.
+
+
.
.
item size is 31676 bytes.
.
      1    31,662 BIG
[221] 'BIG' filed.
:
:
[CP.107] ASCII upload of 1 items in file "BP.LOCAL" completed.
[CP.110] 31727 bytes, 0.32 minutes, 1669 chars/sec.
```

Sending data using CP-SEND-PICK with several options

Runtime Options

During execution you may press a key to interrupt the data transfer with the following message displayed:

```
ASCII upload interrupted by keystroke
Select action (C=continue, A=abort, S=skip wait, V=view toggle):
```

The following actions are available:

C	Continue with the upload
A	Abort the transfer
S	Skip the current pace character wait
V	Toggle the view option on or off

Tips for Using CP-SEND-PICK:

The following are some tips for using CP-SEND-PICK for best results.

- Use the "F" option on AP for fastest speed.

  On AP, this will result in the data being transmitted at full speed. Make sure your modem connection is running at full speed or you are using flow control. By default, the XON/XOFF flow control is turned off when starting a port. When using the "F" option, you may want to re-enable flow control or use the "X" option with CP-START. You can also slow down the baud rate if you have problems receiving the data on the remote.

- Use the "O" option to turn off echo.

  This eliminates the echoing of the editor input commands which can speed up the transfer by reducing the amount of I/O the communications port is processing. When the "O" option is used, it executes the TCL command "ECHO" on the remote system with the "I" option. On R83, this always turns off echo. On AP, the option is ignored. It is possible on AP to get out of sync with the echo commands. Therefore, if you have aborted a previous upload and the echo was already off, you may inadvertently turn it back on by using the "O" option.

  To turn always turn on echo on R83, enter "ECHO (L". On AP, the echo command is a toggle. Use "ECHO" at TCL if it is off in order to turn it back on.

- Enable 8 bit input using XCS-ON.

  In order to receive a full 8 bits, it is necessary to enable 8 bit input by using the XCS-ON command at TCL. This is required if you use the "F" option.

Using CP-RECV-PICK to Receive Files

Before using the CP-RECV-PICK command it is necessary to upload a small program to the remote system. These step-by-step instructions start on page 48.

When using the REMOTE.SEND program on the remote system, the local system must be capable of receiving continuous streaming data because each Pick item is sent as one long packet. If you have a reliable connection and the modems are set up properly to handle flow control, this should not be a problem.

The following example starts out with the user in direct mode on the other system. This example shows how to move a few programs starting with "BM." from the remote system to the local system using REMOTE.SEND.

The "H" option instructs the REMOTE.SEND program to send the data in hex format. If your system does not support 8 bit printing with Basic, you will have to use this option. It will take twice as long since the number of bytes is doubled using hex mode. Hex mode is also used when the data may contain control characters that could stop the transmission.

When the program displays "*Begin your download now*...", you press the exit key to exit direct mode and enter the CP-RECV-PICK command with a local file name.

Example:

```
:SELECT BP.REMOTE = "BM.]"

[404] 4 items selected out of 6 items.
>REMOTE.SEND
Enter file name to send: BP.REMOTE
Enter options (H=hex): H
To quit download at any time, enter 'Q'.
Begin your download now...
[CP.032] Direct connect exited from communications port P2.

:CP-RECV-PICK BP.LOCAL
[CP.124] ASCII download of file "BP.LOCAL" started.
      1       124 BM.GOSUB
      2        63 BM.CALL
      3        10 BM.SUB
      4     3,674 BM.TEST
[CP.122] ASCII download of 4 items in file "BP.LOCAL" completed.
[CP.125] 3990 bytes, 0.08 minutes, 798 chars/sec.
```

Receiving data from a remote system using CP-RECV-PICK

This next example does the same thing but without having to use direct mode to invoke the program. You must already be at TCL on the remote computer. The automatic "A" option is used to prompt for the information locally.

Example:

```
:CP-RECV-PICK BP.LOCAL (A)
Enter remote select statement or * for all: SELECT BP.REMOTE = "BM.]"
Enter remote file name: BP.REMOTE
Enter remote download options (H=hex): H
Enter maximum number of seconds to wait for remote: 10
:SELECT BP.REMOTE = "BM.]"

[404] 4 items selected out of 6 items.
>REMOTE.SEND
Enter file name to send: BP.REMOTE
Enter options (H=hex): H&
To quit download at any time, enter 'Q'.
Begin your download now...<autostart>
[CP.124] ASCII download of file "BP.LOCAL" started.
      1       124 BM.GOSUB
      2        63 BM.CALL
      3        10 BM.SUB
      4     3,674 BM.TEST
[CP.122] ASCII download of 4 items in file "BP.LOCAL" completed.
[CP.125] 3990 bytes, 0.08 minutes, 798 chars/sec.
```

Receiving data from a remote system using CP-RECV-PICK with the "A" option

This next example does the same thing but supplying all the parameters on the command line. This command line feature is described in the *TCL Commands Reference* chapter starting on page 95.

Example:

```
:CP-RECV-PICK BP.LOCAL (A) {SELECT BP.REMOTE = "BM.]"} {BP.REMOTE} {H} {10}
:SELECT BP.REMOTE = "BM.]"

[404] 4 items selected out of 6 items.
>REMOTE.SEND
Enter file name to send: BP.REMOTE
Enter options (H=hex): H&
To quit download at any time, enter 'Q'.
Begin your download now...<autostart>
[CP.124] ASCII download of file "BP.LOCAL" started.
      1       124 BM.GOSUB
      2        63 BM.CALL
      3        10 BM.SUB
      4     3,674 BM.TEST
[CP.122] ASCII download of 4 items in file "BP.LOCAL" completed.
[CP.125] 3990 bytes, 0.08 minutes, 798 chars/sec.
```

Receiving data from a remote system using CP-RECV-PICK with the "A" option and command line parameters

The next example moves the entire contents of the "KEYBOARDS" file to the local system and uses the "I" option to suppress item-ID's as it goes.

Example:

```
:CP-RECV-PICK BP.LOCAL (A,I) {*} {KEYBOARDS} {H} {10}
:REMOTE.SEND
Enter file name to send: KEYBOARDS
Enter options (H=hex): H&
To quit download at any time, enter 'Q'.
Begin your download now...<autostart>
[CP.124] ASCII download of file "BP.LOCAL" started.
[CP.122] ASCII download of 25 items in file "BP.LOCAL" completed.
[CP.125] 31875 bytes, 0.62 minutes, 861 chars/sec.
```

Receiving data from a remote system using CP-RECV-PICK with the "A" option and command line parameters

Runtime Options

During execution you may press a key to interrupt the data transfer with the following message displayed:

```
ASCII download interrupted by keystroke
Select action (C=continue, A=abort, S=sync):
```

The following actions are available:

C	Continue with the download
A	Abort the transfer
S	Synchronize by requesting the last block over again

Uploading the Remote Send Program

These steps are used to upload the "REMOTE.SEND" program to the remote system. It also demonstrates the use of the CP-SEND-PICK command.

▸ **Follow these steps to upload the "REMOTE.SEND" program:**

1. Enter direct mode and create a file to receive the REMOTE.SEND program. If you already have a file you want to use, you may skip to step 4. Otherwise, the file name to contain the "REMOTE.SEND" program is called BP.REMOTE in this demonstration.

```
:CP-DIR
[CP.031] Direct connected to communications port P2.

:CREATE-FILE BP.REMOTE 1 1 (C

[417] file 'BP.REMOTE' created; base = 27419, modulo = 1

[417] file 'BP.REMOTE' created; base = 233838, modulo = 1
```

Creating the remote file

2. On R83 systems, if the "C" option on the CREATE-FILE command does not work, you may need to change the "D" on the file definition item from a "D" to a "DC" using the editor as shown below. The "DC" file designation is not needed for AP systems.

```
:ED MD BP.REMOTE
top
.
001 D
.R
001 DC
.FI
[221] 'BP.REMOTE' filed.
```

Using the editor to change the file type on R83 systems

3. After creating the file, exit direct mode by pressing the exit key.

```
 [CP.032] Direct connect exited from communications port P2.
```

4. Use the CP-SEND-PICK command to send over the "REMOTE.SEND" BASIC program. The "R" option is used to prompt for the remote file name. In this case, "BP.REMOTE" is used. If you are uploading to a different file, enter your file name instead. The "V" option views the progress of the CP-SEND-PICK command as it runs.

```
:CP-SEND-PICK LSCP,BP.UPLOAD REMOTE.SEND (R,V
Enter remote file name to upload to: BP.REMOTE
[CP.109] ASCII upload of file "LSCP,BP.UPLOAD" started.
ED BP.REMOTE REMOTE.SEND (SZ)
.F
..DE32000
.F
.I
+* simple ASCII download routine
+* +----------------------------------------------------------------------+
+* | (C) Copyright 1995 Logitek Systems, San Marcos, CA USA               |
.
.
.
+  GOSUB 500 ;* get response
+UNTIL OK.FLAG OR ABORT.FLAG DO
+REPEAT
+RETURN
+
.FI
[221] 'REMOTE.SEND' filed.

:      1     3,265 REMOTE.SEND

[CP.107] ASCII upload of 1 items in file "LSCP,BP.UPLOAD" completed.
[CP.110] 3315 bytes, 0.37 minutes, 151 chars/sec.
```

Sending the program to the remote using CP-SEND-PICK

5. Enter direct mode and compile and catalog the "REMOTE.SEND" program.

```
:CP-DIR
[CP.031] Direct connected to communications port P2.

:BASIC BP.REMOTE REMOTE.SEND
REMOTE.SEND
..............
[241] Successful compile   1 frame(s) used.

:CATALOG BP.REMOTE REMOTE.SEND
[244] 'REMOTE.SEND' cataloged
```

Compiling and cataloging the program

6. Exit direct mode by pressing the exit key.

```
[CP.032] Direct connect exited from communications port P2.
```

The program is now uploaded, compiled, cataloged and ready to use.

For More Information

For more options and information on using CP-SEND-PICK refer to page 117. And for more options and information on using CP-RECV-PICK refer to page 112.

Using CP-STAT-FILE to Check Sent or Received Files

The CP-STAT-FILE command is used to provide a platform independent way of checking files. This command computes a length and checksum for each item specified. The technique used to computer the checksum is the same on R83 and AP systems.

The default display shows the item number, the item size, the checksum in hex or decimal, and the item ID. The item sizes and checksums will also include the item ID, unless the "D" option is used.

The following example demonstrates CP-STAT-FILE using the 16 bit and hex display options.

Example:

```
:CP-STAT-FILE BP * (16,H
      1       219            F480 GL.AUDIT
      2       318            8B3E GL.POST
      3     3,484            62BA GL.POST.1
      4     3,094            F028 GL.POST.2
      5     3,094            B028 GL.POST.3
      6     4,168            F46A GL.POST.4
      7     3,670            746C GL.POST.5
      8    31,015            E1C9 GL.POST.DETAIL
      9     1,112            2E8C GL.POST.SUMMARY
     10       161            4A03 GL.RPTS
     11       616             3E2 GL.RPTS.1
     12       831            842A GL.RPTS.2
     13     1,763            6EC1 GL.RPTS.3
     14       894            60F2 GL.RPTS.4
     15     1,106            D895 GL.RPTS.5
     16    12,651            4EED GL.RPTS.DETAIL
     17       125            35E1 GL.SUMMARY
[CP.138] Stats of file "BP":
         17 items; 68,321 bytes; 16 bit checksum total = 8FB18.
```

Using CP-STAT-FILE to check file transmission

The following example demonstrates CP-STAT-FILE using the 32 bit checksum option. The "D" option restricts the computation of length and checksum to the data item only and not the item ID. The "$" option was used to display more information.

Example:

```
:CP-STAT-FILE BP (D,32,$
      1       210   3,784,126,494 GL.AUDIT
      2       310     828,451,654 GL.POST
      3     3,474   3,930,124,892 GL.POST.1
      4     3,084   3,046,251,600 GL.POST.2
      5     3,084   3,046,251,600 GL.POST.3
      6     4,158   2,259,359,086 GL.POST.4
      7     3,660   2,567,795,850 GL.POST.5
      8    31,000   2,517,831,703 GL.POST.DETAIL
      9     1,096   1,912,422,640 GL.POST.SUMMARY
     10       153   1,580,840,181 GL.RPTS
     11       606   2,642,016,421 GL.RPTS.1
     12       821   2,703,637,439 GL.RPTS.2
     13     1,753   2,900,435,217 GL.RPTS.3
     14       884   1,676,250,562 GL.RPTS.4
     15     1,096   1,912,422,640 GL.RPTS.5
     16    12,636   3,565,414,936 GL.RPTS.DETAIL
     17       114   1,421,124,793 GL.SUMMARY
[CP.138] Stats of file "BP":
         17 items; 68,139 bytes; 32 bit checksum total = 42,294,757,708.
[CP.139] 114 minimum size; 4,008 average size; 31,000 maximum size.
```

Using CP-STAT-FILE to check file transmission

On R83 systems, using larger checksum options, the final total may overflow yielding unpredictable results. Therefore, on R83 systems, we recommend using a smaller checksum value if you want the final total to be accurate.

For More Information

For more options and information on using CP-STAT-FILE refer to page 123.

Using CP-MSG to Send a TCL Command

At times it may be more convenient to send TCL commands to the remote system without having to enter direct mode, type the command, and exit direct mode. The PowerComm CP-MSG command can be used.

Instead of these 4 steps:	Do these 2 steps:
CP-DIR on remote: CP-RECV BP.REMOTE (M exit direct mode CP-SEND BP.LOCAL	CP-MSG "CP-RECV BP.REMOTE (M" CP-SEND BP.LOCAL

The following example demonstrates the two steps when using the CP-MSG command.

Example:

```
:CP-MSG "CP-RECV BP.REMOTE (M"
[CP.039] Message sent to communications port P2.

:CP-SEND BP.LOCAL *
      1   9,146 color
      2   3,433 pick
      3   3,713 rmbi
      4   2,517 chksum
      5     157 setpibs
      6   1,441 tail
      7   4,258 maxusers
      8     709 INPUTWAIT
[CP.048] Send of BP.LOCAL completed.
         8 items, 25550 bytes, 7 blocks, 0.25 minutes, 1703 chars/sec.
```

Using CP-MSG to invoke a remote file receive command

For More Information

For more options and information on using CP-MSG refer to page 106.

Using CP-PACKAGE to Help Move Groups of Files or Accounts

The Package utility, CP-PACKAGE, is a tool to provide an easier way to move multiple files and accounts between computer systems. It is used to create a 'package' of data and embedded TCL commands. The data packaged may be any number of dict or data level files or accounts. Embedded TCL commands may be included in the package and they can be executed during add or extract modes. These TCL commands may be used to automate GET-LISTS or compiles routinely performed on transferred programs or data.

After the package is created, you can use CP-SEND and CP-RECV to move the 'package' to the remote system. Once on the remote system, the CP-PACKAGE program is run with the 'e'xtract option to move the data into the target files or accounts.

The program works in one of three distinct modes:

- The 'a'dd mode is used when adding items from source data files while creating a 'package' on the source computer system.
- The 'e'xtract mode is used when extracting data from the package to the target destination files on the target computer system.
- The 'v'iew mode runs like the 'e'xtract mode but only views the commands that have been stored in the package.

The following describes advantages and disadvantages of using CP-PACKAGE.

Advantages

- Provides a way to move multiple files and accounts at a time.
- Allows embedded TCL commands to execute when data is extracted.
- Provides a record of uploads when using a command item to control the packaging.
- Easier to check that all of the data you uploaded has arrived intact.
- By sending only one file, it is easier to send up missing pieces using the aborted upload recovery features.

Disadvantages

- Since the packaged data is a copy of the original data, it can take use up twice the disk space on the sending system.
- Unless you use the "K" option while extracting the data on the target system, it can also use up twice the disk space on the target system.

When you need to send just a single file, you may find the three step process of creating a package, sending it, and extracting it, more work than you wanted. It depends on the situation and your need for an audit trail.

Tips for Using Package

The following are some tips and techniques we have found useful over the years of using the package utility.

➤ Use command items to control what gets packaged.

This allows you to have an audit trail of what you are sending. Although you can specify commands on the command line, they are part of the package and not the original item that was used to create the package.

➤ Use multiple command items when sending multiple accounts.

If you dump all the accounts you want to send in one package name, you may find that when you get on the remote system that you need to extract only one account. Right now, package does not have controls to skip commands during extract mode. In other words, package does not allow you to skip the first account, but extract the second.

➤ Choose a naming convention.

We like to name our packages with a preceding "U" or "D" to designate an upload or download. Followed by a 6 digit date; 012495. Then if we send more than one package in a day, we add a letter at the end starting with A, then B, etc.

After a while, our package file "PKGS" looks something like this:

```
D031593
D012895
U011894
U041194
U041194A
U041594
U041894
U041894A
U060294
U060294A
U060294B
U070794
```

➤ Use the "K" option to limit disk space usage on the remote.

To limit the amount of disk space on the remote system, use the "K" option when you are extracting the data. This option removes the 'package' items as it goes. In addition, the "K" option will not remove the command item that controls the packaging leaving you with a record of the contents of the package.

Before using the "K" option, make sure all the pieces are present by using the "V" option to view the contents of the package. Missing items will show up as errors.

➤ Select your package items for subsequent uploading in descending right-justified order.

You can use the "L" option with CP-PACKAGE to automatically generate such a list. Having the list sorted in descending right-justified order will give you an idea of what's left as you watch the item ID's scroll down the screen during a CP-SEND upload.

➤ Remove package data from your package file, leaving behind command items.

Once the package data has been uploaded, you can recover disk space by removing the package data, but leaving behind the command items. To remove package data from your package but leave the command items, use one of the following techniques:

```
SELECT PKGS = "[*]"
DELETE PKGS
```

or

```
CP-PACKAGE PKGS * (V,K
```

What's left are just the command items you used to create the package data.

➤ Size your package file correctly.

On AP, most all of the package data will be indirect pointers because of their size. Assuming most of the items are around 5K in length, you can size your primary space relatively small.

On R83, all items are stored directly. If the primary file space is to small, you will have multiple large items per group, slowing down the overall progress of adding or extracting from packages.

CP-PACKAGE Examples

The following examples in this section shows how CP-PACKAGE can be used to accomplish a variety of useful functions.

This first example illustrates the use and subsequent output from using a command item. It is used to move over a list of program starting with "GL." to a remote system, and once there, compile them.

The following command item was created in advance and called "GL" and saved in the "PKGS" file.

```
GL
001 * MOVE NEW GL PROGRAMS TO REMOTES
002 *
003 TCLA SSELECT BP = "GL.]"
004 DATA BP
005 TCLE SSELECT BP = "GL.]"
006 TCLE BASIC BP
```

The "TCLA" command executes TCL commands during "A"dd, "TCLE" during extract. The "DATA" command retrieves the items in the "BP" file using the results from the select command.

This example uses the "A" option to add to the package named "GL", and the "C" option to tell CP-PACKAGE the commands are in the "GL" item in the "PKGS" file. The "L" option will select the "PKGS" file and prompt for a save list name. The "10000" overrides the default item size target of 5000 bytes.

Example:

```
:CP-PACKAGE PKGS GL (A,C,L,10000
[CP.111] Creating package "GL" in file "PKGS".
[CP.118] 0 commands, 0 entries, 0 bytes, 0 package items.
Average item size target is 10000 bytes.
TCLA> SSELECT BP = "GL.]"

[404] 17 items selected out of 134 items.
DATA BP
##### ID                                    Bytes      Total Indexes External
    1 BP GL.AUDIT                             210        395       1        0
    2 BP GL.POST                              310        723       1        0
    3 BP GL.POST.1                          3,474      4,218       1        0
    4 BP GL.POST.2                          3,084      7,323       1        0
    5 BP GL.POST.3                          3,084     10,428       2        0
    6 BP GL.POST.4                          4,158     14,607       2        0
    7 BP GL.POST.5                          3,660     18,288       3        0
    8 BP GL.POST.DETAIL                    31,000     49,316       3        1
    9 BP GL.POST.SUMMARY                    1,096     50,439       3        1
   10 BP GL.RPTS                              153     50,610       3        1
   11 BP GL.RPTS.1                            606     51,236       3        1
   12 BP GL.RPTS.2                            821     52,077       3        1
   13 BP GL.RPTS.3                          1,753     53,851       3        1
   14 BP GL.RPTS.4                            884     54,755       3        1
   15 BP GL.RPTS.5                          1,096     55,872       4        1
   16 BP GL.RPTS.DETAIL                    12,636     68,536       4        2
   17 BP GL.SUMMARY                           114     68,671       4        2
[CP.118] 6 commands, 28 entries, 68805 bytes, 4 package items.
[CP.116] Package "GL" in file "PKGS" added.
Selecting list> SSELECT PKGS = "GL" "GL*]" USING DICT LSCP BY-DSND R0

[404] 8 items selected out of 185 items.
SAVE-LIST UPLOAD.GL
Saving list> SAVE-LIST UPLOAD.GL

List 'UPLOAD.GL' in file 'pointer-file' saved.
```

Using CP-PACKAGE with a command item

The results of the packaging activity is show below:

```
:SORT PKGS = "GL]" SIZE

PKGS...... size....

GL              124
GL*              40
GL*#1         31016
GL*#2         12652
GL*1           7338
GL*2           7299
GL*3           9163
GL*4           1430

[405] 8 items listed out of 185 items.
```

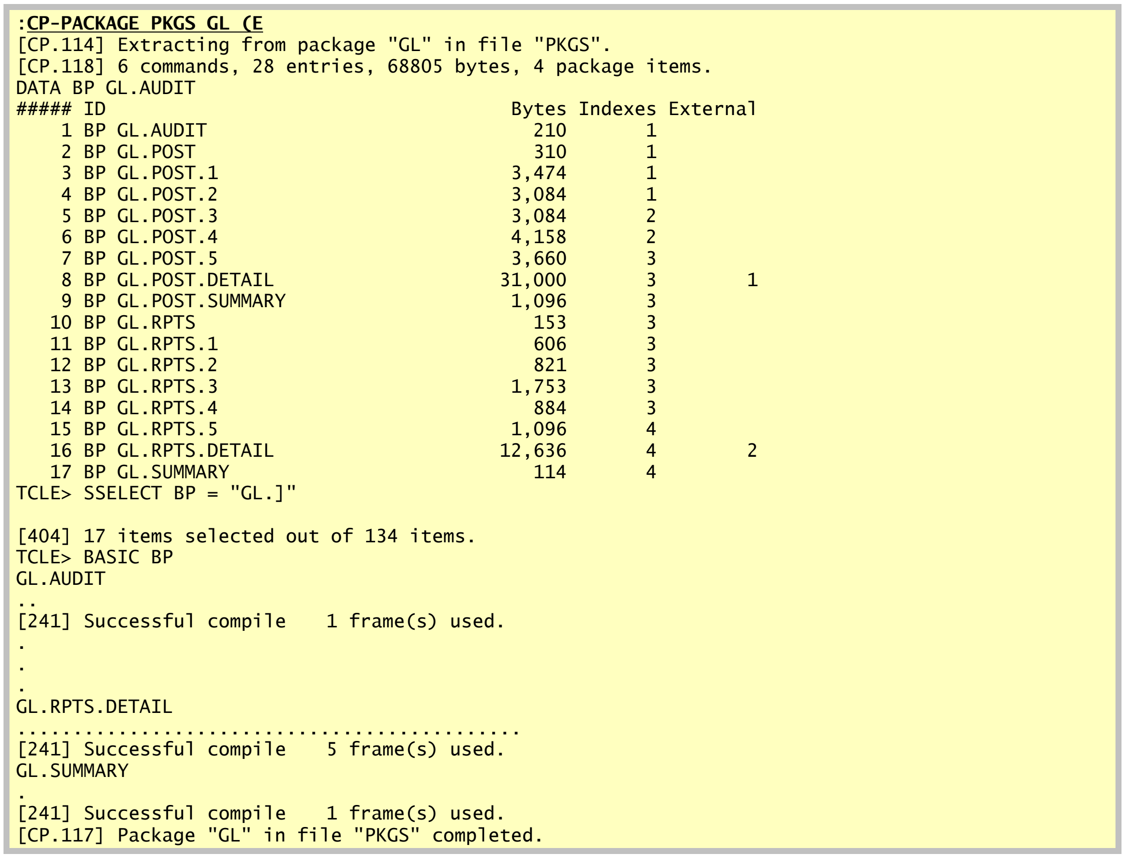
The two items containing "#" are stored separate from the rest of the data because its size exceeded the target item size of 10000 bytes.

After uploading the package items, the following shows the "V" option to verify the items on the remote end.

```
:CP-PACKAGE PKGS GL (V
[CP.113] Viewing package "GL" in file "PKGS".
[CP.118] 6 commands, 28 entries, 68805 bytes, 4 package items.
V-> $S 11:45:09  14 Feb 1995
V-> $$ * MOVE NEW GL PROGRAMS TO REMOTES
V-> $$ *
V-> $$ TCLA SSELECT BP = "GL.]"
V-> $$ DATA BP
V-> $$ TCLE SSELECT BP = "GL.]"
V-> $$ TCLE BASIC BP
V-> $E 11:45:09  14 Feb 1995
[CP.117] Package "GL" in file "PKGS" completed.
```

Verifying the package using the "V" option

The final step is to extract the data using the "E" option. You can see the results of the TCL commands executing during extract.



```
:CP-PACKAGE PKGS GL (E
[CP.114] Extracting from package "GL" in file "PKGS".
[CP.118] 6 commands, 28 entries, 68805 bytes, 4 package items.
DATA BP GL.AUDIT
##### ID                                    Bytes Indexes External
    1 BP GL.AUDIT                             210       1
    2 BP GL.POST                              310       1
    3 BP GL.POST.1                          3,474       1
    4 BP GL.POST.2                          3,084       1
    5 BP GL.POST.3                          3,084       2
    6 BP GL.POST.4                          4,158       2
    7 BP GL.POST.5                          3,660       3
    8 BP GL.POST.DETAIL                    31,000       3        1
    9 BP GL.POST.SUMMARY                    1,096       3
   10 BP GL.RPTS                              153       3
   11 BP GL.RPTS.1                            606       3
   12 BP GL.RPTS.2                            821       3
   13 BP GL.RPTS.3                          1,753       3
   14 BP GL.RPTS.4                            884       3
   15 BP GL.RPTS.5                          1,096       4
   16 BP GL.RPTS.DETAIL                    12,636       4        2
   17 BP GL.SUMMARY                           114       4
TCLE> SSELECT BP = "GL.]"

[404] 17 items selected out of 134 items.
TCLE> BASIC BP
GL.AUDIT
..
[241] Successful compile   1 frame(s) used.
.
.
.
GL.RPTS.DETAIL
.............................................
[241] Successful compile   5 frame(s) used.
GL.SUMMARY
.
[241] Successful compile   1 frame(s) used.
[CP.117] Package "GL" in file "PKGS" completed.
```

Extracting from the package using the "E" option

The next example uses the command line to create a package containing the "PA" account. The "L" option automatically selects a list at the end of the packaging. The "{PA.LIST}" is stacked input to automatically answer the "SAVE-LIST" question from using the "L" option.

Example:

```
:CP-PACKAGE PKGS PA (A,L) ACCOUNT PA {PA.LIST}
[CP.111] Creating package "PA" in file "PKGS".
[CP.118] 0 commands, 0 entries, 0 bytes, 0 package items.
Average item size target is 5000 bytes.
PA
PA > entity
PA > entity > entity
PA > codes
PA > codes > codes
PA > country
PA > country > country
PA > category
PA > category > category
PA > location
PA > location > location
PA > zcf
PA > zcf > zcf
PA > journal
PA > journal > journal
PA > bp
PA > bp > bp
PA > statement
PA > statement > statement
[CP.118] 1 commands, 2806 entries, 172022 bytes, 36 package items.
[CP.116] Package "PA" in file "PKGS" added.
Selecting list> SSELECT PKGS = "PA" "PA*]" USING DICT LSCP BY-DSND RO

[404] 37 items selected out of 180 items.
Saving list> SAVE-LIST PA.LIST

List 'PA.LIST' in file 'pointer-file' saved.
```

Creating a package using a command line command

Once uploaded, the following shows the package being extracted. The "E" option extracts, the "K" option deletes the package items as it goes conserving disk space. The "R" option prompts for a rename account opportunity.

```
:CP-PACKAGE PKGS PA (E,K,R
[CP.114] Extracting from package "PA" in file "PKGS".
[CP.118] 1 commands, 2806 entries, 172022 bytes, 36 package items.
PA
Enter new account name, <ENTER> to quit: PA.NEW
PA.NEW 239376,29
PA.NEW > entity 239364,3
PA.NEW > entity > entity 232782,53
PA.NEW > codes 238713,1
PA.NEW > codes > codes 239641,1
PA.NEW > country 239639,1
PA.NEW > country > country 239645,7
PA.NEW > category 239643,1
PA.NEW > category > category 239655,1
PA.NEW > location 239653,1
PA.NEW > location > location 239659,1
PA.NEW > zcf 239657,1
PA.NEW > zcf > zcf 239665,17
PA.NEW > journal 239687,3
PA.NEW > journal > journal 239693,1
PA.NEW > bp 239208,1
PA.NEW > bp > bp 239698,3
PA.NEW > statement 239225,1
PA.NEW > statement > statement 239702,1
[CP.117] Package "PA" in file "PKGS" completed.
```

Extracting and renaming the account from the package

The account has now been restored to the new name of "PA.NEW".

For More Information

Refer to the command information about CP-PACKAGE starting on page 107. For technical information about CP-PACKAGE, refer to page 66.

Using CP-STATUS-PORT to View Current Port Statuses

The CP-STATUS-PORT can be used to show the current status of a port on the system. It does not have to be a comm port. If it is, additional information will be displayed.

The command, by default, will display 4 blocks of information:

- the current communications port parameters including baud rate, parity, stop bits, word length, type-ahead status, flow control status, extended character set status, and data carrier detect status.
- the output from the WHERE TCL command.
- the interpretation of the PIB status word from the WHERE command.
- the output from the CP-LIST command.

The following example is the output from using the CP-STATUS-PORT command showing all 4 blocks of information available. The "$" option is used to show additional information from the CP-LIST command.

Example:

```
:CP-STATUS-PORT 2 ($
Port: 2
Current Port Communications Parameters
  Baud  Parity  Stop bits  Word length  TA  FC  XCS DCD
 19200  NONE        1          8        ON  OFF ON  OFF
Where Results
 002 000722 AF00 00EFC5      LSCP.AP.START:154 br.conv:235
PIB Status: AF00
A 1010  Active         Sleeping                     Echo Off
F 1111
0 0000  DCD Off                       TA On          XON/XOFF Off
0 0000
CP-LIST Results
      Port Name........... Mode..... Status Input..(size) Output.(size) Entry#
  16 =>  2 P2              DIR       Active   3,953( 20k)       0( 20k)    0
                           Baud.( Link) P S W  (Original settings)
                           19200        N 1 8   19200 N 1 8
                           Original Statuses   Start
                           TA  FC  XCS DCD     Options
                           ON  ON  OFF OFF
                           Started......... Attached........ Mode...........
                           03-05 10:55:44am 03-05 03:08:36pm 03-05 03:08:36pm
```

For More Information

Refer to the command information about CP-STATUS-PORT starting on page 124. And, to the section *CP-STATUS-PORT*, in the *Technical Information* appendix, starting on page 75.

Using CP-VIEW-TCL to Continuously Execute and Display TCL Commands

The CP-VIEW-TCL command will allow you to run several TCL commands together, save the output and compare the output highlighting changed lines from one execution to the next. This can be helpful when you want to continuously monitor a TCL command.

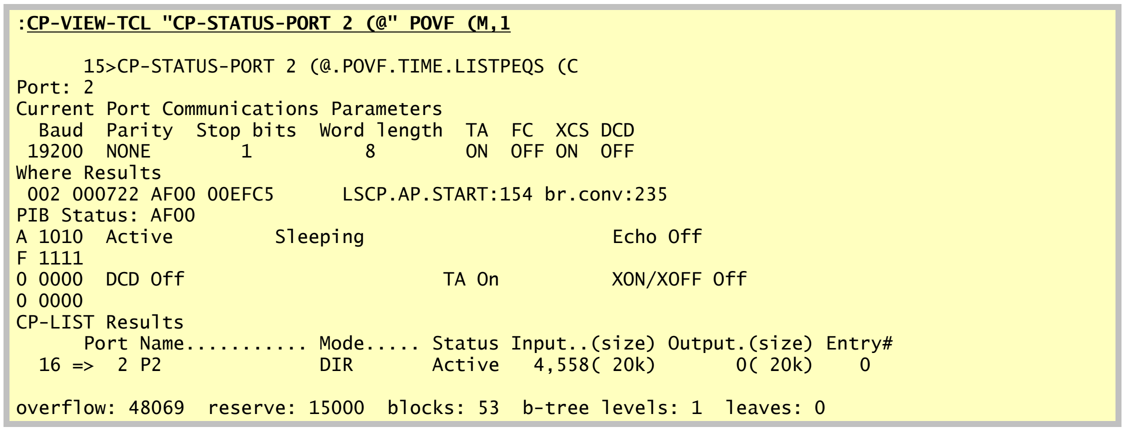
Note: Multiple TCL commands are enclosed in quotes and separated by spaces. The "M" option is also required if multiple TCL commands are specified.

This command may be combined with the CP-STATUS-PORT command to give you a constant update of information about a port on the system.

The following example runs the CP-VIEW-TCL command with the CP-STATUS-PORT command to continuously show the status of a communications port. The option "@" is used to suppress the half intensity that normally is shown with the CP-STATUS-PORT command. After the CP-STATUS-PORT command is executed, the POVF command is executed.

The results from both commands are displayed on the screen. The CP-VIEW-TCL command will sleep for 1 second between runs. To exit the output view, press a key other than "C", "P", or "D".

Example:



```
:CP-VIEW-TCL "CP-STATUS-PORT 2 (@" POVF (M,1

      15>CP-STATUS-PORT 2 (@.POVF.TIME.LISTPEQS (C
Port: 2
Current Port Communications Parameters
  Baud  Parity  Stop bits  Word length  TA  FC  XCS DCD
 19200  NONE        1          8        ON  OFF ON  OFF
Where Results
 002 000722 AF00 00EFC5      LSCP.AP.START:154 br.conv:235
PIB Status: AF00
A 1010  Active         Sleeping                      Echo Off
F 1111
0 0000  DCD Off                        TA On          XON/XOFF Off
0 0000
CP-LIST Results
      Port Name........... Mode..... Status Input..(size) Output.(size) Entry#
  16 =>  2 P2              DIR       Active   4,558( 20k)       0( 20k)    0

overflow: 48069  reserve: 15000  blocks: 53  b-tree levels: 1  leaves: 0
```

Runtime Options

During execution you may press a key to interrupt the display. The following options are available by pressing the following keys:

P	Toggle pause mode
C	Clear the screen
D	Prompt for a new sleep delay

Any other key terminates the display.

For More Information

Refer to the command information about CP-VIEW-TCL starting on page 124.

Caution: Do not run TCL commands with this command that create output that cannot be captured or output that would exceed the screen size. The size of the display is determined by the page width and page depth settings in the TERM command. It has a maximum screen depth limit of 100 lines.

Using the CP-SEND-XMODEM/CP-RECV-XMODEM Commands

The following example sort selects the "TRANS" file into ascending right justified order and sends to the remote system using "CRC" packet check (if available). The "BB" options display the block information indented on the right.

Example:

```
:CP-SEND-XMODEM TRANS *AR (BB,C

[404] 4 items selected out of 4 items.
                    XModem-CRC
      1       658              1
                        1        128   0.02   128
                        2        256   0.02   256
                        3        384   0.03   192
                        4        512   0.03   256
                        5        640   0.03   320
      2       482              2
                        6        768   0.05   256
                        7        896   0.05   299
                        8      1,024   0.05   341
      3       362              3
                        9      1,152   0.05   384
                       10      1,280   0.07   320
                       11      1,408   0.07   352
      4       754              4
                       12      1,536   0.07   384
                       13      1,664   0.08   333
                       14      1,792   0.08   358
                       15      1,920   0.08   384
                       16      2,048   0.10   341
                       17      2,176   0.10   363
                       18      2,304   0.10   384
                    EOT
[CP.146] XModem send of TRANS completed.
         4 items, 2304 bytes, 18 blocks, 0.13 minutes, 288 chars/sec.
[CP.148] Result code: 0, Completed
         Protocol: XModem-CRC
```

Using CP-SEND-XMODEM to send data

The following example receives a XModem file into the "TRANS" file requesting "CRC" packet check with the initial item ID of "INCOMING" and splitting of items at 1000 bytes.

Example:

```
:CP-RECV-XMODEM TRANS INCOMING (C,S) {1000}
      1     1,000 INCOMING
      2     1,000 INCOMING*1
      3     1,000 INCOMING*2
      4     1,000 INCOMING*3
      5     1,000 INCOMING*4
      6     1,000 INCOMING*5
      7     1,000 INCOMING*6
      8     1,000 INCOMING*7
      9     1,000 INCOMING*8
     10       344 INCOMING*9
[CP.151] XModem receive of TRANS completed.
         10 items, 9344 bytes, 73 blocks, 0.37 minutes, 425 chars/sec.
[CP.148] Result code: 0, Completed
         Protocol: XModem-CRC
[CP.155] Error stats:
           Timeouts              0
           Sequence errors       0
           Checksum/CRC errors   0
           Short data errors     0
           Nak'd blocks          0
```

Using CP-RECV-XMODEM to receive data

The following example uses XModem to send a file to AP system. The example starts out by going into direct mode with the other system. The other system's port settings are modified for a full 8 bit no software flow control setup. The "N" option sends the item names and the "2000" option overrides the packet size to 2000 bytes.

Example:

```
:CP-DIR
[CP.031] Direct connected to communications port P3.
:SET-BREAK OFF
Break disabled
:ESC-DATA
:XONOFF (F

[1315] xon/xoff is disabled.
:XCS-ON
:CP-RECV-XMODEM TEMP3 (C,N,M,2000
C
[CP.032] Direct connect exited from communications port P3.

:CP-SEND-XMODEM GL.BP GL.RPTS.DETAIL GL.SUMMARY GL.RPTS.1 (BB,C,E,N,2000
                    XModem-CRC-1K
      1    12,636              GL.RPTS.DETAIL
                        1      2,000   0.05   667
                        2      4,000   0.07  1000
                        3      6,000   0.08  1200
                        4      8,000   0.12  1143
                        5     10,000   0.13  1250
                        6     12,000   0.15  1333
      2       114              GL.SUMMARY
      3       606              GL.RPTS.1
                        7     14,000   0.18  1273
                    EOT
[CP.146] XModem send of GL.BP completed.
         3 items, 14000 bytes, 7 blocks, 0.22 minutes, 1077 chars/sec.
[CP.148] Result code: 0, Completed
         Protocol: XModem-CRC-1K
```

The CP-RECV-XMODEM and CP-SEND-XMODEM to move files between two Pick systems

For More Information

Refer to the command reference in the *File Transfer TCL Commands Reference* section of this document.

Using the CP-SEND-YMODEM/CP-RECV-YMODEM Commands

The following example sort selects 3 items in the "TRANS" file and sends them to the remote system using "CRC" packet check (if available). The "B" option display the block information indented on the right.

Example:

```
:CP-SEND-YMODEM TRANS 1 2 3 (B,C
                        1        128   0.02   128
      1     5,000 1
                        2      1,152   0.02  1152
                        3      2,176   0.03  1088
                        4      3,200   0.05  1067
                        5      4,224   0.07  1056
                        6      5,248   0.07  1312
                        7      5,376   0.10   896
      2     5,000 2
                        8      6,400   0.10  1067
                        9      7,424   0.12  1061
                       10      8,448   0.13  1056
                       11      9,472   0.15  1052
                       12     10,496   0.17  1050
                       13     10,624   0.18   966
      3     5,000 3
                       14     11,648   0.18  1059
                       15     12,672   0.20  1056
                       16     13,696   0.22  1054
                       17     14,720   0.23  1051
                       18     15,744   0.23  1125
                       19     15,872   0.27   992
[CP.149] YModem send of TRANS completed.
         3 items, 15872 bytes, 19 blocks, 0.27 minutes, 992 chars/sec.
[CP.148] Result code: 0, Completed
         Protocol: YModem-CRC
```

Using CP-SEND-YMODEM to send multiple items as files to a remote system

The following example demonstrates using CP-RECV-YMODEM to receive from ProComm host mode to Pick. The "^^" options clears the screen and uses the screen display mode to show file transfer progress. The "P" option with ",,,15" overrides the maximum consecutive errors to 15. The "S" option splits received data at 20000 bytes.

Example:

```
F)iles      U)pload      D)ownload
H)elp       T)ime        C)hat
R)ead mail  L)eave mail  G)oodbye

Your choice? D

A) Ascii    Y) Ymodem (Batch)    S) Sealink
K) Kermit   O) 1K-XMODEM         T) Telink
X) Xmodem   E) 1K-XMODEM-G       W) Wxmodem
Z) Zmodem   G) Ymodem-G (Batch)  I) Imodem

Your choice? Y

File name? stac.log
[CP.032] Direct connect exited from communications port P15.

:CP-RECV-YMODEM TEMP3 (C,P,S,^^) {,,,15} {20000}

     +--------------------------------------+
     | Mode.......... Receiving             |
     | Protocol...... YModem-CRC            |
     | Block size....      1,024            |
     | Time started.. 08:22:00pm            |
     | Time now...... 08:23:29pm 00:01:29   |
     | Item count....          3            |
     | Total bytes...     61,440            |
     | Total blocks..         61            |
     | Corrections...                       |
     | Average speed.        690 cps        |
     | Pick file..... temp3                 |
     | Pick ID....... STAC.LOG*2            |
     | File name..... STAC.LOG              |
     | File size.....    481,758            |
     | Transferred...     61,440  12%       |
     | Message.......                       |
     +--------------------------------------+
```

Using CP-RECV-YMODEM with ProComm host mode

Sending AP to R83 to AP example

It is possible to send data from Advanced Pick to R83 and back to Advanced Pick even though the items being sent from the originating Advanced Pick system are larger than the R83 system can store. For the following example, R83 is the local system:

Example:

On the first (Advanced Pick) system:

```
CP-SEND-YMODEM BP * (C,N,O) {BIG} {99999999}
```

(exit direct mode)

```
CP-RECV-YMODEM BP (C,S) {5000}
```

This sequence of commands sends the entire BP file as one ("O" option) long YModem file with embedded item-ID's ("N" option) to the R83 system which splits the items ("S" option) into 5000 byte records.

Then, connect to the second (Advanced Pick) system:

```
CP-RECV-YMODEM BP (C,N
```

(exit direct mode)

```
CP-SEND-YMODEM BP *AR (C,O} {BIG} {99999999}
```

These commands send the BP file in ascending-right-justified order as one ("O" option) long YModem file to the second Advanced Pick system. Since the "N" option is specified on the receiving side, the filename "big" and item size "9999999" is ignored and the item-ID's are extracted from the long file originally placed there by the "N" option on the first Advanced Pick system's CP-SEND-YMODEM command.

Sending Pick to DOS to Pick example

The same can be done as the prior example, (using XModem or YModem) using a DOS system. The DOS system would replace the R83 system in the prior example. Of course the receive and the sending would be done using the DOS (or Windows) communications software.

For More Information

Refer to the command reference in the *File Transfer TCL Commands Reference* section of this document.

Getting Help from TCL

You can get on-line help by typing in the command and using the "?" option in parenthesis.

Example:

```
:CP-ATT (?
Help for 'CP-ATT':

CP-ATT

     Attach communications port specified.

     CP-ATT portname (options)

     portname       Name of communications port previously started
                    with CP-START command

     options        Options may be any of the following:

     C              Check only.

                    This option does not change the current
                    attachment.  Portname is ignored with this option.

     D              Enter direct mode after attach.

                    This option will invoke the CP-DIR command after
                    successful attachment.
```

Using "?" to get help for a command

For More Information

See the command explanation of *CP-HELP*, starting on page 104 for more options.

Note: Longer help screens will automatically paginate at the screen depth associated with the TERM command. To avoid pagination, use "(?>)" instead of "(?)".

Where to Go from Here...

To use the other TCL commands available, refer to the chapter, *TCL Commands Reference,* starting on page 94.

CHAPTER 5
Technical Information

This chapter provides more detailed information about PowerComm TCL commands. It also contains information about communications ports and more tips for using PowerComm.

TCL Commands - General Notes

Turning On Extra Messages ($)

Many PowerComm TCL commands can generate extra messages during the execution of the command. These extra messages or functions are turned on by using the "$" option. Refer to the *TCL Command Message Control Table* starting on page 152 for more information about what messages and functions can be controlled with this option.

You may also use "!" option with commands to show additional messages. This unsupported option is cumulative and may be specified from 1 to 5 times depending on the level of debug messages you want to see. These are for us to help with debugging and monitoring. They are not documented, nor supported. Use at your own risk.

Turning Off Messages (#)

Like the extra messages option, many PowerComm TCL commands generate messages during the execution of the command. These messages can be suppressed by using the "#" option. Refer to the *TCL Command Message Control Table* starting on page 152 for more information about what messages can be suppressed with this option.

Turning Off Half Intensity (@)

Occasionally you may want to suppress the half intensity variables from being assigned for a command. The half intensity feature may be suppressed by using the "@" option. You may want to do this when you are using the CP-VIEW-TCL command with a PowerComm command such as CP-LIST or CP-STATUS-PORT that uses the half intensity for static information and regular intensity to help highlight the dynamic data. If you did not suppress this when using the CP-VIEW-TCL command, extra characters will be captured on the screen which would mess up the normal display of the CP-VIEW-TCL command.

You may also want to use this option if you are executing a PowerComm TCL command in BASIC and you are capturing the output for subsequent parsing.

TCL Commands - Notes by Command

CP-CAPTURE

During direct mode with capture, the captured data is stored in a buffer until the exit or break key is pressed. Depending upon how long you have been capturing data, there may be a slight delay after using the exit or break key while the captured data is written to the data file.

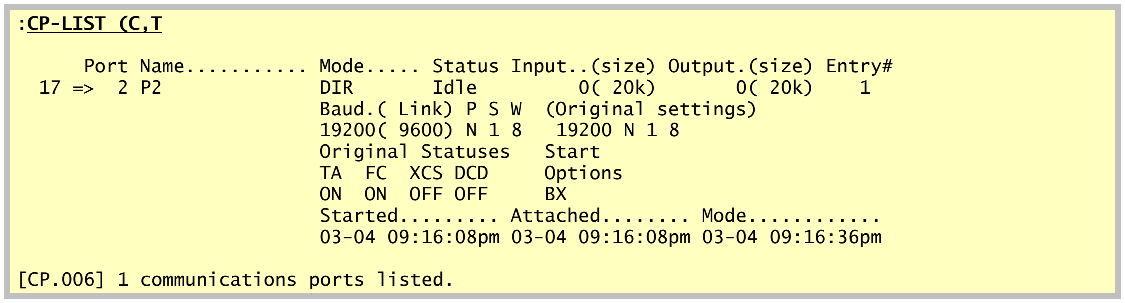
Item sizes of capture items are about the frame size of your system. This is so the items that are created roughly fill up an entire frame. This is nice on AP systems because the item will become indirect and most of the frame will be filled with data.

CP-LIST

Column Headings Explained

The following example using CP-LIST with the "C" and "T" options shows all of the column headings available.

Example:



```
:CP-LIST (C,T

      Port Name........... Mode..... Status Input..(size) Output.(size) Entry#
  17 =>  2 P2              DIR       Idle         0( 20k)       0( 20k)    1
                           Baud.( Link) P S W  (Original settings)
                           19200( 9600) N 1 8   19200 N 1 8
                           Original Statuses   Start
                           TA  FC  XCS DCD     Options
                           ON  ON  OFF OFF     BX
                           Started......... Attached........ Mode............
                           03-04 09:16:08pm 03-04 09:16:08pm 03-04 09:16:36pm

[CP.006] 1 communications ports listed.
```

These are the standard columns that are displayed

Column — **Meaning**

Port — This is the communications port.

If a port has the communications port attached, it is shown to the left of the communications port with either a single arrow "->" or a double arrow "=>". The double arrow represents the current port and the single arrow shows up only when a port has more than one communications port attached. The "*" to the left of the port number is shown to indicate it is your own port number.

Name — This is the name that was used with the CP-START command.

Mode — This is the current program mode the communications port is in. The following documents the modes that may be present in this column:

Mode	Meaning
ABORT	Abort mode from using the CP-ABORT command.
ASCII	ASCII send or receive mode from using the CP-SEND-ASCII, CP-RECV-ASCII, CP-SEND-PICK, CP-RECV-PICK commands.
CONNECT	Connect mode from using the CP-CONNECT command.
DIR	Direct mode from using the CP-DIR or CP-CAPTURE commands
HOST	Host mode from using the CP-HOST command.
MSG	Message mode from using the CP-MSG command.
RECV	Receive mode from using the CP-RECV command.
REMOTE	Remote mode from using the CP-REMOTE command.
SCRIPT	Script mode from using the CP-SCRIPT command.
SEND	Send mode from using the CP-SEND command.

Status — This is the current status of the communications port. This field varies according to current communications port activity. The following documents the values that may appear in this column:

Status	Meaning
Sleep	Comm port is in a sleep mode, from a CP-PAUSE command.
Active	Comm port is actively inputting or outputting data.
?	Comm port status is unknown but probably very actively receiving or transmitting data.
Idle	Comm port is running but no current input or output is occurring.
Ambig.	Comm port is not running and was probably killed. There is still an entry in the comm port table probably from an unsuccessful kill command.

Input (size) — This is the current number of bytes in the input buffer and the size of the input buffer in K bytes.

Output (size)	This is the current number of bytes in the output buffer and the size of the output buffer in K bytes.
Entry#	This is the entry number in the communications port table.

These additional columns are displayed when the "C" option is used

Column	Meaning
Baud (Link)	This is the communications port baud rate that was used when it was started. If present, the link rate in parenthesis controls timeout computation.
P S W	This is the communications port parity, stop bits and word length that was used when it was started.
(Original Settings)	This is the original port's baud rate, parity, stop bits and word length before the communications port was started. These are the parameters that will be used to restore the port to its original settings.
Original Statuses	This is the original port's other mode settings including: type-ahead mode, flow control mode, extended character set mode, and data carrier detect mode.
TA	"ON" if the type-ahead feature was originally on, otherwise "OFF".
FC	"ON" if the XON/XOFF flow control feature was originally on, otherwise "OFF".
XCS	"ON" if the extended character set feature was originally on, otherwise "OFF".
DCD	"ON" if the data carrier detect feature was originally on, otherwise "OFF".
Start Options	May contain "X" if the "X" option was used with CP-START, and "B" if the "B" option was used.

These additional columns are displayed when the "T" option is used

Column	Meaning
Started	This is the date and time the communications port was started using the CP-START command.
Attached	This is the date and time the communications port was last attached using the CP-ATT command.
Mode	This is the date and time the port that has the communications port attached entered the mode indicated.

Note: The baud rate displayed with CP-LIST may not be the current baud rate if it was changed from TCL after the port was started using any of the Pick commands to change the baud rate.

CP-PACKAGE

The CP-PACKAGE program works by creating a series of items in the package file specified suffixing the package name with an asterisk and a sequential number. The items created are targeted for 5,000 bytes. You may change the default size using a command line option. As each package item grows and exceeds this value the package program increments the sequential counter and creates another item.

If the size of the item about to be added to the package item is larger than the target size, an external item is created with a pound sign "#" suffix and a sequential number. This eliminates concatenation workspace problems with R83 systems.

The last index number used along with the total number of bytes is stored in a control item in which the ID is the package name followed by an asterisk. (E.g. pkgname*). The following are the attributes:

```
    pkgname*
001 version
002 reserved
003 last index number
004 last index size
005 number of entries (number of items or TCL commands)
006 total number of bytes
007 number of user commands
008 last external index number
```

An example control item looks like:

```
    U110894*
001 3.00
002
003 1
004 837
005 7
006 13473
007 1
008 1
```

An example data item looks like:

```
    U110894*1
001
002 _TCLA SSELECT BP = "BM.]"]0
003
004 _DATA BP BM.BIG]12636]1
005
006 _C  BM.CALL]63
007 P1 = "ORIGINAL P1"
008 CALL BM.SUB(P1, "THIS IS ANOTHER P1")
009 CRT P1
010 _C  BM.GOSUB]127
011 T = SYSTEM(12)
012 SUB.NAME = "BM.SUB"
013 FOR I = 1 TO 1000
014   CALL@ SUB.NAME
015 NEXT I
016 CRT 1000/((SYSTEM(12) - T) / 1000)
017 STOP
018 100 RETURN
019 _C  BM.LOCATE]487
020 *
021 FMT = "AL"
```

Other Information about Package

- Invalid commands may not be detected in the package control item. This is because some commands are applicable during add mode, and others are applicable during extract mode. Others are applicable during both modes.
- To view individual commands in the package during add or extract modes, you may use the "$" option.
- Bytes displayed on the screen may be different in add mode vs extract mode because of the byte values they represent.

CP-SEND, CP-RECV

Precautions about Sending and Receiving Files

If you store data in dictionary level files that look like "D" pointers, you may confuse Pick into thinking they are file defining items. Especially on R83 where there are no extra levels of protection against this.

PowerComm employs a certain level of checking concerning writing or overwriting of "D" pointers to dictionary level files. However, this checking is only invoked if you use the word "DICT" as part of the received file name. If you omit the "DICT" option, PowerComm will not perform this check.

Therefore, if you wish to send the MD, or the SYSTEM file and you would like PowerComm to filter out "D" pointers, specify "DICT MD" or "DICT SYSTEM".

You may, at your own peril, override the checking option with DICT files by using the "O" option.

The following table illustrates the checking of "D" pointers (aka File Defining Items), and "CC", "CL" pointers (aka BASIC Object Code Pointers, Lists). with and without the "O" option.

File Name	Normal	"O" option
DICT *file*	checks	no check
file	no check	no check
Examples:		
DICT CUSTOMER	checks	no check
DICT SYSTEM	checks	no check
DICT MDS	checks	no check
DICT MD	checks	no check
CUSTOMER	no check	no check
SYSTEM	no check	no check
MDS	no check	no check
MD	no check	no check

Caution: On R83, the receive program is not smart enough to know the file is a "DC" file pointer and check only the "DC" files when checking for "CC" or "CL" items. In other words, all dictionary files are checked when using "DICT" unless the "O" option is used. In addition, if you have "CL" items in a data level file, and the receive file is a "DC" data level file, you may inadvertently create file corruption problems on the target system.

Notes about Characters Displayed During Communications

During data communications transfers you may see the following groups of characters on the screen. Following is a brief description of what they represent.

This Character	Means
<S>	Synchronization attempt. This is displayed on the screen when the software is trying to synchronize with the other end. It is normal to see a few of these at the beginning of a transmission.
<B>	Buffer overrun detected on the remote system. This shows up when you have overloaded the other computers input buffer causing Pick to object with its bell character. You can either; reduce your block size, slow the baud rate, or upload when there is less disk activity on the other system. It can also be caused by Pick's 10 second flush feature.
<R>	Re-send requested by remote system. This is displayed when the communications software on the other end has detected an incomplete or incorrect data block. It is normal to see one or two at the beginning of a transmission during syncronization.
%	Time-out detected by local system. This is displayed when a period of time has elapsed with no response from the remote system. It could mean that the other system is too slow to process the data, lost part of the packet, or the modem has hung up the connection. It is not normal to see this character unless there are problems somewhere. If you suspect the other end is quite busy, you might try changing the time-out value in the *Port Definition* screen for this port.

Showing Extra File Transmission Statistics

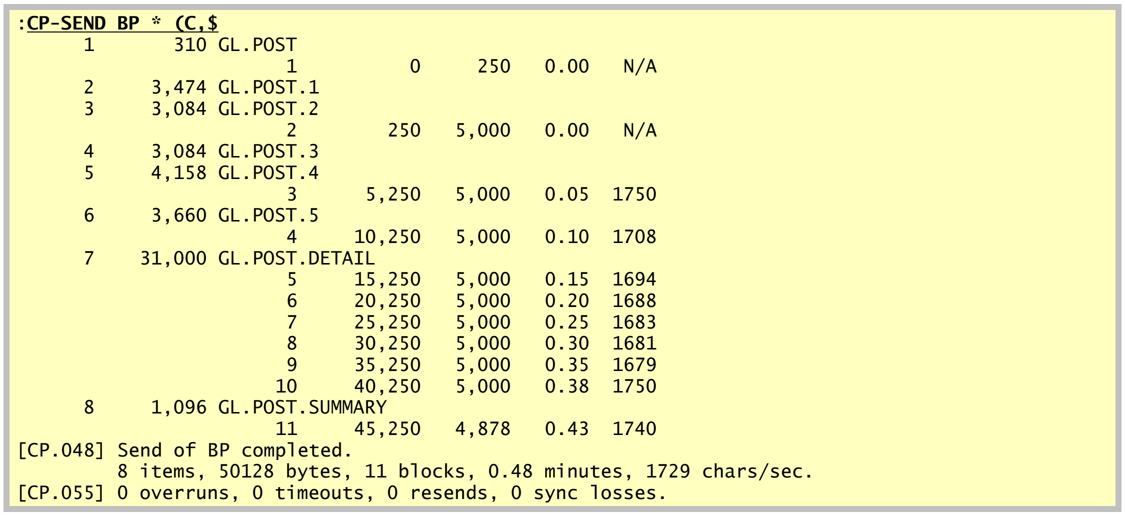
The "$" option when used with the CP-SEND and CP-RECV commands will show how many occurrences of retries, timeouts, buffer overruns and synchronizations occurred during the file transfers.

Showing Communications Blocks

You can use the "C" option with the CP-SEND and CP-RECV command to view the communications blocks as they are sent. The following examples are shown to explain each column displayed.

This first example is using the CP-SEND command. The second shows the display when using the CP-RECV command.

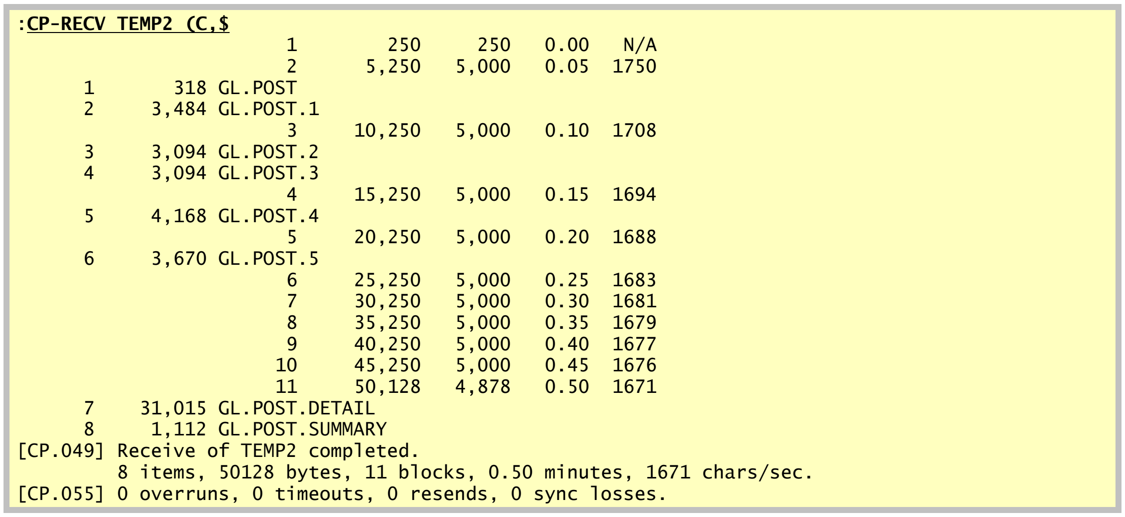
Example 1:



```
:CP-SEND BP * (C,$
      1       310 GL.POST
                        1          0     250   0.00   N/A
      2     3,474 GL.POST.1
      3     3,084 GL.POST.2
                        2        250   5,000   0.00   N/A
      4     3,084 GL.POST.3
      5     4,158 GL.POST.4
                        3      5,250   5,000   0.05  1750
      6     3,660 GL.POST.5
                        4     10,250   5,000   0.10  1708
      7    31,000 GL.POST.DETAIL
                        5     15,250   5,000   0.15  1694
                        6     20,250   5,000   0.20  1688
                        7     25,250   5,000   0.25  1683
                        8     30,250   5,000   0.30  1681
                        9     35,250   5,000   0.35  1679
                       10     40,250   5,000   0.38  1750
      8     1,096 GL.POST.SUMMARY
                       11     45,250   4,878   0.43  1740
[CP.048] Send of BP completed.
         8 items, 50128 bytes, 11 blocks, 0.48 minutes, 1729 chars/sec.
[CP.055] 0 overruns, 0 timeouts, 0 resends, 0 sync losses.
```

Showing more information using the "C" and "$" options with the CP-SEND command

Example 2:



```
:CP-RECV TEMP2 (C,$
                        1        250     250   0.00   N/A
                        2      5,250   5,000   0.05  1750
      1       318 GL.POST
      2     3,484 GL.POST.1
                        3     10,250   5,000   0.10  1708
      3     3,094 GL.POST.2
      4     3,094 GL.POST.3
                        4     15,250   5,000   0.15  1694
      5     4,168 GL.POST.4
                        5     20,250   5,000   0.20  1688
      6     3,670 GL.POST.5
                        6     25,250   5,000   0.25  1683
                        7     30,250   5,000   0.30  1681
                        8     35,250   5,000   0.35  1679
                        9     40,250   5,000   0.40  1677
                       10     45,250   5,000   0.45  1676
                       11     50,128   4,878   0.50  1671
      7    31,015 GL.POST.DETAIL
      8     1,112 GL.POST.SUMMARY
[CP.049] Receive of TEMP2 completed.
         8 items, 50128 bytes, 11 blocks, 0.50 minutes, 1671 chars/sec.
[CP.055] 0 overruns, 0 timeouts, 0 resends, 0 sync losses.
```

Showing more information using the "C" and "$" options with the CP-RECV command

The first set of numbers is shown unless the "I" option is used. These are the column descriptions:

Column#	Meaning
1	Item number
2	Item size
3	Item ID

The second set of numbers shown are the communications blocks information and are indented to differentiate them from the item information. These are the column descriptions:

Column#	Meaning
1	Block#
2	Total bytes sent or received so far
3	Current send or receive block size
4	Total time in minutes so far
5	Average characters per second so far

About Timeouts

When using CP-SEND and CP-RECV, the timeout value is computed by using the baud rate of the connection when it was started with the CP-START command. If you specify a throughput different from the baud rate, the throughput will be used to compute the timeout value instead.

If you subsequently change the baud rate of the communications port without using the CP-START command, PowerComm will not know of this change and it may use a timeout value that is inappropriate. Instead, kill and restart the port at the baud rates that you want.

The total timeout value is based the current packet size, the throughput associated with the CP-START command, a 75% multiplier factor, and the timeout value from the *Port Definition* screen.

The timeout formula is:

Total timeout = packet.size / (throughput / 10) * 75% + timeout.value

Example:

Total timeout = 5000 / (9600 / 10) * .75 + 5 = ~ 12 seconds

This example shows a block (packet) size of 5000, a baud rate of 9600 and a timeout value in the *Port Definition* screen of 5 seconds. This example shows it will be approximately 12 seconds before PowerComm sends a synchronization packet.

This timeout value is computed dynamically with any packet size changes as a result of retries.

Send and Receive Protocol Information

PowerComm automatically masks all characters less than ASCII 32 from being sent to the remote computer to eliminate lock ups with these control characters.

In addition when using the "B" option with CP-SEND and CP-RECV, all characters above ASCII 126 are also masked. The "B" option is automatically selected if the communications port was started with the "B" option, or if using "ME" mode, the *Start port options* field in the *Port Definition* screen contains a "B".

Note: When using "ME" mode, the computer you are using the "M" option on is the same computer that the *Start port options* field is accessed for the port number the "ME" mode is occurring on. In other words, the "ME" mode uses its own port number to read the port definition for *its* port on *its* computer. It does not reach out over the communications link and grab the options from the local computer's port definition file.

With any masking, certain files may take longer to transmit than others depending upon the amount of masked characters.

Proprietary Standard Protocol Features

PowerComm uses its own proprietary protocol for maximum speed and reliability. The following explains the features of this protocol.

- Remote computer overrun detection.

  This feature automatically detects when the remote computer buffer is overrun. This is detected when the bell character is received from the remote. When this occurs, the block size is adjusted downwards and the block is re-transmitted.

- User definable block size provides maximum data throughput under user control.

  The block size can be preset globally, by port, or at runtime with a command line option.

- Multiple items per block dramatically increases data throughput.

  Unlike other communications products, PowerComm sends multiple items per block. This is especially beneficial when sending items that are less than the block size because this reduces the amount of overhead and time delays associated with processing a block for each individual item.

- Automatic error detection and correction maintains data integrity.

  A block number, checksum and length is supplied with each packet ensuring each packet is received correctly by the remote. If the packet received is not the same, the packet is re-transmitted.

- Automatic block size leveling protocol increases data throughput for busy systems or degraded connections.

  This involves an algorithm that we have developed over many years of using PowerComm. From time to time, the remote computers activity can vary which can effect how well it can receive streaming data. If you were to continually transmit to a busy computer with a fixed block size, it would finally give up when the number of retries established was exceeded.

  To avoid the early termination of the transfer completely, we have developed our protocol, which is user-definable, to accommodate a remote computer that may have activity on an irregular basis. The idea is to send the packets at a certain size until a buffer overrun occurs. When this happens, we reduce the packet size and try again. When the packet gets through successfully, we slowly increase the packet size until the buffer overrun occurs again. At this point we back off the buffer size just a little to be just below the maximum data the remote can receive at one time without overrunning its buffer.

  At any given time, the packet size can automatically vary between the maximum specified and the minimum specified. The minimum is calculated based on the maximum times the *Overrun buffer min. size* field. The current block size can be viewed with the CP-SEND and CP-RECV commands using the "C" option.

  If you do not want the protocol to change its buffer sizes like this, you can change all of the values to "1.00", except for the minimum size which should be set to "0.00".

  The following paragraphs and tables illustrate this concept in more detail.

Understanding the Automatic Block Size Leveling Protocol

The fields in the following table are maintained in the *Port Definition* screen on the main menu. The example session following this table uses these parameters defined for the communications port:

Field name	Value	Used in block number
Default S/R packet size	5000	1,2,3
First block adjust down (%)	0.05	1
Overrun buffer adjust down (%)	0.50	4,5
Overrun buffer adjust up (%)	1.15	6,7
Overrun buffer stabilize (%)	0.85	8
Overrun buffer min size (%)	0.10	(not shown)

The overrun buffer minimum size, not shown in this example, is used to terminate the transfer should the block size fall below this value. In this case, the transfer would abort if the block size was ever adjusted below 500 bytes.

In the following example, the parenthesis in the 'data' column shows the resulting packet size after the formula shown is applied. The 'max.block' is 5000 bytes. The 'last.block' is the size of the last block sent. The block number is the sequential block number sent to the remote computer. The packets are automatically reassembled and resent at a new size if needed.

This example shows what happens during a file transfer in which the receiving computer input buffer is overrun.

Example:

Block	Sending		Receiving
No.	Computer	Data	Computer
1	packet 1 (max.block * [first block adjust down])		
		first packet sent (250) →	received ok
		← ok	
2	packet 2 (max.block)		
		packet sent (5000) →	received ok
		← ok	
3	packet 3 (max.block)		
		packet sent (5000) →	buffer overrun
		← bell	
		synchronize →	incomplete packet
		← re-send	
4	packet 3 (last.block * [overrun buffer adjust down])		
		packet resent (2500) →	buffer overrun
		← bell	
		synchronize →	incomplete packet
		← re-send	
5	packet 3 (last.block * [overrun buffer adjust down])		
		packet resent (1250) →	received ok
		← ok	
6	packet 4 (last.block * [overrun buffer adjust up])		
		packet sent (1438) →	received ok
		← ok	
7	packet 5 (last.block * [overrun buffer adjust up])		
		packet sent (1654) →	buffer overrun
		← bell	
		synchronize →	incomplete packet
		← re-send	
8	packet 5 (last.block * [overrun buffer stabilized])		
		packet resent (1405) →	received ok
		← ok	
9	packet 6 (last.block)		
		packet sent (1405) →	received ok
		← ok	

Block	Packet	
No.	**No.**	**Explanation**
1	1	The first packet is sent at a reduced size and received successfully.
2	2	The next packet is sent and also received ok.
3	3	The next packet is sent and the receiving computer is perhaps busy at the time and the bell character is echoed to the sender. This causes a synchronize to occur, then the receiving computer responds with a re-send because of an incomplete packet.
4	3	The next block is the prior packet halved and resent. It also overruns the buffer of the remote computer.
5	3	The next block is the prior packet halved and resent. It is then received ok.
6	4	This packet is adjusted up because the last packet is received ok. This occurs to increase the buffer size until the buffer overrun occurs again.
7	5	This packet is also adjusted up but the remote's buffer was overrun.
8	5	Because of the overrun, the size of packet 5 is resent after backing off the buffer size a little.
9	6	At this point, given the load on the remote system, the optimum buffer size has been established and the packets continue.

Packet Structure

The following packet structure table illustrates the relationship of the protocol characters.

packet start character	checksum information	data start character	packet data	data end character	packet end character	packet termination character

When using the CP-SEND, and CP-RECV commands, PowerComm masks all control characters from the data before transmission. This is to eliminate data from locking up modems and/or ports that may be sensitive to control characters. PowerComm uses a few control characters as part of its packet structure. These characters are defined in the *Port Definition* screen starting on page 129 and must be unique.

In some circumstances, you may need to change these characters to suit a particular configuration issue. For example, on AP/Unix, you can re-map the escape and break keys to a different character. If this character is also used by PowerComm you may encounter an unexpected level push or break condition on the remote port. If you make changes, you also have to change each system that PowerComm may be connected with. In other words, the protocol characters have to match on both ends of the communications link or it will not function properly.

CP-START, CP-KILL

About the Throughput Start Port Parameter

This is used in situations where the transmission rate is running at a slower rate than the DTE rate to the modem. This parameter effects the computation of wait time for acknowledgment from the remote system. If you have a fixed DTE rate of 19200 but a connect rate of 2400, the timeout value would be insufficient for the actual amount of time it should take for data to arrive at the remote end.

To handle this situation, the throughput parameter can be used to indicate a different throughput rate from the actual baud rate. It is then used to calculate timeouts for sending and receiving of files.

See the section, *About Timeouts*, starting on page 70 for more information.

Using the Before Start and After Kill Port Program Parameters

We have provided a hook that allows you to run a subroutine before a communications port is started with optional parameters from the *Port Definition* screen. Another subroutine can be called after the port is killed.

The following fields are used from the *Port Definition* screen with this feature:

Before start port program
Before start port parameters
After kill port program
After kill port parameters

If you use this feature, your program subroutine uses the following parameter format on the first line as shown:

SUB subname(Option, Parameters, CommPortNo, bError)

The 'subname' is the name of the program from either the *Before start port program* or the *After start port program* field.

The following describes the variables passed or returned.

Parameters Passed To Your Subroutine

Option Option may be one of the following:

S If called from the CP-START command.

K If called from the CP-KILL command or an unsuccessful CP-START command.

Parameters Set to either the *Before start port parameters* or the *After start port parameters* field defined in the *Port Definition* screen.

CommPortNo Set to the communications port being started or killed.

Parameters Returned From Your Subroutine

bError You may set this to "1" to indicate an error. If this is set to "1" the CP-START command will not start the port. When "option" is set to "K" there is no interpretation of this return parameter.

About the CP.AP.UNIX.UTY.SUB Example

We have provided an example subroutine located in the LSCP,BP.EXAMPLES file which is called "CP.AP.UNIX.UTY.SUB". Its purpose is to start the Unix Pick (AP) process and to kill it when the communications port is killed. You may wish to use this program as a template to make more modifications. For example, you might want to modify this program to disable the Unix TTY before starting Pick. And when the communications port is killed, and the Pick process is killed, you might want to re-enable the Unix TTY.

The following fields from the *Port Definition* screen can be filled in as follows to use this subroutine:

Field Name	Field Value
Before start port program	CP.AP.UNIX.UTY.SUB
Before start port parameters	tty1a VIEW
After kill port program	CP.AP.UNIX.UTY.SUB
After kill port parameters	tty1a VIEW

You can replace the 'tty1a' with the Unix TTY you want to use on your system. The "VIEW" option is used in the program to display the TCL commands as they are executed.

CP-STAT-FILE

Checksums Between AP and R83

R83 does not support larger numbers than around 11 billion. Consequently, when adding up the checksum for a 16 or 32 bit value, you can quickly exceed this number and cause an overflow to occur. If you are sending files from AP to R83 and vice versa, you may want to check the file using a smaller checksum value of 8 or 16 bits. This is specified in the options for the CP-STAT-FILE command.

CP-STATUS-PORT

Using CP-STATUS-PORT to view port statuses can be very useful in troubleshooting problems.

The following example continuously displays the results from CP-STATUS-PORT commands on the screen every 5 seconds.

Example:

```
:CP-VIEW-TCL "CP-STATUS-PORT 2 15 (@,C,L" (5
Port: 2
Where Results
 002 000722 AF00 00EFC5      LSCP.AP.START:154 br.conv:235
PIB Status: AF00
A 1010  Active         Sleeping                      Echo Off
F 1111
0 0000  DCD Off                       TA On          XON/XOFF Off
0 0000
Port: 15
Where Results
 015 00072F AF00 00EFC5      LSCP.AP.START:154 br.conv:235
PIB Status: AF00
A 1010  Active         Sleeping                      Echo Off
F 1111
0 0000  DCD Off                       TA On          XON/XOFF Off
0 0000
```

Understanding the CP-STATUS-PORT Output

The following describes the information from the following CP-STATUS-PORT example:

Example:

```
:CP-STATUS-PORT 2
Port: 2
Current Port Communications Parameters
  Baud  Parity  Stop bits  Word length  TA  FC  XCS DCD
 19200  NONE        1          8        ON  OFF ON  OFF
Where Results
 002 000722 AF00 00EFC5      LSCP.AP.START:154 br.conv:235
PIB Status: AF00
A 1010  Active         Sleeping                      Echo Off
F 1111
0 0000  DCD Off                       TA On          XON/XOFF Off
0 0000
CP-LIST Results
      Port Name........... Mode..... Status Input..(size) Output.(size) Entry#
  16 =>  2 P2              DIR       Idle          0( 20k)       0( 20k)    0
```

Region 1 - Current Port Communications Parameters

The first region, the *Current Communications Port Area*, shows the current communications parameters reported by various Pick commands.

```
Current Port Communications Parameters
  Baud  Parity  Stop bits  Word length  TA  FC  XCS DCD
 19200  NONE        1          8        ON  OFF ON  OFF
```

Region 2 - Where Results

The *Where Results* section shows the output from the TCL command, "WHERE".

```
Where Results
 002 000722 AF00 00EFC5      LSCP.AP.START:154 br.conv:235
```

Region 3 - PIB Status

The *PIB Status* interpretation region takes the PIB Status from the WHERE command results and breaks out the bytes into the corresponding bits 4 bits at a time.

```
PIB Status: AF00
A 1010  Active         Sleeping                      Echo Off
F 1111
0 0000  DCD Off                       TA On          XON/XOFF Off
0 0000
```

Region 4 - CP-LIST Results

The 4th region, shows the output from the CP-LIST command. Refer to page 65, for more information about the CP-LIST command output.

```
CP-LIST Results
      Port Name........... Mode..... Status Input..(size) Output.(size) Entry#
  16 =>  2 P2              DIR       Idle         0( 20k)       0( 20k)    0
```

PIB Status Meanings on Advanced Pick

The following is a break down of the PIB status bits, their possible values and meanings:

Bit#	"1"	"0"	Description
1	Active		Process is running or may be activated
2		Sleeping	Process is sleeping
3		Frame Fault	Process waiting for a disk frame
4	Echo On	Echo Off	Echo is on or off
5		I/O Blocked	Port is waiting to output, usually XOFF'd
6		Input	Port is at input
7		Output	Port is outputting
8		Comatose	Port is in a 'comatose' state
9	DCD On	DCD Off	Data carrier detect is on or off
10	XOFF'd		Port is in an XOFF state
11	TA Off	TA On	Type ahead is on or off
12	XON/XOFF On	XON/XOFF Off	XON/XOFF flow control is enabled for this port
13			Undefined
14			Undefined
15			Undefined
16			Undefined

This table may vary between Pick platforms. Do not rely on this information until you have confirmed the relationship between what you see happening with a port and what this PIB status display shows.

Possible PIB Status Displayed Values on Advanced Pick

```
PIB Status: FFFF
F 1111  Active                                          Echo On
F 1111
F 1111  DCD On          XOFF'd          TA Off          XON/XOFF On
F 1111
```

Possible view if all bits one

```
PIB Status: 0000
0 0000                  Sleeping        Frame Fault     Echo Off
0 0000  I/O Blocked     Input           Output          Comatose
0 0000  DCD Off                         TA On           XON/XOFF Off
0 0000
```

Possible view if all bits zero

Caution: The accuracy of these values depends on the platform of Pick you are running. On AP/Unix, several of these statuses can be misleading. The "I/O Blocked" and "XOFF'd" fields don't always seem to be accurate. This may be because the Pick statuses do not always reflect the Unix statuses in real time.   On R83, these statuses seem to be pretty accurate.

PIB Status Meanings on R83

The following is a break down of the PIB status bits, their possible values and meanings:

Bit#	"1"	"0"	Description
1	Active		Process is running or may be activated
2		Sleeping	Process is sleeping
3		Frame Fault	Process waiting for a disk frame
4	Echo On	Echo Off	Echo is on or off
5		I/O Blocked	Port is waiting to output, usually XOFF'd
6		Output	Port is outputting
7		Input	Port is at input
8		Comatose	Port is in a 'comatose' state
9	Debugger		Port is in the debugger
10	XOFF'd		Port is in an XOFF state
11	TA On	TA Off	Type ahead is on or off
12	XON/XOFF On	XON/XOFF Off	XON/XOFF flow control is enabled or disabled for this port
13			Undefined
14			Undefined
15			Undefined
16			Undefined

Possible PIB Status Displayed Values on R83

```
PIB Status: FFFF
F 1111  Active                                  Echo On
F 1111
F 1111  Debugger       XOFF'd         TA On          XON/XOFF On
F 1111
```

Possible view if all bits one

```
PIB Status: 0000
0 0000                 Sleeping       Frame Fault    Echo Off
0 0000  I/O Blocked    Output         Input          Comatose
0 0000                                TA Off         XON/XOFF Off
0 0000
```

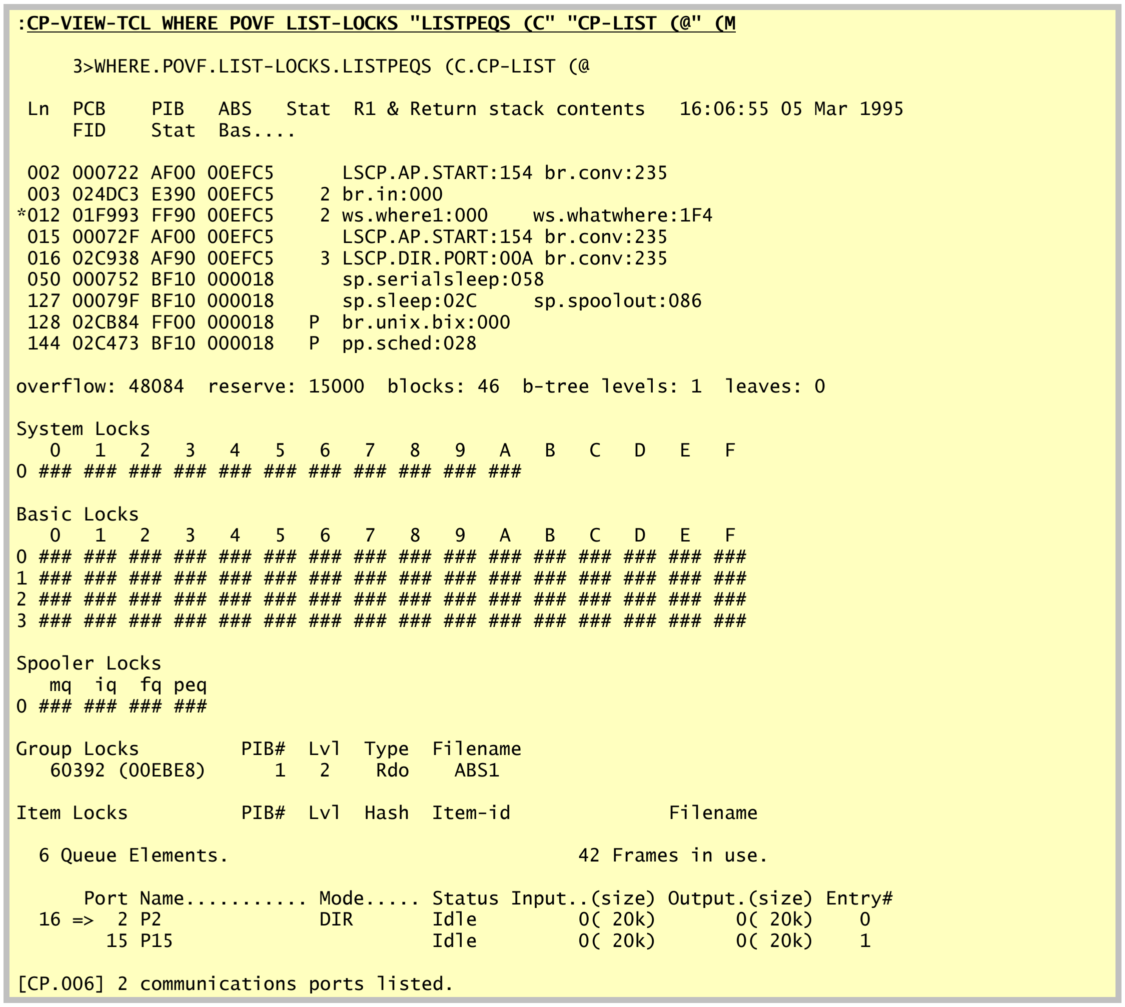
Possible view if all bits zero

CP-VIEW-TCL

If you have a terminal emulator that supports a lot of rows and you adjust your TERM page depth to use the extra rows, you can use the CP-VIEW-TCL command with a few key commands to find out what the system activity looks like.

On AP, the following example, using a large terminal depth parameter, shows a lot of information continuously updated several times a second.

Example:



```
:CP-VIEW-TCL WHERE POVF LIST-LOCKS "LISTPEQS (C" "CP-LIST (@" (M

     3>WHERE.POVF.LIST-LOCKS.LISTPEQS (C.CP-LIST (@

 Ln  PCB    PIB   ABS   Stat  R1 & Return stack contents   16:06:55 05 Mar 1995
     FID    Stat  Bas....

 002 000722 AF00 00EFC5      LSCP.AP.START:154 br.conv:235
 003 024DC3 E390 00EFC5    2 br.in:000
*012 01F993 FF90 00EFC5    2 ws.where1:000    ws.whatwhere:1F4
 015 00072F AF00 00EFC5      LSCP.AP.START:154 br.conv:235
 016 02C938 AF90 00EFC5    3 LSCP.DIR.PORT:00A br.conv:235
 050 000752 BF10 000018      sp.serialsleep:058
 127 00079F BF10 000018      sp.sleep:02C     sp.spoolout:086
 128 02CB84 FF00 000018   P  br.unix.bix:000
 144 02C473 BF10 000018   P  pp.sched:028

overflow: 48084  reserve: 15000  blocks: 46  b-tree levels: 1  leaves: 0

System Locks
   0   1   2   3   4   5   6   7   8   9   A   B   C   D   E   F
0 ### ### ### ### ### ### ### ### ### ### ###

Basic Locks
   0   1   2   3   4   5   6   7   8   9   A   B   C   D   E   F
0 ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ###
1 ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ###
2 ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ###
3 ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ###

Spooler Locks
   mq  iq  fq peq
0 ### ### ### ###

Group Locks         PIB#  Lvl  Type  Filename
   60392 (00EBE8)      1   2    Rdo    ABS1

Item Locks          PIB#  Lvl  Hash  Item-id             Filename

  6 Queue Elements.                               42 Frames in use.

      Port Name........... Mode..... Status Input..(size) Output.(size) Entry#
  16 =>  2 P2              DIR       Idle         0( 20k)       0( 20k)    0
        15 P15                       Idle         0( 20k)       0( 20k)    1

[CP.006] 2 communications ports listed.
```

Using CP-VIEW-TCL to show the world of activity

Caution: Running this command can be very CPU intensive.

Communications Notes

About Communications Ports

Communication ports on PowerComm use very little resources when idle. However, if you do not plan on using the communications port for an extended period of time, we recommend that you kill the port. Some versions of Pick, like R83, do not flush memory unless the system is completely idle. On AP, you can use the TCL "FLUSH" command to force an immediate memory flush to disk.

On R83, the communications ports do not show as being logged on unless the "L" option is used with the CP-START command. On AP, the communications ports show up with LISTU as "CP:COMM".

Me Mode Through a File

In addition to the "ME" mode of operation, some commands support the "F" option. This option when used with the "M" option, causes the communications to occur through a file on the local system and not on a communications port at all. This option may be used in testing and is supported with the following commands:

CP-ABORT
CP-RECV
CP-SEND

Note: You may not run more than two commands at the same time on the same system using the "M" and "F" options together.

The following example uses the send and receive commands with the "M" and "F" options to move data from one file to another on the same system.

Example:

```
CP-SEND BP1 * (M,F
```

On another port enter:

```
CP-RECV BP2 (M,F
```

Start Port Options Configuration

Binary 7 Bit Mode

Ideally, you want to use the full 8 data bits when moving files between systems. This way no additional masking of characters greater than 127 is required. This would include all system delimiters such as attribute marks, value marks and sub-value marks.

However, if you need to use 7 bits for data communications, you can automatically invoke the "B" option with the CP-START command by entering a "B" into the *Start port options* field in the *Port Definition* screen. When a port is started with the "B" option, it automatically selects the "B" option with the following commands:

CP-HOST
CP-RECV
CP-REMOTE
CP-SEND

This field also effects the "ME" mode of operation on the remote computer. If the remote computer contains a "B" in this field for the port you are using the "ME" mode on, it will automatically select the "B" option and use only 7 bits for sending or receiving the data.

XON/XOFF Mode

In most cases using XON/XOFF flow control only works on the transmitting side. That's because Pick does not support inbound flow control (except on AP/Unix).

This option would only be used if you had a high DTE rate but a slow connect rate. In this case it might be possible to overrun the modem's buffer without some sort of flow control. The "X" option is used to disable the turning off of XON/XOFF flow control that normally occurs when a communications port is started or "ME" mode is used.

Caution: Because the port is using XON/XOFF, it is possible for an outstanding XOFF from a modem to leave the port in an unusable state should this occur. On R83, there is now way to clear this condition short of pressing CTRL+Q on the port effected. On AP, you can use the RESET-PORT command to clear the XOFF'd port. (A good reason to upgrade to AP if there ever was one!)

The "X" option in this field is used by the CP-START command and the "ME" mode just as the "B" option documented above.

XModem and YModem Technical Information

Overview

The XModem protocol was created by Ward Christensen in 1977. It is capable of sending one file at a time with 128 data bytes per packet. The CRC version improves on the basic XModem checksum packet check by using a 2 byte CRC. The 1K version adds another data size packet type of 1024 characters.

The YModem protocol is essentially XModem CRC 1K with the ability to send 0 or more files at a time. This is accomplished by adding a zero block number header packet that contains the file name and file size.

Both XModem and YModem are receiver driven protocols. That is, the sender does not send a packet of data until the receiver requests it. This is why there is a long 60 second default timeout for both X and Y Modem sending commands. If no data is received within that period of time, the receiver has died or been disconnected. The sender does not try to initiate handshaking on its own.

Protocols Supported

The most common XModem and YModem protocols are listed below and are supported by PowerComm. In addition, PowerComm's implementation of the XModem and YModem protocols was done in such a way as to handle as many combinations of protocol variations possible. This is done by using option letters after the command. In addition, all of the timing parameters can be changed (permanently in the *Global Parameters Definition* screen) or temporarily at runtime by using the "P" option. With this flexibility, you should be able to create compatible file transfers with as many communications programs as possible.

This implementation has been tested with *ProComm for DOS*, *AccuTerm for Windows*, rb/rx and sb/sx on Unix and *Windows 95* built in communications. In our testing we found bugs in *AccuTerm for Windows* using their YModem receive with mixed 1024 and 128 byte packets, and timing irregularities with *Windows 95*. *ProComm for DOS* followed the protocol specifications the closest.

XModem

XModem is a single file protocol using 128 byte data blocks. It uses a checksum packet check method which is a simple 8 bit summation of all of the packet's data bytes. The file name cannot be transmitted therefore the receiving side must specify a destination for the transfer. In addition, because the file size is not specified in a XModem file transfer, it is possible for the file to grow up to 128 bytes (or 1024 with XModem-1K) with CTRL-Z padding in the last block.

XModem-CRC

The XModem-CRC protocol has better error detection by substituting the checksum method for a 16 bit CRC. Not all versions of XModem support the CRC check method. Negotiation for support of CRC is accomplished by the receiver sending "C"s at the beginning of the handshaking phase instead of the usual NAK every few seconds.

XModem-1K

This protocol adds another packet type that tells the receiver the a data block is 1024 bytes in length instead of 128 bytes. This version achieves much greater throughput because the data portion is eight times larger and the turn-around delays for acknowledging packets are eight times fewer. (This protocol typically implies CRC error detection and in PowerComm it is shown as XModem-CRC-1K).

XModem-1K-G

Like XModem-1K, but the receiver does not acknowledge each packet, only the final EOT. It should only be used with a reliable connection. In addition, the receiving program must be capable of keeping up with the continuous data stream. (This protocol typically implies CRC error detection and in PowerComm it is shown as XModem-CRC-1K-G).

YModem

YModem is like XModem-1K with CRC but it can also transmit the file name and file size to the remote system and it can send multiple files in one transfer session. YModem is also sometimes known as YModem Batch. With the file size, the receiver can truncate the final block at the proper position eliminating the file padding that occurs with XModem. (This protocol typically implies CRC error detection and in PowerComm it is shown as YModem-CRC).

Some versions of YModem are implemented without 16 bit CRC and use the checksum packet check method instead. In PowerComm, omit the "C" option when receiving from a remote system to speed up the initial handshake.

YModem-G

Like YModem but the receiver does not acknowledge each packet, only the final EOT packet for each file in the session. It should only be used with a reliable connection. In addition, the receiving program must be capable of keeping up with the continuous data stream.

Prerequisites for XModem and YModem File Transfers

You can use the XModem and YModem file transfer TCL commands with a communications port or in "Me Mode" on the local port. In either case, the port characteristics must be capable of sending the full 256 character set, from hex 00 to hex FF.

By default, the CP-START command sets up the communications port in this state. However, in "Me Mode" using the "M" option, you have to make sure the local port is set up correctly. See the section, XModem and YModem TCL Command Option Information, starting on page 81 for the commands when using the "M" option.

Debugging Protocol Problems

To help debug protocol problems with other packages (or PowerComm), you can use the CP-DIR command with the "MM" option to see what the other system is sending (and the delays) when the other system is in the YModem or XModem receive mode.

Getting Out of a XModem or YModem Transmission

Most XY/Modem protocols support the CTRL+X as the abort transfer character when sent several times (3 or 4). Depending upon what state the remote system is in, you may have to pause for up to 10 or 15 seconds, then press CTRL+X 3 or 4 times to terminate the transfer.

XModem and YModem TCL Command Option Information

Using the "A" (mask) option

When sending data to another system, you may need to send the segment mark character, CHAR(255) or hex "FF". In order to do this, you can mask the "FF" by putting a prefix character in front of its hex representation. For example, by using a CTRL-G (CHAR(7)) followed by "FF" (2 ASCII "F"s, 3 characters total) and specifying the "A" option with "07" this is easily done. Do not select the mask character CHAR(0) to avoid problems when using the "N" option. You are not allowed to select a mask character that can be mistaken as a hex character "0-9,A-F". When formatting your data for transmission, you must always remember to mask the mask character itself.

If you do not use the "A" option to mask received data when using the CP-RECV-XMODEM or CP-RECV-YMODEM commands, the hex "FF" character will automatically be converted to a hex "00" character.

Using the "C" (CRC 16 check) option

When the "C" option is used with the CP-SEND-XMODEM/CP-SEND-YMODEM commands, the sender will send the first data block after the sender receives the "C" handshake character. Without the "C" option, the sender does not send using the CRC protocol. Instead, sending will start when the sender receives the NAK character.

Using the "CC" option with CP-SEND-XMODEM/CP-SEND-YMODEM, the sender will only start sending if a "C" is received, and will fail to start if a NAK is received. This is how you force a CRC only transmission.

When the "C" option is used with the CP-RECV-XMODEM/CP-RECV-YMODEM commands, the receiver begins sending a "C" for the default 4 times, with a 4 second pause between each "C". If the first block is received within the "C" handshaking period, the receiver will expect CRC style formatted packets. If no data is received during the "C" handshaking, the receiver will switch to NAK handshaking for the next 6 attempts. When data is received during this period, the receiver will expect checksum style formatted packets.

Using the "CC" option, the receiver will only send 'C's to begin a transmission. This is how you force a CRC only transmission.

The total number of attempts is controlled by the maximum number of successive errors defined in the timing parameters.

Using the "D" (DOS filename) option

Since Pick item-ID's are not as restricted as DOS filenames, it is possible to abort a YModem receive on a DOS computer if you send an invalid filename in the file transfer. To avoid this, you can create DOS compatible filenames when you build your transmit data, or you can use the "D" option to create DOS filenames for you. The DOS filename created with the "D" option is based on the first valid characters of the original Pick item-ID followed by a "." (dot) and a 3 digit sequential number. This will assure uniqueness for the first 999 items you send regardless of the original Pick item-ID.

Using the "E" (escape) option

The "E" option can be used to provide a method of aborting a file transfer without using <BREAK> and "END". By using "E", the escape key can be used to abort a transfer. To specify another character, use "EE" and specify the character in hex to be the "escape" key.

Using the "G" (streaming) option

When the "G" option is used with the CP-RECV-XMODEM command, the receiver begins sending a "G" for the default 2 times, with a 4 second pause between each "G". If the first block is received within the "G" handshaking period, the receiver will expect streaming protocol. No error correction is available during the streaming protocol and should only be used in error-free situations. PowerComm continues to validate each packet using the checksum or CRC methods and will abort the sender if an invalid packet is received.

Without the "G" option, X and Y Modem send requires acknowledgment of each packet sent. By using the "G" option, only the final packet is acknowledged.

Using the "K" (1K XModem) option

When the "K" option is used, CP-RECV-XMODEM will understand the "STX" 1024 data byte block header packet format. When the "K" option is used, CP-SEND-XMODEM will send 1024 byte data blocks as well as 128 byte data blocks. The receiver must be running the XModem-1K protocol as well.

Using the "M" (me mode) option

The "M" option is used when sending or receiving to the current port the command is entered on. You must make sure that XON/XOFF flow control is disabled and a full 8 bit data path is available.

On R83:

`FC-OFF`	(disables XON/XOFF flow control)
`XCS-ON`	(enables 8 bit character set)

On Advanced Pick:

`XONOFF (F`	(disables XON/XOFF flow control)
`XCS-ON`	(enables 8 bit character set)
`SET-BREAK OFF`	(disables alternate break key mapping on Unix hosted systems)
`ESC-DATA`	(disables level pushing when escape (or assigned level-push key) is pressed)

Using the "O" (one item) option

The "O" option used with CP-SEND-YMODEM requires a file name and file size. To calculate the file size, you can simply total all of the bytes for all items. If you are including the item-ID (using the "N" option), you have to add the length of the item-ID, an attribute mark, and the item body trailing null character. If the source data includes masking characters, you must subtract 2 bytes for each mask character found.

Tip: You can use the CP-STAT-FILE command with the "A" and "D" options to compute the length of the selected items.

Using the "P" (timing parameters override) option

The "P" option can be used to override the default parameters from the *Global Parameters Definition* screen.

Using the "S" (split) option

The "S" option can be used when sending data to a system that cannot receive large (> 32K) items or when receive speed is an issue.

Using the "T" (translate) option

The "T" option allows translation between attribute marks and carriage returns. Additionally, using "TT" will translate attribute marks to carriage return/linefeed sequences when sending. When using CP-RECV-XMODEM/CP-RECV-YMODEM the "T" option converts carriage returns to attribute marks and line feeds are removed. When using the "T" option with large items, it is possible for the delays in translating these characters to cause timeouts by the receiving system. If this occurs, you may want to perform translation on the data before using the file transfer commands.

Using the "Z" option with YModem

You may want to send several Pick files as one YModem batch transfer. This may be done by using the "Z" option on all but the final CP-SEND-YMODEM command:

```
CP-SEND-YMODEM pick.file1 * (O,Z
CP-SEND-YMODEM pick.file2 * (O,Z
CP-SEND-YMODEM pick.file3 * (O
```

Using the "^" (screen display) option

The last Pick item ID shown on receive is the last ID filed. It does not reflect the current ID being received.

Using the "&" option to get message results

If you are controlling file transfers from a BASIC program, you may want to use the "&" option to save the error messages and results for subsequent use in your basic program. See page 99 for more details about this option.

Debugging Tips for Using XModem and YModem:

The following are some debugging tips for help in using the XY/Modem protocols.

➤ Use the "BBB" option to help debug protocol compatibility problems in non screen display mode.

By using this option, you can see which handshaking character is being sent or received.

- Use "CP-DIR (MM,1" to help debug protocol problems.

  This will eliminate screen lockup problems that can occur with some control characters and emulations while showing you the character's hex value in brackets, one character at a time.

XModem and YModem Result Codes Table

These are the result codes associated with error message CP.148.

Code	Description	Explanation
0	Completed	Normal completion of file transfer
1	Aborted - by operator	Transfer aborted by local operator pressing "escape" key
2	Aborted - timeout	Handshake timeout occurred
3	Aborted - maximum errors exceeded	Maximum number of consecutive errors has occured
4	Aborted - by remote system	Transfer aborted by remote system sending CTRL+X

PowerComm XModem and YModem Options Summary Table

The following tables shows the TCL command options used to obtain the PowerComm protocol names indicated.

CP-SEND-XMODEM, CP-RECV-XMODEM

PowerComm Protocol Name	Block Size	Checksum	CRC	Stream	Options
XModem-CHK	128	x			
XModem-CRC	128		x		C
XModem-CRC-1K	128/1024		x		C,K
XModem-CRC-1K-G	128/1024		x	X	C,G,K

CP-SEND-YMODEM, CP-RECV-YMODEM

PowerComm Protocol Name	Block Size	Checksum	CRC	Stream	Options
YModem-CHK	128/1024	x			
YModem-CRC	128/1024		x		C
YModem-CRC-G	128/1024		x	X	C,G

Note: More combinations are possible, but not all communications products support all combinations

Other Notes

How to Recover Global Parameters and the Global Port Definition

If you make changes to the global parameters item you do not want, or you want the original settings because the record was deleted, you can copy over the item from the LSCP,CONTROL file back to the LSCP.TBL,CONTROL file. The item name is "GLOBAL".

The global port definition item is also in the LSCP,CONTROL file and you may copy it back to the LSCP.TBL,CONTROL file as needed. The item name is "PORT*G".

Tips for Setting up Ports

You can check to see if a port works on AP before trying to get PowerComm to work with it by using the AP converse command.

Example:

```
CONVERSE 1
```

This will start a conversation with port 1. To terminate converse, press ESCAPE x. The "x" must be lowercase.

Verify System on Pick R83 Systems

On Pick R83, the TCL command "VERIFY-SYSTEM" utilizes the CHECK-SUM item in the ERRMSG file for comparing the ABS area frames against a checksum. These checksums are stored in that item. If during a verify system there are errors reported, it could be because a boot was done with the wrong diskettes, another product was loaded after the boot, or there really is a problem with the ABS. If you are not sure, the safest course of action is to reload the ABS using the ABS diskettes that correspond to your release. To find out which release you are running, type in the command "WHICH" from TCL:

```
>WHICH

PICK's R83 Ver 3.1 04 Jul 1990
```

You can restore this CHECK-SUM item from the original Pick data diskettes by using the SEL-RESTORE command with the "O" option:

```
>SEL-RESTORE ERRMSG CHECK-SUM (O)
ACCOUNT NAME ON TAPE: SYSPROG
FILE NAME: ERRMSG
```

Restoring the checksum item on R83

If you are absolutely sure your system should verify, you can rebuild the CHECK-SUM item by using the "O" option on the VERIFY-SYSTEM command as a last resort.

About Item Sizes

Pick removes a trailing attribute mark any time an item is saved to a file. This can throw off your byte counts when checking your uploaded data files against your source files. Also, R83 and AP compute item sizes differently and cannot be directly compared. You may want to use the CP-STAT-FILE command to show the size of the items along with a checksum to verify the data. If it is off by a byte, most likely it is because the original item contained a trailing attribute mark.

About PowerComm Naming Conventions

We recognize that we are not the only application program on your Pick system, and as such we have prefixed all of our subroutines with "CP", "LSCP" or "LSIU". All of the PowerComm TCL commands begin with "CP-". In addition, all of the software and data files are prefixed with "LSCP". This, hopefully, will prevent the collision of any subroutines that you may be using in your applications

The CP prefix stands for "communications program" and the LS prefix stands for "Logitek Systems", and "IU" stands for installation utility. Had we used "PC" for PowerComm instead of "CP" in all the commands we may have inadvertently confused users with PC's and PC oriented commands, i.e. PC-DIR.

CHAPTER 6
Troubleshooting

This chapter contains information helpful in finding the cause and solution to problems while using PowerComm.

Additional information is also contained in the chapter, *Technical Information*, starting on page 64.

Troubleshooting Techniques

The following are some techniques you can use when trying to track down problems:

- Use CP-STATUS-PORT to view detailed information about a port on the system. It may also be used for ports other than PowerComm communications ports. Refer to the topic, *CP-STATUS-PORT*, in the *Technical Information* chapter starting on page 75, for information about using and interpreting the results from this command.
- Use additional options with commands to see more information. This is especially useful when using CP-SEND and CP-RECV. Refer to the *TCL Command Message Control Table* on page 152.

General Problems

User Exit Abort

On AP, you might see the message:
[112] in program "CP.INIT", User exit 'LSCP.GET.PORT' is not valid.

Possible Causes	Possible Solutions
On AP, you have not EXEC'd the proper ABS file.	Make sure the EXEC command is used for each account logon that will be using PowerComm.

Item Size Mismatch Problems

You may notice an item size mismatch when comparing files between systems.

Possible Causes	Possible Solutions
R83 Pick truncates records at around 32K in length. If you send a file slightly larger than the R83 imposed limit, R83 will truncate the record possibly causing an item size mismatch.	Do not send items from AP to an R83 system that the R83 version of Pick cannot store.
If the final byte of an item is an attribute mark, Pick removes the trailing attribute mark when it is saved to the file causing a byte count mismatch of 1 byte.	Do not store trailing attribute marks in your data records, which may be unavoidable if a data value in an attribute several attributes past the end of the majority of the data has been removed.

Communication Problems

Port Lockup Problems

You may encounter a problem where the port appears to lock up.

Possible Causes	Possible Solutions
Cabling. A floating signal can cause intermittent port lock ups.	Correct the cable so that all of the pins required are connected to a signal.
The XOFF character has been received by the system.	▪ Press CTRL+Q. ▪ On AP, clearing a port lockup can usually be cleared with the RESET-PORT command. This essentially clears the XOFF condition. On Advanced Pick there is a command "RESET-PORT" that can be used to clear a port lockup problem. This command takes a port number as an argument: `RESET-PORT portno` Where "*portno*" is the port you wish to reset. ▪ There is no way to do this on R83 short of pressing CTRL-Q on the locked port. Sometimes the port on an R83 system gets so locked up that you must reset the system. This is a known O/S problem with Pick R83.
Modem feedback wars caused by the modem in command mode with echo, echoing back the Pick logon prompt to Pick and back to the modem and so on.	The port lockup problem can be avoided by making sure your modem is not set to echo result codes which is what typically causes the lockup to occur.

Garbled Characters in Direct Mode

You are in direct mode and garbage characters are on the screen when you type.

Possible Causes	Possible Solutions
Baud rate, parity, bits or word length may be wrong on port.	Use SET-PORT to change the port characteristics of the port, or kill and restart the port at the proper communications settings.
Bad connection when using a modem.	Manually reset modem or use the modem's escape sequence to hangup and try again.

No Response When Using A Modem

You have tried to type in direct mode and you have no response from the modem or the remote system.

Possible Causes	Possible Solutions
Modem result codes and echo has been turned off or reset.	Type "ATE1Q" and ENTER. The modem should say "OK" if you are in command mode. If you are on-line, you must enter the escape sequence, usually "+++" with a guard time on each side. The modem should respond with "OK" or in addition, you may need to type in "ATE1Q" ENTER.
Bad connection has locked up the modem.	Manually reset modem or try "ATZ" ENTER.

Data Loss in Direct Mode

This shows up as missing data on the screen.

Possible Causes	Possible Solutions
Video card or terminal baud rate is slower than the communications port connection rate.	Use a faster video card. Select a slower baud rate on the connection. Use a faster baud rate on the terminal.
Input buffer size is set too small.	Restart the communications port using a larger input buffer size with the CP-START command.
System activity is causing Pick to lose serial port interrupts.	Use a faster system or reduce system activity.
The communications port input buffers are not core-locked causing frame faults.	Increase the number of *Max. frame locked comm ports* in the *Global Parameters* screen.
Modem flow control is not set up correctly.	Set up the modem to correctly handle flow control. Pick on Unix is the only version that supports inbound flow control. On other systems it may be possible to drop data if Pick cannot handle the data from the modem fast enough.

Buffer Overrun Problems

This is characterized by the sending computer receiving a bell character back from the remote computer while sending a file. This occurs when the remote computer's type-ahead buffer is overrun.

Possible Causes	Possible Solutions
The TA-ON verb is missing from the current account.	Type-ahead must be turned on for all PowerComm communications.
Slow hard disk subsystem. When Pick writes to the hard disk, it can go away for long periods of time, not allowing other processes to activate. When this occurs, the type-ahead buffer can fill up.	▪ You can use a write-caching disk controller which will fix most of the busy system problems. However, there still may be times when even the disk controller's memory is saturated and Pick must still wait on a disk write. ▪ Send data in smaller chunks. This way, when a buffer overrun is detected, the turnaround time to send the block over again is reduced. With the CP-SEND command you can specify the block size in the options.
If the system is very busy with either CPU or disk intensive activity, it is possible that the receive program cannot empty the input buffer fast enough causing Pick's type-ahead buffer to overflow.	Reduce the size of the communications block you are sending to the system.

Retry Problems Sending and Receiving Data

You are using CP-SEND or CP-RECV and you are receiving lots of <R>'s on the screen.

Possible Causes	Possible Solutions
The XCS-ON verb is missing from the current account.	For 8 bit communications, the XCS-ON command must be used to set the port's input mode to 8 bits.
Full 8 data bits not available on this modem or port connection.	Use the "B" option with the CP-SEND command to use only 7 bit data communications. You can also use the "B" option with the CP-START command to automatically set the "B" option on the CP-SEND and CP-RECV commands.
If the system is very busy with either CPU or disk intensive activity, it is possible that Pick cannot service the serial interrupts at a quick enough pace to prevent loss of data.	Reduce the communications block size to a smaller number of bytes.
Advanced Pick supports unlimited item sizes and Pick R83 has a size limit of 32K. If you send an item from AP to an R83 system that is larger than 32K, the receive program, CP-RECV, will abort and enter the debugger. If this occurs, the sending program will enter a retry phase and eventually terminate with an error message.	Do not send items from AP to an R83 system that the R83 version of Pick cannot store.
Level pushing problems when sending data	
On AP/Unix, you can change the ESC-LEVEL key from escape to another character. If the character you have chosen is used as part of the PowerComm data transfer protocol, this could cause a level push right in the middle of receiving or sending data.	You can change your ESC-LEVEL key using the SET-ESC command, or you can change the protocol characters PowerComm uses in the *Port Definition* screen. If you change the protocol, each computer you communicate with must also have the matching protocol characters.

YModem Send Problems

The remote system aborts a transfer when using YModem protocol.

Possible Causes	Possible Solutions
Remote host will not accept filename.	Use the "D" option to create DOS compatible filenames.
You did not select the "C" option.	Use the "C" option since most YModem programs expect to use CRC packet check,

General XY/Modem Problems

The following causes may be the result of many XModem and YModem problems.

Possible Causes	Possible Solutions
XON/XOFF has not been disabled on the communications port (or the local port).	Use the appropriate command to disable XON/XOFF since XModem and YModem are 8 bit protocols and require all 256 characters.
XON/XOFF has not been disabled on the modem.	Use the appropriate modem commands to disable XON/XOFF since XModem and YModem are 8 bit protocols and require all 256 characters.
You only have a 7 bit data connection with the remote system.	Use the appropriate command to enable 8 bit communications since XModem and YModem are 8 bit protocols and require all 256 characters.
Too many errors when using a modem.	Force the modem to only connect with a reliable connection. On US Robotics modems, this is the "&M5" setting.
On R83, you get the OUT OF WORKSPACE error message when receiving data.	Use the "S"plit option to limit item sizes to reduce workspace usage.

Note: Since the standard XModem and YModem protocols are limited to a maximum of 1024 data bytes, it is not essential that you use hardware flow control since most modern modems contain enough RAM buffer space to buffer a single data packet at this maximum size. However, running the connection rate at a slower rate (like 1200 baud) and the DTE rate (to the modem) at a higher baud rate (like 19200 baud), may cause certain modems to drop the data without hardware flow control. If your modem has this problem, you should start the communications port with the same rate as the anticipated connection rate.

CHAPTER 7
Questions and Answers

This chapter contains many common questions and answers about PowerComm.

File Transfers

How do I prevent received items from overlaying existing data?

Use the CP-RECV command with the "N" (new items only) option.

How do I resume an aborted upload or download?

You may resume an aborted upload or download automatically if you have used the "A" option on the CP-RECV command. When you use the CP-SEND command again, use the "A" option.

How do I use the maximum receive block size when sending files to the remote system?

To ask the remote system what the maximum block size it has been configured to receive, use the "V" option on the CP-SEND command.

How do I send groups of files and TCL commands between systems?

This may be accomplished using the CP-PACKAGE command.

How do I suppress item ID's when sending or receiving files?

Use the "I" option with CP-SEND or CP-RECV.

How can I speed up file transfers?

There are several techniques you can use.

- Use data compression on your modem.

  If you are sending compressible data, use the data compression feature on your modem.

- Increase the block size.

  If your remote system can receive larger block sizes, select a larger send block size. The default block size is configured by port number and defined in the *Port Definition* screen on the main menu. You may also override this value with the command line option using the CP-SEND command.

- Suppress item ID's

  If you are sending or receiving lots of small items, the scrolling of item ID's on the screen can actually slow down the throughput. You can use the "I" option to suppress the display of item ID's. If you still want some visible progress of data transmission, use the "C" option to show communications blocks instead.

How can I move accounts between systems?

This is best accomplished using the CP-PACKAGE command.

How do I abort a file transfer?

The best way is to use the CP-ABORT command. You may also enter direct mode with the communications port and press BREAK and type "END" ENTER.

How do I view received data when using CP-RECV-ASCII?

Normally the view is suppressed. To turn on the viewing of received data use the "V" option. Depending upon your system speed, this may slow down your data transfer throughput.

How do I check to make sure the files I sent to the remote system are the same as the original?

The CP-STAT-FILE command may be used to provide a platform independent method of double checking the success of the file transmission. If you are sending files between R83 and AP systems, make sure the checksum size you can specify as an option will not cause overflow on the R83 system.

How do I view the communications block statistics while using CP-SEND and CP-RECV?

Normally only item ID's are shown as files are transferred. Using the "C" option will show you the communications blocks. In addition, the "$" option will show some additional file transfer statistics at the completion of the command.

What are the characters displayed on the screen during file transfers and what do they mean?

These characters: %, <R>, <S>, and <B> are shown when certain communications events occur. For more details, refer to page 68 in the *Technical Information* chapter.

Direct Mode, Capture and Scripts

How do I find out when another user entered direct mode?

You may view the date and time of a user entering a particular mode with a communications port by using the CP-LIST command with the "T" option. Next to the date and time of communications port start and attach is the mode date and time.

How do I detach another user that is in direct mode?

This may be accomplished with the CP-DET command and the "U" option. To detach port 5 from communications port 2 you can enter "CP-DET 2 (U)".

How do I send a break?

While in direct mode you may press the break key, or the alternate break key. At TCL you may use the CP-SEND-PORT command with the BREAK keyword.

How do I capture my session to a file?

This can be done using the CP-CAPTURE command designating a file and initial item ID to use.

How do I play back a captured session?

The CP-PLAYBACK command is used with a file and the name of the capture ID previously specified with the CP-CAPTURE command.

How do I remove capture items from a file?

Use the "K" option with the CP-PLAYBACK command.

How do I find out what exactly is being received when executing scripts?

You can use the "M" option with CP-SCRIPT to monitor the data received in hex and ASCII during script execution.

Miscellaneous Communications

How do I change the buffer sizes on a communications port?

You may override the default size specified in the *Port Definition* screen by using a command line parameter with the CP-START command.

How do I find out which communications port I have attached?

You may use either the CP-ATT with the "C" option or you may use the CP-LIST command. The CP-LIST command will show all ports. The currently attached port is shown with "=>". To restrict the output to only the current attached port, use "CP-LIST (MM)".

Why is the user 'CP:COMM' logged on to the modem port?

On AP systems, the communications port server program is logged on to the CP:COMM account.

How can I show the comm port is running using LISTU?

Use the "L" option on the CP-START command to update the ACC file for R83 systems, or the PIBS file on AP systems.

How do I send a command to the remote system without using direct mode?

The CP-MSG command is used to send text to a remote port. The CP-SEND-PORT command may also be used.

How do I kill only my own ports?

You may use the "M" option with the CP-KILL command to kill only the ports you have attached.

How do I find out the original port parameters that were in effect before starting a port?

Use the CP-LIST command with the "C" option to view the communications parameters. The original settings are shown under the heading "original settings".

How do I view the current port communications characteristics?

Use the CP-STATUS-PORT command with the port number you want to find out detailed information about.

Using TCL Commands

How do I find out the options for a command at TCL?

Use the "?" option after the open parenthesis. For example "CP-DIR (?)" will display the options for the CP-DIR command. Or you may use the CP-HELP command followed by the names of the commands you want help on.

How can I get a selected list with TCL commands?

You may specify a previously saved selected list by using the "^name" option with the command. For example, "CP-SEND BP ^UPLOAD" will get the list "UPLOAD" and use it to send the items from the "BP" file.

How do I suppress half intensity when using commands?

You may use the "@" option with any TCL command to suppress the initialization of the half intensity on and off variables.

How do I suppress all CP.nnn messages when running TCL commands?

You may use the suppress option five times "#####" to suppress all CP.nnn error messages. Or the "&&" option.

How do I get more information when running TCL commands?

Many commands support the "$" option to specify the display of more information during command execution. Refer to the table starting on page 152 for specific additional information that is displayed for each command.

Menu Operations

How do I remove a value in a screen?

This is accomplished by pressing SPACE and ENTER at a field.

How do I enter multiple options on a menu?

To select multiple options on a menu at a time, enter each option separated by a space.

Miscellaneous

How do I free up some unused frames?

Use the CP-CLEAR command with the "R" option to release frames to overflow.

How can I make sure Pick has flushed all of memory to disk?

R83 Pick does not flush memory to the hard disk unless there is no activity on the system. However, a communications port server maintains a small amount of activity even if no communications is occurring. For this reason, Pick will not flush memory to disk. To pause the communications ports and allow Pick to detect an idle system and begin its flush, use the CP-PAUSE command.

For AP versions you can just use the FLUSH command and skip the CP-PAUSE command.

PART 3

User Reference

CHAPTER 8
TCL Commands Reference

This chapter documents all of the TCL commands available in PowerComm.

TCL Commands by Category

These are the PowerComm TCL commands organized by type of function. Use this list as a quick index to the TCL command reference section following. Page numbers are indicated after the short descriptions.

Starting & Stopping a Communications Port

CP-START	Starts communications port on port specified, 120
CP-KILL	Terminates communications port specified, 104

Attaching & Detaching a Communications Port

CP-ATT	Reserves communications port for your port, 101
CP-DET	Disconnect your port from communications port, 103

Sending & Receiving Data

CP-SEND	Send a file to another system running PowerComm, 115
CP-RECV	Receive a file from another system running PowerComm, 110
CP-ABORT	Aborts a CP-SEND or CP-RECV data transfer on the remote system, 101
CP-SEND-SPOOL	Sends spooler data to another system running PowerComm, 119
CP-RECV-SPOOL	Receives spooler data from another system running PowerComm, 113
CP-SEND-ASCII	Sends data to a remote computer in ASCII text format, 116
CP-RECV-ASCII	Receives data from a remote computer in ASCII text format, 111
CP-SEND-PICK	Sends a file to a remote Pick computer in dumb mode, 117
CP-RECV-PICK	Receives a file from a remote Pick computer in dumb mode, 112
CP-SEND-XMODEM	Sends a file to a remote computer using XModem protocol, 120
CP-RECV-XMODEM	Receives a file from a remote computer using XModem protocol, 113
CP-SEND-YMODEM	Sends a file to a remote computer using YModem protocol, 120
CP-RECV-YMODEM	Receives a file from a remote computer using YModem protocol, 113

Direct Mode

CP-DIR	Enter direct mode with currently attached communications port, 103
CP-CAPTURE	Enter direct mode with currently attached communications port and capture all data to the capture file specified, 102
CP-PLAYBACK	Plays back to the screen a file captured previously with CP-CAPTURE, 109

Other Commands

CP-CLEAR	Clears the communications port tables, 102
CP-HELP	Displays on-line help for commands specified, 104
CP-LIST	List out currently running communications ports, 105
CP-MENU	Runs the PowerComm main menu, 105
CP-MSG	Send text out currently attached communications port, 106
CP-PACKAGE	Create or extract from a package of embedded files, accounts or TCL commands, 107
CP-PAUSE	Pauses all communications ports for number of seconds specified, 109
CP-SCRIPT	Execute a PowerComm script to the current attached communications port, 113
CP-SEND-PORT	Sends special actions or characters to the currently attached comm port, 118
CP-STAT-FILE	Computes length and checksums for items specified, 123
CP-STATUS-PORT	Displays port status of port specified, 124
CP-VERSION	Displays software version information of PowerComm, 124
CP-VIEW-TCL	Displays changed output from continuous TCL command execution, 124

TCL Command Syntax

PowerComm TCL commands usually take one of two forms. Those that reference files and those that do not.

The following syntax is for the TCL commands that do not reference a file:

tcl.command parameters ... (options) command ... {input} {...}

PowerComm TCL commands that reference a file and an item list take on the following general syntax:

tcl.command fileref itemlist (options) command ... {input} {...}

Please Note: The braces "{ }" in the above command line formats do not represent an optional component which is used in some documentation. These braces are part of the stacked input syntax and are used to surround stacked input.

The following section documents each of the command line parameters in more detail.

tcl.command

PowerComm TCL command from the list of available commands.

parameters ...

Parameters vary according to the command. Refer to the specific command for information about the meaning of each parameter.

Any command that uses **portname** as a parameter may also use the comm port number as well.

Example:

```
CP-ATT 3
CP-ATT P3
```

These two examples are equivalent assuming port 3 was named "P3" when started.

fileref

File name reference.

The format of the file name reference may be any valid Pick file reference including file pathing on AP.

Standard File Reference Formats

Fileref Format	Description
filename	This form of a file reference accesses the data section of a given file.
DICT filename	This form accesses the dictionary level of a file
dict.filename,data.filename	This accesses a data section in the given dictionary, when the data section's name is different from the dictionary's name. This format is used when a file has multiple data sections.

Example:

```
CP-SEND LSCP,BP.EXAMPLES *
```

This example sends all items in the "BP.EXAMPLES" data level file off the "LSCP" dictionary.

AP "File Pathing" File Reference Formats

Fileref Format	Description
account.name,,	This accesses another MD (account), or any file defining item found in the "MDS" file.
mds,md.name	This is the "fully qualified" format for referencing another MD.
account.name,filename,	This form accesses the data section of the filename in the account name specified.
dict account.name,filename,	This form accesses the dictionary section of the filename in the account name specified.

Example:

```
CP-SEND dm,bp, *
```

This example sends all items in the "bp" file in the "dm" account.

Additional PowerComm Account Name File Reference Format

PowerComm provides an additional feature to the file reference that allows you to supply an account name in front of the file reference. This feature may be used on both R83 and AP. It provides a platform independent mechanism for referencing files in other accounts without having to pre-establish a "Q" pointer to that file.

Fileref Format	Description
@acctname fileref	Specifies an optional account name. The "@" sign must be used immediately in front of an account name to designate the following text represents an account name instead of a file reference. The file reference must come after the account name separated by a space.

Example:

```
CP-SEND @SYSPROG KEYBOARDS *
```

This example sends all items in the "KEYBOARDS" in the "SYSPROG" account.

Note: Refer to your Pick reference documentation for more information about file references and file pathing on AP.

itemlist

The item list may take several different forms including any standard pick item list format. Item ID's may be quoted by surrounding them with any of the following delimiters:

"	double quote
'	single quote
\	backslash

Standard Item List Formats

The following is the standard Pick item list formats:

Itemlist Format	Description
item-id	A single item ID.
item-id item-id ...	Multiple item ID's separated by spaces.
*****	All items in the file in hash file order. A pending selected list will be discarded if present.
(none)	May be used if a selected list is pending.

Example:

```
CP-SEND CUSTOMER 1 \2\ "BUBBA'S"
```

This example sends 3 items from the "CUSTOMER" file.

Example:

```
CP-SEND BP *
```

This example sends all items in the "BP" file.

Example:

```
SELECT BP = "GL.]"
CP-SEND BP
```

This example sends items from the "BP" file using the results from the SELECT command.

Additional PowerComm Item List Format to Automatically Sort Select a File

The following is an additional PowerComm item list format that allows you to automatically sort select a file into one of four orders; ascending left-justified, ascending right-justified, descending left-justified and descending right-justified.

Using "*" by itself, described above, will discard a pending list. However, these additional four formats will use a pending selected list if present. This capability allows you to re-sort a pending list into the sequence you want. This option works by using the SSELECT verb against the file specified on the command line.

Itemlist Format	Description
***AL**	Items are sorted into ascending left-justified sequence.
***AR**	Items are sorted into ascending right-justified sequence.
***DL**	Items are sorted into descending left-justified sequence.
***DR**	Items are sorted into descending right-justified sequence.

Example:

```
CP-SEND BP *AL
```

This example sends all items in the "BP" file sorted into ascending left-justified sequence.

Example:

```
SELECT ORDERS WITH STATUS "S"
CP-SEND ORDERS *DR
```

This example first selects the ORDERS file, then sends the ORDERS file using the item ID's from the results of the selected list sorted into descending right-justified order.

Additional PowerComm Item List Format to Get a Saved List

The following is an additional PowerComm item list format that allows you to get a selected list. On AP, the GET-LIST command supports getting multiple lists from the same file and combines the lists into a larger selected list. This is shown below by the "..." syntax.

Itemlist Format	Description
^listname ...	Specifies the name of a list to use that must be available using the GET-LIST listname command.
^filename listname ...	On AP, the listname may be a file name, space and an item.

Example:

```
CP-SEND BP ^UPLOAD
```

This example sends items from the "BP" file using a list called "UPLOAD".

Example:

```
CP-SEND BP ^LISTS UPLOAD1 UPLOAD2
```

This AP example sends items in the "BP" file using the results of combining a the list called "UPLOAD1" and a list called "UPLOAD2" from the "LISTS" file.

Additional PowerComm Item List Format to Execute a TCL Statement

The following is an additional PowerComm item list format that allows you to execute a TCL statement that generates a selected list. The TCL statement may not contain a left parenthesis "(". Everything from the "^^" to the left parenthesis is considered the TCL statement.

Itemlist Format	Description
^^tcl.statement	Specifies the TCL statement to execute which should return a selected list of items. It may be a SELECT, a GET-LIST, a QSELECT, or any TCL statement that will generate a list of items.

Example:

```
CP-SEND BP ^^SSELECT BP = "GL.]" AND # "[BAK]"
```

This example sends items in the "BP" file using the results from the SSELECT statement.

Note: Refer to your Pick reference documentation for more information about item lists.

options

Specifies options for the command.

Options may be specified in any order, separated by spaces, commas or not separated at all. In this documentation, commas are shown in examples for readability but may be omitted.

Important Note: A closing parenthesis ")" is not required unless you are using the command line parameters called "command" and "input". If no options are required for the command and you need to use "command" and "input", simply use "()" in the command line as a place holder.

Example:

```
CP-MENU PD () {G}
```

This example runs the PowerComm menu, selects menu option "PD" and passes the "G" as stacked input to the *Port Definition* screen.

Special Options

The following are special options that can modify the output of certain messages from commands. Refer to the *TCL Command Message Control Table* starting on page 152 for details about which messages can be suppressed or added to which commands.

Option	Description
@	Suppresses the half intensity variable assignment. This may be useful when you wish to capture the output of a command and you do not want the escape sequences in the captured output. Note that you can also change the half intensity characters globally in the *Global Parameters Definitions* screen.

Suppresses certain messages from programs.

This option is cumulative and may be specified from 1 to 5 times depending on the level of message suppression you need.

Five pound signs "#####" is special and will suppress all CP.nnn messages for all commands. See also "&&" below.

$ Adds extra messages to certain programs.

This option is cumulative and may be specified from 1 to 5 times depending on the level of additional messages you need.

! Debug mode. (Unsupported).

This unsupported option is cumulative and may be specified from 1 to 5 times depending on the level of debug messages you want to see. These are for us to help with debugging and monitoring. They are not documented, nor supported. Use at your own risk.

& Saves generated "CP.nnn" error messages and parameters to a file.

This option saves the error messages generated by the command to the LSCP.TBL,BUFFER file with an ID of "EMSG*n" where 'n' is the port number executing the command (not the communications port). The error messages and parameters are separated by sub value marks in value 1, and the date and time is in value 2. Each message is in a new attribute.

&& Same as "&" but also suppresses the output of the error message to the screen.

This option may be preferred over using "#####" (above) when you do not wish to suppress some of the error messages that using "#" causes.

Example:

```
CP-LIST (@,$$
```

This example prevents the CP-LIST command from using half intensity in its display.

Example:

```
CP-DIR (##
```

This example suppresses the normal messages "direct mode started" and "direct mode exited" from being displayed.

Example:

```
CP-SEND CUSTOMER * ($,&
```

This example generates an extra message about communications transmission statistics. In addition, the "&" option added information to the 'buffer' file:

```
    LSCP.TBL,BUFFER EMSG*1
001 CP.048\CUSTOMER\1\124\2\0.05\8]10573\55633
002 CP.055\0\1\0\1]10573\55633
```

command

Specifies a command line command for certain TCL commands.

This field comes after the options "()" and can pass additional parameters to PowerComm TCL commands that use this feature.

Example:

```
CP-PACKAGE PKGS UPLOAD (A) ACCOUNT QA
```

This example specifies a package command that adds the "QA" account to the package named "UPLOAD" in the "PKGS" file.

{input} ...

This specifies stacked input to the command.

Each stacked input must be enclosed within braces "{" and "}", All characters are permitted between braces, including spaces and quotes. However, you may not use braces within braces.

Example:

```
CP-DIR (A) {1B} {03}
```

This example runs the CP-DIR command and supplies the "A" ask option with values for two questions.

When multiple inputs are available with certain commands that take input, the order of input is in the same order as the alphabetic option letter.

Example:

```
CP-SEND-PICK BP * (U,R) {BP.REMOTE} {null-line}
```

This particular command uses two options that can supply parameters. To avoid ambiguity between which input comes in what order, we have ordered the inputs according to the alphabet.

The "R" question comes before the "U" question regardless of the order specified in the options.

Standard Options on Some TCL Commands †

The following options are available on many of the PowerComm TCL commands documented in this section. To avoid repeating the lengthy descriptions for each option on each command, they are shown here for reference. Wherever they are used in a command, they are marked in the documentation with the † symbol after the short option description. When you see that symbol, refer back here for the detailed information.

Option	Description
B	7 bit binary mode. This option may be used when a full 8 bit communications connection is not available. All characters with the high order bit set will be masked for transmission. This will add a little additional overhead and time to communications depending upon the data being transmitted.
C	Communications buffer display mode. This option is used to show the lower level communications block information as files are transferred. The information is indented from the left in order to differentiate it from item ID's if they are displayed. The "I" option was used to suppress the item ID's sent. The columns shown with the "C" option are as follows: the block number total bytes successfully sent or received so far current block size time elapsed in minutes average transfer rate in characters per second
F	Through a file (used in conjunction with the "M" option). This option, together with the "M" option provides a mechanism for certain commands to operate through a special file instead of a serial communications port. This is only used when you want to test some commands without having to utilize a serial port. Refer to the section, *Me Mode Through a File*, starting on page 79 for additional information.
I	Item ID suppress. This option is used to suppress the item ID's of transferred items.
L	Line display mode. This option, together with the "C" or "L" option causes the output from the "I" or "C" option to occupy one line on the screen. By default, the "I" and "C" options scroll down the screen.
M	"Me" mode (not through a communications port). This option selects the "Me" mode of operation for the command. The communications for this command will happen through the same port and not a comm port on the system. Refer to page 30 for additional information about the "Me" mode.
T	Timeout mode (used in conjunction with "M" option).

This option together with the "M" option turns on the timeout mode for the command. By default, no timeout occurs on the "Me" mode end of a communications connection. To change the timeout value for this setting, refer to the *Port Definition* screen, field "*Packet ack. timeout*" starting on page 129. In addition, the "T" option also assumes a slower 1200 baud connection for computing the adjusted timeout value. Refer to page 70 for additional information on how PowerComm computes timeout values.

X Enable XON/XOFF flow control (used in conjunction with "M" option).

This option together with the "M" option enables XON/XOFF flow control. By default, XON/XOFF is disabled to reduce the possibility of control characters from inadvertently stopping data transmission.

Getting Help from TCL

You can get on-line help by typing in the command and using the "?" option in parenthesis. See the explanation of CP-HELP for more options.

Note: Longer help screens will automatically paginate at the screen depth associated with the TERM command. To avoid pagination, use "(?>)" instead of "(?)".

TCL Commands (In Alphabetical Order)

The following is a listing of all of the TCL commands in alphabetical order.

CP-ABORT

Aborts a CP-SEND or CP-RECV data transfer on the remote system.

CP-ABORT (options)

options Options may be any of the following:

A Suppress the 'abort message'.

This option suppresses sending the error message that indicates the abort was caused by a user abort.

C Cancel the remaining data in the input buffer coming from remote system.

D Enter direct mode after abort.

F Through a file (used in conjunction with the "M" option). †

This option, used with the "M" option uses the "File" mode of operation to cancel the send or receive in process.

M "Me" mode (not through a communications port). †

This option has no meaning unless it is used together with the "F" option. There is never an instance where you would abort your own port.

P Prefix the 'abort packet' with a 'terminate prior packet' sequence.

This option can yield more reliable aborts when the receiver has lost the end of the prior packet.

See Also

- CP-SEND for related information on page 115.
- CP-RECV for related information on page 110.
- The section, *Aborting File Transfers with CP-ABORT*, starting on page 40.

CP-ATT

Attach communications port specified.

CP-ATT portname (options)

portname Name of communications port previously started with CP-START command

options Options may be any of the following:

C Check only.

This option does not change the current attachment. **Portname** is ignored with this option.

D Enter direct mode after attach.

This option will invoke the CP-DIR command after successful attachment.

See Also

- CP-DET for related information on page 103.
- The section, *Using CP-ATT to Attach a Communications Port*, starting on page 32.

CP-CAPTURE

Enter direct mode with communications port specified and capture all data to a capture file specified.

CP-CAPTURE fileref ID (options)

fileref Capture file name.

ID Optional capture item ID.

If the capture item ID is omitted, a capture ID of the following format will be created:

MM-DD-YY/HH:MM:SSxx

Example:

02-11-95/01:09:32PM

options Options may be any of the options provided with the CP-DIR command.

Example:

```
CP-CAPTURE CAPTURE.FILE TODAY
```

This example enters direct mode and creates or appends to a capture item called "TODAY".

See Also

- CP-DIR for additional information on page 103.
- The technical information section on *CP-CAPTURE* starting on page 64.

CP-CLEAR

Clear communications port tables. This command may also be used to release overflow space before removing PowerComm.

This command should be a part of the coldstart procedure. A program called LSCP.COLDSTART is installed during installation and documented with more details on page 23.

CP-CLEAR (options)

options Options may be any of the following:

A Release all buffers to overflow space.

This option used with the "R" option releases all frames that have been allocated for PowerComm's data buffers even if they are allocated to a specific port. Use this option with care, since this may cause GFE's if other processes are actually still using these frames. The only time you may use this is when you have aborted a program and the frames are still allocated to ports that are no longer using these frames. And, you want to run an overflow table rebuild routine.

I Inquire communications port table status and frames in use.

This option reports whether or not the communications port table is active or not for all communications ports. In addition, it reports how many frames are used by various tables and buffers. This option may not be combined with any other option.

R Release available unused buffers to overflow space.

This option releases frames that have been allocated for PowerComm's data buffers that are not in use. Use this option if you need a few additional frames in an emergency or you are removing PowerComm from the system.

U Unconditionally clear all ports, even if found to be running.

This option will kill all running ports before the comm port tables are cleared. This could have undesirable results if any communications programs are currently accessing the comm ports at the time of clearing. Use with caution and only when no programs using the communications software are running.

To release space to overflow before running an overflow rebuild routine, make sure no communications ports are running and use the following options with the CP-CLEAR command:

```
CP-CLEAR (A,R,U
```

This will release all buffer frames to overflow.

Important Caution: Use the "A" option with care. Use only when you are absolutely sure that no other PowerComm programs are running. Otherwise, you may end up with GFE's on your system.

CP-DET

Detach the specified communications port from current port.

CP-DET portname (options)

portname Name of communications port previously started with CP-START command and attached with CP-ATT.

options Options may be any of the following:

A Attach after detach.

This option will automatically attach the port after the port was successfully detached from your port or another port.

D Enter direct mode after attach.

This option, used with the "A" option, will automatically enter the direct mode of communications after the port is attached.

U Unconditionally detaches.

This option detaches the specified communications port from another port that is in any of the following modes:

direct mode
script mode
ASCII upload
ASCII download

The user in any of the modes will be detached from the port, their program terminated and the port made available for any other user. Use can use the CP-LIST command to see which mode the port is in.

See Also

- Using CP-DET starting on page 34.

CP-DIR

Enter direct mode with communications port specified.

CP-DIR (options)

options Options may be any of the following:

A Ask for exit and alternate break characters to use for exit and break.

This option allows you to dynamically change the exit key and alternate break key at runtime without having to change it in the menu. This is a temporary change for the execution of the command only.

You may use stacked input with this command, however the stacked input is the character to use in hex format. (See the example following).

C Clear input buffer.

This option clears the input buffer of the communications port.

M Monitor mode.

This option displays received data in both ASCII and hex modes. It is not available for capture mode. See the "n" option to control display characteristics.

On R83, the true break key will not be acknowledged in this mode. Instead, use the alternate break key. On AP, it is possible after using the real break key that the program calling CP-DIR will enter the debugger when direct mode is exited.

Q Sends CTRL+Q before entering direct mode.

This option may be used to resume output from a remote port that is I/O blocked with XOFF. CTRL+Q is the XON flow control character.

R Reset port.

This option may be used on AP versions to reset the communications port if it is locked up. It uses the TCL command "RESET-PORT".

T Time how long in direct mode.

This option displays an additional message after exiting direct mode showing how many minutes direct mode was running.

n Maximum bytes shown at a time with hex data in monitor mode.

This option, available only with the "M" option, specifies the maximum amount of data that can be shown before viewing the corresponding hex data.

Example:

```
CP-DIR (A) {1B} {}
```

This command invokes the CP-DIR command and changes the exit key to ESCAPE (hex "1B") on the command line, but leaves the alternate break key definition unchanged.

Example:

```
CP-DIR (A) {} {03}
```

This command invokes the CP-DIR command and changes the alternate break key to CTRL+C but leaves the escape key unchanged.

See Also

- CP-CAPTURE for additional information on page 102.
- The section, *Using CP-DIR to Enter Direct Mode*, starting on page 36.

CP-HELP

Show on-line help for the command entered. In addition, help for all TCL commands is shown by entering the command followed by "(?)".

CP-HELP command ... (options)

command Optional command (from this chapter) to display or print. If no command given, a list of all available commands is output.

options Options may be any of the following:

P Send output to printer.

Use this option to send the information to the currently assigned printer via the spooler.

See Also

- The section, *Getting Help from TCL*, starting on page 59.

CP-KILL

Terminate communications ports specified.

CP-KILL portname ... (options)

portname ... Communications port to terminate. The portname may be either the name or the port number. Multiple portnames are separated by spaces.

options Options may be any of the following:

A Kill all ports started that are not attached to other ports.

This option, with the "U" option will also attempt to kill all ports even if they are attached to other ports.

M Kill my ports only.

This option will kill only those ports that you have attached. Use "MM" to kill your currently attached port only.

U Unconditionally kill.

This option invokes CP-DET (U) for a comm port that is not attached by your process. If successful, it will be killed.

Example:

```
CP-KILL (A,U
```

This is a quick way to kill all ports even if not yours. It in turn uses CP-DET with the "U" option to try and detach each port.

See Also

- CP-DET for related information on page 103.
- The section, *Using CP-KILL to Stop Communications Ports*, starting on page 35.
- The technical information section on CP-KILL starting on page 73.
- CP-CLEAR for clearing communications ports starting on page 102.

CP-LIST

Lists out all communications ports running.

CP-LIST port.range (options)

port.range This optional parameter may be a single port number or a range separated by a dash "-". If none specified, all ports will be listed.

options Options may be any of the following:

C Show communications parameters.

This option shows communications parameters such as baud rate, parity, word length, and stop bits. The original port parameters are also shown before they were changed with the CP-START command.

H Suppress headers.

This option suppresses the headings shown with the information.

M Show only my attached ports.

This option shows the ports that I have attached. The current attached port is shown with "=>". To restrict the listing to only the currently attached port, use "MM".

P Send output to printer.

Use this option to send the information to the currently assigned printer via the spooler.

T Show time / date of start and attach.

This option shows the time the port was started and the time of last attachment. Use "TT" to see the duration of each category listed.

Example:

```
CP-LIST (C,T
```

This example displays the comm ports with the start and attach times along with the communications parameters.

See Also

- Several examples using CP-LIST starting on page 33.
- Technical information about communications ports starting on page 64.
- Technical information about interpreting the CP-LIST command starting on page 65.

CP-MENU

Invoke the PowerComm menu system.

CP-MENU selection ... (options)

selection An optional menu selection.

The menu selection may be a number or a keyword on the menu. If no selections are specified, the main menu is run and the user is prompted for a menu selection.

If the selection is present, the menu display is suppressed. Multiple selections may be entered separated by spaces. A period "." may be used to represent pressing ENTER at a menu selection prompt.

options Options may be any of the following:

K Display menu keywords instead of numbers.

This option turns on the menu selection keywords on the menus. However, it is not needed in order to select the keywords from a menu.

Example:

```
CP-MENU PD
```

This example runs the *Port Definition* screen on the main menu.

Shortcut Tip: CPM is a short cut for typing CP-MENU.

CP-MSG

Send text to currently attached comm port followed by CR. By default, the command waits for the message sent to echo back exactly as sent. If it does not match, an appropriate message will be displayed.

CP-MSG "text" ... (options)

"text" Text to send to communications port.

If no spaces are in the text, the quotes may be omitted. Multiple messages may be separated by spaces. However, when using multiple messages, the message matching will not work correctly because the output of the first message could affect the matching of the subsequent message. In this case, use the "T" option to suppress the timeout and matching.

options Options may be any of the following:

C Suppress sending the automatic CR after sending message text.

D Enter direct mode after sending message text.

H Hex output. Automatically invokes the "C" option.

The "text" must be in hexadecimal format. For example, the word HELLO would be "48454C4C4F" in hex format.

Q Sends CTRL+Q before sending message text.

This option may be used to resume output from a remote port that is I/O blocked with XOFF. CTRL+Q is the XON flow control character.

T Suppress timeout and match. Also renders option "V" unavailable.

V View text sent as received by communications port.

This option views the text that is being received (as a result of echo) by the command until the timeout period expires.

X Sends CTRL+X before sending message text.

n Wait for "n" seconds before timing out (default is configured in the *Port Definition* screen).

The default timeout value may be adjusted in the *Port Definition* screen, field "*Default message packet timeout*" beginning on page 129

Example:

```
CP-MSG "CP-RECV BP.REMOTE (M" (15
```

This command sends the string to the comm port and waits for echo verification for up to 15 seconds.

Example:

```
CP-MSG "01" (H,T
```

This command sends CTRL+A with no carriage return or timeout.

See Also

- The section, *Using CP-MSG to Send a TCL Command*, starting on page 51.
- CP-SEND-PORT for related information on page 118.

CP-PACKAGE

Create or extract from a package of embedded files, accounts or TCL commands.

CP-PACKAGE fileref pkgname (options) command

fileref Specifies the file name of where the package is located for 'e'xtract mode or where to put the package during 'a'dd mode.

pkgname Specifies the name of the package to create or extract from. Do not use "*" in the 'pkgname' because “*” is used to separate the package name from the numerical sequence numbers that are created.

options Options may be any of the following, however these three options are mutually exclusive — they can only be used one at a time:

A	Add to package
E	Extract from the package
V	View the package

A The commands and resultant data are 'added' to the file. The opposing option is "E" for extracting from the package.

C The commands are stored in the item "pkgname" in the file specified on the command line.

This option is equivalent to using the READ command with the same file and package name on the command line. See the examples for more details.

D Ask for destination. (E)

This option causes the extract program to prompt for an alternate destination file name each time it encounters a new DICT or DATA command.

E Extracts all of the data and embedded commands from the named package. The opposite is the "A" option for adding to the package. To view the contents of the package without extracting, use the "V"iew option.

K When used with the "A"dd option, "K" removes old package items before processing any commands.

When used with the "E"xtract option, package items are removed from the package file as they are processed. This can be useful when you are low on disk space and you want to remove the package items as you move the data from the package to the target files.

When used with the "V" option, package items are removed after they are viewed. This is a way of removing the package items from the file without using other TCL commands.

L Create a list.

This option, when used with the "A" option, selects the package file after creating the package data. You are prompted for a list name to save the list with. This feature is useful for subsequent uploading of the data.

O Override "D" pointer checking on "DICT" files.

This option overrides the checking of writing or overwriting of existing "D" pointers on "DICT" files. It also suppresses the checking of overlaying "CC" and "CL" items on R83. Using this option is not a good idea unless you are 100% sure of not creating items on the target system that Pick may think of as a file definition item or a BASIC object code or list pointer item. **THIS IS NOT A RECOMMENDED OPTION.**

R Prompt for renaming the account when "E"xtracting. Unless the "R" option is specified, the extract will update an existing account if found. Otherwise, the account will be created.

This option is only relevant when extracting data after having used the ACCOUNT command to store an account in the package. See the examples for more details.

S Suppress runtime messages and display stars "*" instead. To suppress stars, refer to the additional options on page 152.

V Views all of the commands in the package and does not extract the data.

This command is used to verify the integrity of the package. Each time the package "A"dd option is used, a timedate stamp is added before adding data and after adding data to the package.

The view option shows the start timedate stamp with "$S", user commands with "$$", and the ending timedate stamp with "$E".

X Ignore "DX" logic where found.

This option overrides the normal behavior of skipping "DX" files. This option works like "X" option on the SAVE verb on AP.

Y Ignore "DY" logic where found.

This option overrides the normal behavior of not saving the data found in "DY" files. This works like the "Y" option on the SAVE verb on AP.

n Specify target item size override. Where "n" is a number that overrides the default item size target of 5000 bytes.

command Commands may be any of the following list of commands that may only be specified during "a"dd mode:

! comments Lines preceded with "!" are comment lines.

*** comments** Lines preceded with "*" are comment lines.

ACCOUNT acountname The account named "accountname" is added to the package. When the account is extracted, the account is created if it does not exist. If any of the files extracted do not exist, they will automatically be created. Note that when extracting data, existing data is overwritten with new data in the package. However, existing data in the file is NOT cleared before adding new data. Also, program object code is never stored in a package.

DATA filename itemlist Add records to the package from the named file.

Where "itemlist" is:

Item-id item-id ... List of items to add separated by spaces, or

^listname Get list named "listname", or

. Use a selected list previously gotten with a TCLA or TCL command.

DICT filename itemlist Same as "data" except the named file's dictionary is used instead. See the "DATA" command above for more information.

READ filename itemname Read in the commands from the item "itemname" stored in the file "filename".

This is the best way to run this program as it affords a way to edit a definition of what the "package" is going to contain. See the examples section to see an example of how this works. Typical usage would be:

>PACKAGE pkgfile pkgname (ops) READ pkgfile pkgname

Where all of the commands to execute are in the "pkgfile", item "pkgname".

The shortcut for this is to simply use the "C" option.

Note that the command item may have READ commands in it as well. It is recursive.

QFILE account filename Sets up a "qfile" to the named "account" and "filename". This is useful for accessing data or dictionaries in other accounts where there may not be a Q pointer on the target machine.

TCLA tcl\data The TCL command "TCL" is executed during the "A"dd mode and stacked data may be present for the command separated by subvalue marks (control "\").

TCLE tcl\data The TCL command "TCL" is executed during the "E"xtract mode and stacked data may by be present for the command separated by subvalue marks (control "\").

TCL tcl\data Same as TCLA or TCLE except that the command happens both at "A"dd time and "E"xtract times.

Example:

```
CP-PACKAGE PKGS U (A,K) DATA BP ^BP.MODS
```

This example creates a package item "U" in the "PKGS" file. The "K" deletes the existing package, if present, and the "A" option adds to the file the items in the "BP" file from the list "BP.MODS".

Example:

```
CP-PACKAGE PKGS U (A,K,C)
CP-PACKAGE PKGS U (A,K) READ PKGS U
```

These two commands are equivalent. The first uses the "C" option to tell the package program to find the commands in the item "U" in the "PKGS" file. The second form uses the READ command to do the same.

Example:

```
CP-PACKAGE PKGS U (E,R)
```

This example extracts the data from the "U" package in the "PKGS" file. The "R" option will prompt for an alternate account name if and when accounts are extracted from this package.

Important Note: PowerComm employs a certain level of checking concerning writing or overwriting of "D" pointers to dictionary level files. However, this checking is only invoked if you use the word "DICT" as part of the file name. If you omit the "DICT" option, PowerComm will not perform this check. You may, at your own peril, override the checking option with DICT files by using the "O" option. (See the option description for further information and precautions).

Caution: On R83, the package program is not smart enough to know the file is a "DC" file pointer and check only the "DC" files when checking for "CC" or "CL" items. In other words, all dictionary files are checked when using "DICT" unless the "O" option is used. In addition, if you have "CL" items in a data level file, and the extract file is a "DC" data level file, you may inadvertently create file corruption problems on the target system.

See Also

- The section, *Using CP-PACKAGE to Help Move Groups of Files or Accounts*, starting on page 51.
- The technical information section on *CP-PACKAGE* starting on page 66.

CP-PAUSE

Pauses all communications ports for the number of seconds specified. This command is provided as a way to allow Pick to sense an Idle period and force flush write required frames to the hard disk. The amount of pause to allow the system to completely flush depends on the version of Pick, speed of the hard disk subsystem and the amount of write required frames in memory at the time.

CP-PAUSE seconds (options)

seconds Specifies how many seconds to pause all active communications ports.

The maximum number of seconds is 60 seconds. If no time is specified, the pause time will be 10 seconds.

options Options may be any of the following:

U Unconditional pause, even if active or in use.

This option will force all the comm ports into a pause for the number if seconds specified even if they may have been active or in use at the time. The success of the pause is not guaranteed if there is active I/O because I/O takes precedence over a pause.

Any ports running in any mode, including direct mode, will most likely experience a pause in I/O for the specified seconds.

See Also

- Technical information about comm ports starting on page 64.

CP-PLAYBACK

Plays back to the screen a session previously captured with the CP-CAPTURE command.

CP-PLAYBACK fileref itemlist (options)

fileref Name of capture file previously specified with the CP-CAPTURE.

itemlist Standard item list format.

options Options may be any of the following:

C Clear screen before playing back each item.

K Kill (delete) playback items from capture file.

This option deletes the capture records from the capture file specified. It does not play them back to the screen when using this option.

O Only show one sequence number n or 1 if n not specified.

P Send output to the printer.

W Wait at end of each played back item for a single keystroke before resuming.

n Sequence number to start playback.

Example:

```
CP-PLAYBACK CAPTURE * (C,O,W,1
```

This will play back each capture item in the "CAPTURE" file by clearing the screen, displaying the first record found, and waiting for a keystroke before showing the next item found.

Note: The terminal emulation in use at the time the capture file is created may effect the playback on a different emulation resulting in corrupted screen displays and possibly a locked keyboard.

See Also

- CP-CAPTURE for related information on page 102.
- The section, *Using CP-CAPTURE to Enter Direct Mode with Capture*, starting on page 36.

CP-RECV

Receives a file from the currently attached communications port or the current port if using the "M" option. Any existing items in the receive file will be overwritten unless the "N" option is used.

CP-RECV fileref itemlist (options)

fileref Name of file to receive.

itemlist Optional list of item names used for renaming the sending item ID's.

options Options may be any of the following:

A Abort recovery option.

This option makes the receive program save the last item ID it saves every time it writes a record to a file. At the expense of a little additional overhead, this option provides a mechanism to resume an aborted file transfer by using the "A" option with a subsequent send command.

B 7 bit binary mode. †

C Communications buffer display mode. †

F Through a file (used in conjunction with the "M" option). †

I Item ID suppress. †

L Line display mode. †

M "Me" mode (not through a communications port). †

N New items only.

This option prevents the receive program from overwriting any existing records in that target file.

O Override "D" pointer checking on "DICT" files.

This option overrides the checking of writing or overwriting of existing "D" pointers on "DICT" files. It also suppresses the checking of overlaying "CC" and "CL" items on R83. Using this option is not a good idea unless you are 100% sure of not creating items on the target system that Pick may think of as a file definition item or a BASIC object code or list pointer item. **THIS IS NOT A RECOMMENDED OPTION.**

R Remove items received from target file.

This option will delete each item ID received from the target file. When using this option you can use the "R" option with the CP-SEND command to speed up the process by only transmitting the item ID's and not the item body.

T Timeout mode (used in conjunction with "M" option). †

X Enable XON/XOFF flow control (used in conjunction with "M" option). †

Example:

```
CP-RECV BP (I
```

This will receive data from the currently attached comm port from the remote system into the "BP" file suppressing item ID's on the screen as data is received.

Important Note: PowerComm employs a certain level of checking concerning writing or overwriting of "D" pointers to dictionary level files. However, this checking is only invoked if you use the word "DICT" as part of the received file name. If you omit the "DICT" option, PowerComm will not perform this check. Therefore, if you wish to send the MD, or the SYSTEM file and you would like PowerComm to filter out "D" pointers, specify "DICT MD" or "DICT SYSTEM". You may, at your own peril, override the checking option with DICT files by using the "O" option. (See the option description for further information and precautions).

Caution: On R83, the receive program is not smart enough to know the file is a "DC" file pointer and check only the "DC" files when checking for "CC" or "CL" items. In other words, all dictionary files are checked when using "DICT" unless the "O" option is used. In addition, if you have "CL" items in a data level file, and the receive file is a "DC" data level file, you may inadvertently create file corruption problems on the target system. PowerComm does not support sending of "CL" items.

See Also

- CP-SEND for related information on page 115.
- The section, *Using CP-SEND and CP-RECV to Send and Receive Files*, starting on page 38.
- The section, *Using the Aborted Send and Recover Options*, starting on page 41.
- The technical information section on *CP-RECV* starting on page 67.
- The section, *Precautions about Sending and Receiving Files*, starting on page 83.
- The section about changing the send and receive packet sizes in the *Port Definition* screen, field "*Default S/R packet size*", starting on page 129.

CP-RECV-ASCII

Receives a file to from the remote computer in ASCII text format. No protocol or error checking is performed. The remote end should be in ASCII send mode prior to starting this command.

CP-RECV-ASCII fileref item-id (options)

fileref Name of file to receive the data from the remote.

item-id Item id unless the "N" option is used.

options Options may be any of the following:

C Clear input buffer.

This option clears the input buffer of any pending data before starting the ASCII receive.

D Enter direct mode at completion.

I Item ID suppress. †

M "Me" mode (not through a communications port). †

Using this option precludes the use of the following options; "I", "D", and "V". See the "n" option for specifying a start delay.

N Item names are in data.

This option automatically expects the first attribute to contain the item ID. The item body follows the item ID until a null character.

P Pace mode.

This option will prompt for a pace character.

The pace character must be in hex format. Typically this would be CR which is CHAR(13) or hex "0D". To skip pacing, do not use this option. If pacing is not used, the computer must be fast enough to receive data in streaming mode from the remote.

Pace mode sends the pace character to the remote after receiving a line of data. The remote computer, using pacing mode, then sends the next line of data and so on.

S Split records.

This option then prompts for number of bytes maximum for each received item. When the number of bytes reaches this maximum a sequential number is appended to the original ID specified separated by "*". This option is not available with the "N" option.

T Translation mode.

This option automatically translates carriage returns to attribute marks; CHAR(13) → CHAR(254).. Line feeds, when found, are always deleted; CHAR(10) → NULL.

X Use XON/XOFF flow control.

This option used only with the "M" option flow controls the received data using XON/XOFF.

V View mode.

This option turns on the view of received data as it is received from the remote. This option is not available with the "M" option.

Z Suppress quitting on EOF, CTRL+Z, (end of file), while receiving data.

This option prevents the normal termination of the receiving process when an EOF, or CTRL+Z, CHAR(26) character is received from the remote. Using this option will require manual termination of receive mode by pressing a key to interrupt, then pressing "F" to finish.

Note: It is possible to send Pick data between two computers running PowerComm using the CP-SEND-ASCII and CP-RECV-ASCII commands. However, no control characters are masked, nor is any error checking performed. When sending from Pick to Pick, the "N" option should be used with both commands in order for the item ID's to be properly sent and received.

On R83, the "T" translation option can run slow depending upon the amount of carriage return and line feeds in the data. You may want to use pacing to control the speed in this case. When sending from Pick to Pick, you should omit the "T" option to eliminate an unnecessary step.

On all systems, ASCII decimal character 255 is translated to character 0.

See Also

- CP-SEND-ASCII for related information on page 116.
- The section, *Using CP-RECV-ASCII to Receive Data*, starting on page 43.

CP-RECV-PICK

Receives a file from the currently attached communications port from a Pick system without the full PowerComm product installed. This program relies on the use of a program called "REMOTE.SEND" on the remote computer.

CP-RECV-PICK fileref (options)

fileref Name of file on the local system to receive the data from the remote.

options Options may be any of the following:

A Automatic mode.

This option will prompt for the remote information locally, then send the information to the remote computer, and enter a view mode until the download starts.

The following information is prompted for locally:

Enter remote select statement or * for all: Enter an access statement or "*" for all items in the file. You may enter any valid TCL command that generates a list of items, including a get list.

Enter remote file name: Enter the name of the file to send.

Enter remote download options (H=hex): If your remote computer does not support 8 bit printing with Basic, you must enter "H" here. If it does not, the receive program will detect this, print an appropriate message and try again with the remote computer in hex mode.

Enter maximum number of seconds to wait for remote: This controls the amount of time to wait for the sending program to start sending data. It waits this amount of time for the "<autostart>" sequence of characters. If not received within the amount of time entered. The receive program will terminate with a timeout error.

D Enter direct mode after completion.

I Item ID suppress. †

n Maximum number of consecutive retries.

This numeric option specifies how many consecutive retries can occur before failing the transfer.

See Also

- CP-SEND-PICK for related information on page 116.
- The section, *Using CP-SEND-PICK and CP-RECV-PICK to Send and Receive Files*, starting on page 44.
- How to change the name of the remote server program using the *Global Parameters* screen, field "*Remote send program name*".

CP-RECV-SPOOL

Receives a spooler hold file from the currently attached communications port or the current port if using the "M" option.

CP-RECV-SPOOL (options)

options Options may be any of the following:

A Ask for spooler assignment.

This option will prompt for a spooler assignment.

B 7 bit binary mode. †

C Communications buffer display mode. †

F Through a file (used in conjunction with the "M" option). †

I Spooler entry number suppress. See item ID suppress. †

L Line display mode. †

M "Me" mode (not through a communications port). †

O Create one spooler hold file for all entries received.

The default is for a one-to-one correspondence of source entries to destination entries. This option overrides this default and creates one large entry containing all received spooler entries.

R Remove bytes at beginning of entry.

This option removes 1 byte for each “R” in the option string. See note following.

T Timeout mode (used in conjunction with "M" option). †

X Enable XON/XOFF flow control (used in conjunction with "M" option). †

Y Suppress “M” option spooler assignment change when using “ME” mode.

To prevent the ‘HOLD ENTRY #’ messages from corrupting data communications the CP-RECV-SPOOL command automatically invokes the SP-ASSIGN command with the “M” option to suppress these messages. You may override this behavior by using the “Y” option.

Note: Advanced Pick hold entries contain a segment mark (hex FF) at the beginning of a spooler hold entry. Because this is not documented it may be a bug on Advanced Pick. When sending a spooler entry from Advanced Pick to any other system, you should use the “R” option with the CP-RECV-SPOOL command to remove this extra character. When receiving spooler hold files from an R83 system, this option may be omitted.

See Also

- CP-SEND-SPOOL for related information starting on page 119.

CP-RECV-XMODEM

Receives a file from the remote computer using XModem protocol.

CP-RECV-YMODEM

Receives a file from the remote computer using YModem protocol.

CP-RECV-XMODEM fileref itemname (options)

CP-RECV-YMODEM fileref (options)

fileref Name of file to receive data.

itemname Item name. (XModem only)

options Options may be any of the following:

A Mask character mode.

Prompts for hex mask character that is imbedded within the data. This provides a way of sending segment marks, (hex "FF") as part of the X or Y Modem data. The mask character is followed by the 2 byte hex representation of the character you are masking.

B Block display mode.

Show communications packet blocks on screen. Use "BB" to show additional protocol information. Use "BBB" to show handshaking information.

C Request CRC check instead of simple checksum check.

Without the "C" option, X and Y Modem checksum packet check is used instead. With the "C" option, CRC check is requested from the sender but not forced if the sender does not support CRC check. Use "CC" to force CRC check and fail if sender does not support CRC check.

E Allow operator "escape" abort.

Use "EE" to prompt for the hex "escape" character to override the default of escape, CHAR(27), hex "1B".

G Request streaming protocol.

Use "GG" to force streaming protocol only. The packet check method in effect is determined by whether or not the "C" option is specified.

K Selects 1K XModem protocol. (XModem only)

File transfer progress is displayed in a box on the screen instead of scrolling down the screen. Use "KK" to first clear the screen before painting the box.

I Item ID suppress. †

M "Me" mode (not through a communications port). †

Using this option precludes the use of the following options; "I", "B", and "E". You must manually make sure the current port settings are compatible with full 256 character set, 8 bit data transfers. The "MM" option also turns off the break key. See the technical section regarding the "M" option.

N Parse for item ID's in the first attribute.

This option automatically uses the item ID in the first attribute and the item body follows until a null character, CHAR(0).

O Prompt for file size. (XModem only)

File size prompt is used to truncate saved data at specified size.

P Timing parameter override.

Prompts for new timing parameters. Each parameter is separated by a comma. You may omit override parameters by not specifying a value. (The defaults are configured in the *Global Parameters Definition* screen).

S Split records.

This option prompts for number of bytes maximum for each received item. When the number of bytes reaches the maximum a sequential number is appended to the original ID specified separated by "*". This option should be used when speed is important.

T Translation mode.

This option automatically translates carriage returns to attribute marks; CHAR(13) → CHAR(254). Line feeds, when found, are always deleted; CHAR(10) → NULL.

n Override default packet size with "n" bytes.

This option may only be used when using PowerComm to PowerComm X and Y Modem file transfers. Each side must use matching packet sizes. If the XModem protocol is used, it is reported as XModem-1K protocol in use.

^ Screen display mode.

File transfer progress is displayed in a box on the screen instead of scrolling down the screen. Use "^^" to first clear the screen before painting the box.

See Also

- The section, *XModem and YModem TCL Command Option Information*, starting on page 81.
- CP-SEND-XMODEM and CP-SEND-YMODEM for related information.

CP-SCRIPT

Executes a PowerComm script to the currently attached communications port.

CP-SCRIPT fileref itemlist (options)

fileref Name of script file.

itemlist Script name. May be a Item list or * for all items or no item list when using a pending selected list.

options Options may be any of the following:

D Enter direct mode when complete.

L Script line display mode.

This option displays the source script lines as they are executed which may be useful when debugging scripts.

M Monitor mode.

This option displays received data in both ASCII and hex modes. Which may be useful when debugging scripts.

See Also

- The appendix, *PowerComm's Script Language*, starting on page 150 for the script command language information.

CP-SEND

Sends a file to the currently attached communications port or the current port if using the "M" option.

CP-SEND fileref itemlist (options)

fileref Name of file to send.

itemlist Item list or * for all items or no item list when using a pending selected list.

options Options may be any of the following:

A Aborted send recovery option.

This option asks the receive program, CP-RECV, what the last item ID it saved was. CP-SEND then skips all item ID's in the item list and resumes sending data after the last ID saved. Successful use of this feature requires that the order of the item ID's in the send list do not change.

B 7 bit send mode. †

C Communications buffer display mode. †

D Delete source item.

This option deletes source items as it sends the data. It does not delete until the packet is acknowledged. In this mode, each item becomes a minimum of one packet, slowing overall transmission speeds.

F Through a file (used in conjunction with the "M" option). †

I Item ID suppress. †

L Line display mode. †

M "Me" mode (not through a communications port). †

O Override "D" pointer checking on "DICT" files.

This option overrides the checking of writing or overwriting of existing "D" pointers on "DICT" files. It also suppresses the checking of overlaying "CC" and "CL" items on R83. Using this option is not a good idea unless you are 100% sure of not creating items on the target system that Pick may think of as a file definition item or a BASIC object code or list pointer item. **THIS IS NOT A RECOMMENDED OPTION.**

R Send null items to receiver.

This option is used in conjunction with the CP-RECV "R" option to quickly remove items from the target receive file. If you specify this option and you do not use the CP-RECV "R" option, you will end up with NULL items in the receive file.

T Timeout mode (used in conjunction with "M" option). †

V Verify link version and maximum packet size of remote system.

This option asks the remote system its protocol version and maximum receive packet size. The packet size received from the remote will be used instead of the default packet size.

X Enable XON/XOFF flow control (used in conjunction with "M" option). †

n Packet size in bytes.

This option provides a way to override the default packet size at runtime. You may override the default packet size in the *Port Definition* screen.

Important Note: PowerComm employs a certain level of checking concerning writing or overwriting of "D" pointers to dictionary level files. However, this checking is only invoked if you use the word "DICT" as part of the received file name. If you omit the "DICT" option, PowerComm will not perform this check. Therefore, if you wish to send the MD, or the SYSTEM file and you would like PowerComm to filter out "D" pointers, specify "DICT MD" or "DICT SYSTEM". You may, at your own peril, override the checking option with DICT files by using the "O" option. (See the option description for further information and precautions).

Caution: On R83, the receive program is not smart enough to know the file is a "DC" file pointer and check only the "DC" files when checking for "CC" or "CL" items. In other words, all dictionary files are checked when using "DICT" unless the "O" option is used. In addition, if you have "CL" items in a data level file, and the receive file is a "DC" data level file, you may inadvertently create file corruption problems on the target system. PowerComm does not support sending of "CL" items.

See Also

- CP-RECV for related information on page 110.
- The section, *Using CP-SEND and CP-RECV to Send and Receive Files*, starting on page 38.
- The section, *Using the Aborted Send and Recover Options*, starting on page 41.
- The technical information section on *CP-SEND* starting on page 67.
- The section, *Precautions about Sending and Receiving Files*, starting on page 83.
- The section about changing the send and receive packet sizes in the *Port Definition* screen, field "*Default S/R packet size*", starting on page 129.

CP-SEND-ASCII

Sends a file to the remote computer in ASCII text format. No protocol or error checking is performed. The remote end should be in ASCII receive mode prior to commencing this command.

CP-SEND-ASCII fileref itemlist (options)

fileref Name of file to send.

itemlist Item list or * for all items or no item list when using a pending selected list.

options Options may be any of the following:

D Enter direct mode at completion.

I Item ID suppress. †

F Fast mode.

This mode sends the data as fast as possible without any translation, or pacing available. Each line or attribute of data is not sent separately. This option automatically turns off the "T" and the "P" options.

M "Me" mode (not through a communications port). †

Using this option precludes the use of the following options; "I", "D", and "V". See the "n" option for specifying a start delay.

N Include item ID name and a null character between items.

This option automatically adds the item ID before the item body. The item ID and the item body is separated by an attribute mark unless translation is on. The item body is followed by a null character, CHAR(0).

P Pace mode.

This option will prompt for three parameters separated by commas:

The pace character: The pace character must be in hex format. Typically this would be CR which is CHAR(13) or hex "0D". To skip pacing, leave this parameter blank.

Character transmission delay: This optional number controls how much time to delay between each character sent. The larger the number, the longer the delay. The delay you specify will depend a large part upon the system speed and activity at the time. You may need to experiment with this parameter for best results.

Line transmission delay: This optional number controls how long to delay in between lines of data. The larger the number, the longer the delay. The delay you specify will depend a large part upon the system speed and activity at the time. You may need to experiment with this parameter for best results. This parameter when used, specifies a delay that may be 100 times longer than the character transmission delay.

The "P" option is not available when using the "F" option.

T Translation mode.

This option automatically translates attribute marks to a carriage return; CHAR(254) → CHAR(13). Using "TT" in the options automatically translates from attribute marks to carriage return and line feed;
CHAR(254) → CHAR(13):CHAR(10).

This option is not available when using the "F" option.

V View mode.

This option turns on the view of transmitted data as it is received from the remote. If the remote system echo is off, you may not see much of anything. This option is not available with the "M" option.

Z Suppress sending CTRL+Z, (end of file), at the end of the file.

This option prevents the normal EOF, or CTRL+Z, CHAR(26) character from being the last character sent. Using this option may require manual termination of the remote computers ASCII receive function.

n Sleep time before starting send. (M)

This option specifies how many seconds to wait before sending the data.

Note: It is possible to send Pick data between two computers running PowerComm using the CP-SEND-ASCII and CP-RECV-ASCII commands. However, no control characters are masked, nor is any error checking performed. When sending from Pick to Pick, the "N" option should be used in order for the item ID's to be properly sent and received. We recommend using the CP-SEND and CP-RECV commands for complete error detection and correction.

See Also

- CP-RECV-ASCII for related information on page 111.
- The section, *Using CP-SEND-ASCII to Send Data*, starting on page 42.

CP-SEND-PICK

Sends a file to the remote computer using the Pick line editor as the receiving program. No protocol or error checking is performed.

CP-SEND-PICK fileref itemlist (options)

fileref Name of file to send.

itemlist Item list or * for all items or no item list when using a pending selected list.

options Options may be any of the following:

C Clean up transmitted data.

This option cleans up the transmitted attributes using the BASIC "MCP" conversion code. This conversion code converts all non-printable characters below space to a dot ".". This also has the potentially undesirable effect of converting value marks and sub value marks to a dot. This option is not available with the "F" option.

D Enter direct mode after completion.

F Fast mode.

This mode sends the data as fast as possible. Each line or attribute of data is not sent separately. This requires that the editor is able to receive one long continuous stream of binary data. This option only works on AP. This option precludes the use of the "C" and "U" options.

I Item ID suppress. †

N New item treatment.

This option eliminates a few editor commands speeding up the transfer of data. Each item on the remote computer is assumed to be new which eliminates the editor "DE32000" commands. If an item is not new, the data will be inserted before the old.

O Turn off echo on remote system.

This option can dramatically increase the speed as the comm port does not have to receive the echoed data as it is being sent. Note that it is possible to get the echo in the reverse state if you have broken out of a prior send and the echo was already off. In this case, do not specify the "O" option. Or turn on the echo in direct mode by entering "ECHO" at TCL on the remote, then exit and use the "O" option.

R Rename target file.

This option prompts for the name of the remote file name to edit when using the editor. By default, the same name is used as the source file.

U Prompt for the null string.

This option prompts for the string to use to represent null lines during upload. Normally the accent mark "`" is used. However, if the data you are sending contains a "`" on an attribute by itself, you may want to choose a different string to represent null by selecting this option.

V View mode.

This option turns on the view of transmitted data as it is received from the remote. If the remote system echo is off, you may only see the dots from the editor commands.

Z Show item size sent.

This option uses the editor "S?" command before saving the record so that you can see how many bytes the item contains before it is filed.

Note: Control characters in the data may affect how the editor behaves. It is best to only send data that does not have characters less than ASCII space. For example, the ESCAPE value is often translated to CHAR(251) on most Pick systems.

Only AP supports the use of the "F" option to receive a long continuous input command in the editor. It works quite well even with characters above 127 when the extended character set is on. Use the TCL command "XCS-ON" to turn on the extended character set.

On R83, using its shorter editor input length may turn one long attribute into several attributes.

See Also

- The section, *Using CP-SEND-PICK to Send Files*, starting on page 44.
- CP-RECV-PICK for related information starting on page 112.

CP-SEND-PORT

Sends special actions or characters to the currently attached comm port. After each keyword, when using the view option, a pause of 3 seconds occurs unless overridden with the "n" option.

CP-SEND-PORT keyword ... (options)

keyword Keywords are separated by spaces and may be any of the following:

"string" **'string'** **\string**	Sends the 'string' enclosed in the delimiters to the comm port.
BREAK, or B	Sends a break to the comm port.
CHAR:nnn	Sends the ASCII decimal character 'nnn' to the comm port.
END, or E	Sends the word "END" and ENTER to the comm port.
HEX:xx...	Sends the hex data 'xx' to the comm port. The 'xx' data may be as long as necessary and valid hex data and not contain the 'FF' character.
OFF, or O	Sends the word "OFF" and ENTER to the comm port.
PAUSE:n	Pauses for "n" seconds. If "n" is not specified, the pause lasts for 1 second.
RESET	On AP, executes the RESET-PORT command on the communications port.
VIEW:n	Views data on screen for "n" seconds. If "n" is not specified, the view lasts for 1 second.
XOFF	Sends the XOFF character, CTRL-S, to the comm port.
XON	Sends the XON character, CTRL-Q, to the comm port.

options Options may be any of the following:

D Enter direct mode after completion.

V View data.

This option views the data received from the comm port after the keyword action is performed.

n View data for "n" seconds.

This option with the "V" option displays the data from each keyword action for "n" seconds. The default, when not specified is 3 seconds.

Example:

```
CP-SEND-PORT BREAK END (V
```

This command sends a break, followed by "END" ENTER and views the data as it happens to the currently attached comm port. This command will last for about 6 seconds, 3 seconds for each keyword since the default was not changed.

Example:

```
CP-SEND-PORT XON "END" CHAR:13 PAUSE:1 HEX:57484F0D VIEW:5 (##
```

This command sends CTRL-Q, followed by the string "END" and a carriage return, which is character 13. Then pauses for 1 second, sends the hex data, which is the 3 letters "WHO" and ENTER, followed by a view period of 5 seconds. The "##" suppresses the normal display of error messages.

See Also

- CP-MSG for related information on page 106.

CP-SEND-SPOOL

Sends a spooler hold file to the currently attached communications port or the current port if using the "M" option.

CP-SEND-SPOOL entries (options)

entries Spooler hold entries to send separated by commas.

Entry numbers may be a combination of ranges and individual entries. Ranges are separated by dashes.

options Options may be any of the following:

B 7 bit send mode. †

C Communications buffer display mode. †

D Delete source spooler entry after successful transmission.

F Through a file (used in conjunction with the "M" option). †

I Spooler entry number suppress. See item ID suppress. †

L Line display mode. †

M "Me" mode (not through a communications port). †

T Timeout mode (used in conjunction with "M" option). †

X Enable XON/XOFF flow control (used in conjunction with "M" option). †

n Packet size in bytes.

This option provides a way to override the default packet size at runtime. You may override the default packet size in the *Port Definition* screen.

Example:

```
CP-SEND-SPOOL 1,4,5-10,15 (D
```

This command sends several spooler entries to the remote system and removes them as they are successfully sent.

See Also

- CP-RECV-SPOOL for related information starting on page 113.

CP-SEND-XMODEM

Sends a file to the remote computer using XModem protocol.

CP-SEND-YMODEM

Sends a file to the remote computer using YModem protocol.

CP-SEND-XMODEM fileref itemlist (options)

CP-SEND-YMODEM fileref itemlist (options)

fileref Name of file to send.

itemlist Item list or * for all items or no item list when using a pending selected list.

options Options may be any of the following:

A Mask character mode.

Prompts for hex mask character that is imbedded within the data. This provides a way of sending segment marks, (hex "FF") as part of the X or Y Modem data. The mask character is followed by the 2 byte hex representation of the character you are masking.

B Block display mode.

Show communications packet blocks on screen. Use "BB" to show additional protocol information. Use "BBB" to show handshaking information.

C Allow CRC check instead of simple checksum check.

Without the "C" option, X and Y Modem checksum packet check is used instead. With the "C" option, CRC check is allowed but not forced if the receiver does not support CRC check. Use "CC" to force CRC check and fail if receiver does not support CRC check.

D Create DOS compatible filenames. (YModem only)

This creates DOS filenames (8.3 format) where up to 8 valid characters are used from the original Pick item-ID and the last 3 characters (after the dot) are sequentially numbered from 001 to 999. This option must be used when sending to DOS based systems if the Pick item ID is not a valid DOS filename.

E Allow operator "escape" abort.

Use "EE" to prompt for the hex "escape" character to override the default of escape, CHAR(27), hex "1B".

G Allow streaming protocol.

Without the "G" option, X and Y Modem send requires acknowledgment of each packet sent. By using the "G" option, only the final packet is acknowledged. Use "GG" to force streaming protocol only. The packet check method in effect is determined by whether or not the "C" option is specified.

K Selects 1K XModem protocol. (XModem only)

File transfer progress is displayed in a box on the screen instead of scrolling down the screen. Use "KK" to first clear the screen before painting the box.

I Item ID suppress. †

M "Me" mode (not through a communications port). †

Using this option precludes the use of the following options; "I", "B", and "E". You must manually make sure the current port settings are compatible with full 256 character set, 8 bit data transfers. The "MM" option also turns off the break key. See the technical section regarding the "M" option.

N Include item ID name and a null character between items.

This option automatically adds the item ID before the item body. The item ID and the item body is separated by an attribute mark. The item body is followed by a null character, CHAR(0).

O One file mode.

Prompts for transfer file name and transfer file size.

XModem:

File name response is ignored. File size prompt is only used with screen display mode to show percentage complete.

YModem:

Instead of creating one-for-one YModem files on the receiving system, this option will send all data as one YModem file. Because the YModem protocol requires the file name and the file size, you must supply both of these parameters. In order to specify the file size parameter, you must know the total size of all items being sent. See the supplied sample program for help.

P Timing parameter override.

Prompts for new timing parameters. Each parameter is separated by a comma. You may omit override parameters by not specifying a value. (The defaults are configured in the *Global Parameters Definition* screen).

Q Ask for pad character.

Prompts for hex pad character that is used as filler at the end of a Xmodem or Ymodem file. The default pad character is CTRL+Z.

T Translation mode.

This option automatically translates attribute marks to a carriage return; CHAR(254) → CHAR(13). Using "TT" in the options automatically translates from attribute marks to carriage return and line feed;
CHAR(254) → CHAR(13):CHAR(10).

Z End of file options.

XModem:

This option prevents the normal EOF, or CTRL+Z, CHAR(26) character from being appending to the last file sent.

YModem:

This option prevents the final null block zero file header packet from being sent.

n Override default packet size with "n" bytes.

This option may only be used when using PowerComm to PowerComm X and Y Modem file transfers. Each side must use matching packet sizes. If the XModem protocol is used, it is reported as XModem-1K protocol in use.

^ Screen display mode.

File transfer progress is displayed in a box on the screen instead of scrolling down the screen. Use "^^" to first clear the screen before painting the box.

See Also

- The section, *XModem and YModem TCL Command Option Information*, starting on page 81.
- CP-RECV-XMODEM and CP-RECV-YMODEM for related information.

CP-START

Starts communications software on port specified. Unless the "X" option is used, flow control on the port will be disabled. This is to prevent garbage XON/XOFF characters from stopping data communications.

CP-START port, baudrate:throughput:parity:stopbits:wordlength, portname, isize, osize (options)

port Port number to start.

This is the port number that will be used for communications. It is the port usually connected to the modem or the port directly connected to another computer.

baudrate Baud rate of port:
(300,600,1200,2400,4800,9600,19200,38400)

throughput Optional baud rate of actual link:
(300,600,1200,2400,4800,9600,19200,38400)

This is used in situations where the transmission rate is running at a slower rate than the DTE rate to the modem. This parameter effects the computation of wait time for acknowledgment from the remote system. If you have a fixed DTE rate of 19200 but a connect rate of 2400, the timeout value would be insufficient for the actual amount of time it should take for data to arrive at the remote end.

parity Optional parity of port:
(NONE, MARK, SPACE, EVEN, ODD)

stopbits Optional number of stop bits of port:
(1, 2)

wordlength Optional data word length of port:
(7, 8)

portname Name of communications port (default is "P":port).

You may enter a non-numeric name up to 15 characters in length.

isize Size of input buffer in bytes.

(default is 20,000 bytes or as configured in the *Port Definition* screen)

osize Size of output buffer in bytes.

(default is 20,000 bytes or as configured in the *Port Definition* screen)

options Options may be any of the following:

A Attach after start.

This option will automatically attach the port after the start command completes.

B Select 7 bit binary mode.

This option controls the CP-SEND and CP-RECV commands binary option. When this option is used, it automatically forces the "B" option when using the CP-SEND and CP-RECV commands with this communications port.

D Enter direct mode after attach.

This option, used with the "A" option, will automatically enter the direct mode of communications after the port is attached.

L Log entry to ACC file.

On R83, this option creates an ACC file entry with "CP:" and the name of the port. This option on AP, uses the CHARGE-TO command to update the PIBS file.

X Enable XON/XOFF flow control.

This option turns on XON/XOFF flow control for the port. Normally it is turned off to avoid port lockups when the XOFF character is received from noisy or unreliable connections.

Note: R83 does not support the highest baud rate of 38400 nor does it support SPACE or MARK parity.

See Also

- CP-KILL for related information on page 104.
- The section, *Using CP-START to Start a Communications Port*, starting on page 31.
- The technical information section on *CP-START* starting on page 73.

CP-STAT-FILE

Computes length and checksum on each item specified in the itemlist. This includes the item ID, attribute mark, and the data item. The size of the checksum may be specified in the options from 8 to 32 bits. The default is 16 bits, however specifying a 32 bit value will create a more unique checksum value.

This command is supplied to provide a platform independent way of determining the length and checksum of items on whichever platform PowerComm is installed on.

CP-STAT-FILE fileref itemlist (options)

fileref Name of file to send.

itemlist Item list or * for all items or no item list when using a pending selected list.

options Options may be any of the following:

A Ask for mask character.

Reduces the total item length by 2 for each mask character found. This may used when using the CP-SEND-XMODEM and CP-SEND-YMODEM commands to compute the total file size when using the "O" option.

D Compute length and checksum on data item only.

Normally, this command computes the length and checksum of the ID, an attribute mark, and the data item. Using the "D" option computes the item length and checksum on the data item only. This may be used to check items that have different item ID's.

F Fast mode.

This option only computes the length of each item skipping the checksum step.

H Hex mode.

The checksum is displayed in hex. This also causes the final checksum total to be displayed in hex.

I Item ID suppress. †

L Line display mode. †

n Specifies size of checksum in bits.

This option specifies the size of the checksum that is computed. Valid options are "8", "16", or "32". Unless specified, an 16 bit checksum is generated.

Example:

```
CP-STAT-FILE BP * (H,16,$
```

This will compute statistics on all items in the BP file and display the 16 bit checksums in hex mode. In addition a final display of minimum, average and maximum items sizes will be displayed using the "$" option.

Example:

```
CP-STAT-FILE TRANS * (A,D,F,I) {07}
```

This will compute statistics on all items in the TRANS file and reduce the computed lengths by the mask character (hex character "07") if found.

Note: Due to the maximum number size on R83 systems, it is possible when using the larger bit values of 16 or 32, to overflow when totalling the final checksum for all of the items in the file. However, individual items will not overflow. If you care about the final checksum on R83 systems, you may want to use only the 8 bit option. If the final checksum overflows, it will be shown with a decimal in the number.

See Also

- The section, *Using CP-STAT-FILE to Check Sent or Received Files*, starting on page 50.
- The technical information section on *CP-STAT-FILE* starting on page 74.

CP-STATUS-PORT

Displays port status information including the current communications parameters, the output from the where TCL command, the PIB status information, and the CP-LIST command.

This command does not need to be run on a PowerComm communications port. It can be run on any port you want to get information on. By default, all 4 of the following blocks of information for the port specified are shown unless explicitly suppressed with the appropriate options:

- the current communications port parameters including baud rate, parity, stop bits, word length, type-ahead status, flow control status, extended character set status, and data carrier detect status.
- the output from the WHERE TCL command.
- the interpretation of the PIB status word from the WHERE command.
- the output from the CP-LIST command.

CP-STATUS-PORT portnum ... (options)

portnum Port number(s) to display separated by spaces.

options Options may be any of the following:

C Suppress the current communications parameters block.

L Suppress the CP-LIST output block.

O Override the R83 suppression of using 'SET-PORT' with this command. (See caution below)

P Suppress the PIB status interpretation block.

W Suppress the WHERE output block.`

Example:

```
CP-STATUS-PORT 5
```

This displays all of the information available for port 5.

See Also

- The technical information section on *CP-STATUS-PORT* starting on page 75.

Caution: The Pick command "SET-PORT" is used to set and return a port's communications parameters including the baud rate, parity, word length and stop bits. However, on R83 the SET-PORT command can intermittently crash the system when a port is sending or receiving data at the same time. This SET-PORT command is used in the PowerComm CP-STATUS-PORT command to show the current port communications parameters. Because of the potential for the SET-PORT command to crash the system, we have removed its use in the CP-STATUS-PORT command unless you override with the "O" option. If you use the "O" option, at your own risk, you should only do so when the port is not sending or receiving data. Otherwise you may crash the system and create GFE's.

CP-VERSION

Displays version and license information of PowerComm.

CP-VERSION (options)

options Options may be any of the following:

D Show full details.

This option, shows full version information including any updates that have been applied.

W Show "WHICH" information.

This option executes the Pick TCL command "WHICH".

CP-VIEW-TCL

Displays changed output from continuous TCL command(s) execution.

This displays the difference from execution to execution of multiple TCL commands by highlighting only the changes on the screen.

CP-VIEW-TCL tcl.statement1 tcl.statement2 ... (options)

tcl.statement TCL statements separated by spaces.

If the TCL statement requires options, the TCL statement must be surrounded by any of the following delimiters: `" ' \`

Stacked data may follow the TCL command separated by "|".

Multiple TCL statements must be separated by spaces and enclosed in quotes (if the command has embedded spaces) and the "M" option must be specified.

options Options may be any of the following:

M Multiple TCL statements are present.

Without the "M" option, one TCL command is assumed which may contain spaces without having to enclose the command in delimiters. When multiple TCL statements are present, each TCL statement must be enclosed with any of the following delimiters: `" ' \`

P Suppress default pause for duration of 1 timeslice.

n Pause for "n" number of seconds between command executions.

Examples:

```
CP-VIEW-TCL WHERE 10
```

```
CP-VIEW-TCL "WHERE 10 (Z"
```

The first command will execute the TCL command "WHERE 10". No quotes are needed because the command is not using options. The second requires quotes to prevent the TCL commands options being interpreted as CP-VIEW-TCL options.

```
CP-VIEW-TCL "WHAT (S" POVF (M,1
```

This command will execute the TCL command "WHAT (S", followed by "POVF". Since there are two TCL commands, each command must be enclosed in delimiters (if spaces) and the "M" option specified. There will be a pause for 1 second in between executions.

Caution: This command can be very CPU intensive on a system. Therefore to avoid system degradation, the "n" option should be used to pause in between screen displays.

See Also

- The technical information section on *CP-VIEW-TCL* starting on page 77.

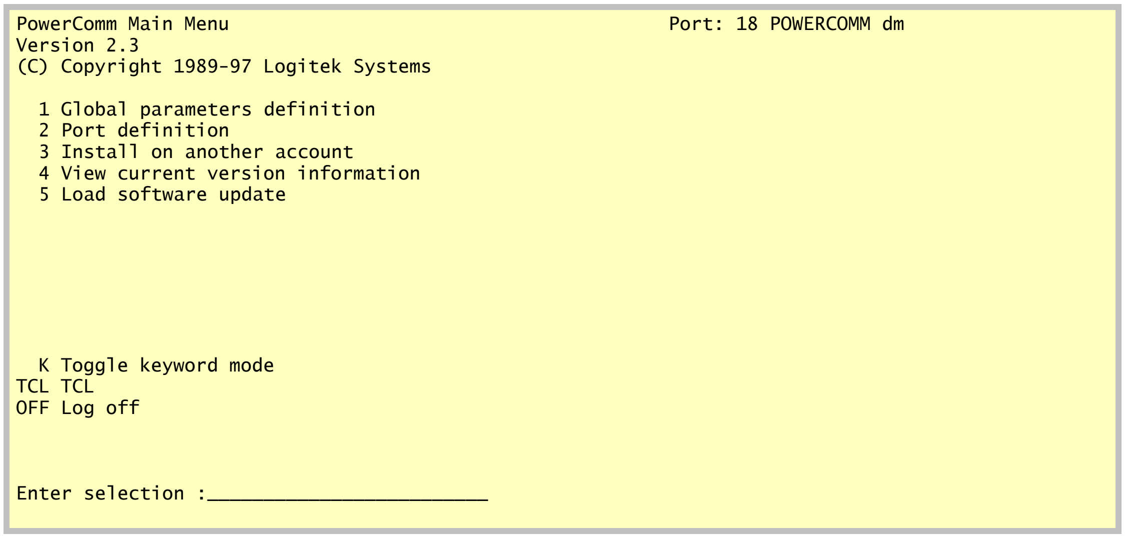
CHAPTER 9
Menu Selections Reference

This chapter documents the menus and entry screens in PowerComm.

Refer to the software operation conventions on page 16 for details on how to operate the menus and entry screens.

PowerComm Main Menu

The PowerComm main menu shows all of the processes available for setting up and maintaining communication parameters, and installing the software on other user accounts. The main menu is accessed by the CP-MENU command from TCL.



```
PowerComm Main Menu                                       Port: 18 POWERCOMM dm
Version 2.3
(C) Copyright 1989-97 Logitek Systems

  1 Global parameters definition
  2 Port definition
  3 Install on another account
  4 View current version information
  5 Load software update








  K Toggle keyword mode
TCL TCL
OFF Log off



Enter selection :_________________________
```

The PowerComm Main Menu

Enter selection: Enter the menu selection desired:

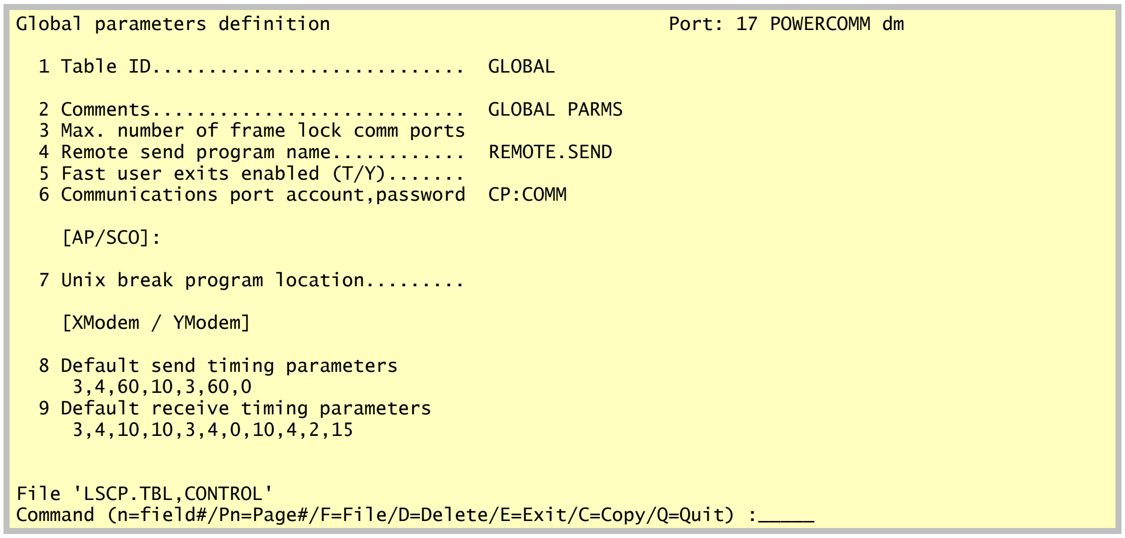
1-5	Selects option 1 through 5 on the menu
K	Toggles the keyword display mode
TCL	Exit to TCL
OFF	Log off

Global Parameters Definition

The *Global Parameters Definition* screen is used to define the parameters that are used throughout the communications software.

Global Parameters Definition

The *Global Parameters Definition* screen is used to define the parameters that are used throughout the communications software.



```
Global parameters definition                          Port: 17 POWERCOMM dm

  1 Table ID...........................  GLOBAL

  2 Comments...........................  GLOBAL PARMS
  3 Max. number of frame lock comm ports
  4 Remote send program name...........  REMOTE.SEND
  5 Fast user exits enabled (T/Y).......
  6 Communications port account,password  CP:COMM

    [AP/SCO]:

  7 Unix break program location.........

    [XModem / YModem]

  8 Default send timing parameters
     3,4,60,10,3,60,0
  9 Default receive timing parameters
     3,4,10,10,3,4,0,10,4,2,15


File 'LSCP.TBL,CONTROL'
Command (n=field#/Pn=Page#/F=File/D=Delete/E=Exit/C=Copy/Q=Quit) :_____
```

Use CP-MENU GPD from TCL to go directly to this menu selection

Table ID: This field contains the name of the table item "GLOBAL". This is a display only field, and cannot be changed.

Comments: This field may contain any text you want to enter.

Max number of frame lock comm ports: This field contains the number of simultaneous communications ports that may core lock their input and output (I/O) buffers. This field is provided because on some AP/Native and AP/DOS platforms it is possible to have very limited amounts of memory left over after the operating system and drivers are loaded.

If too little memory is available it is possible to lock up the system if you start several communications ports with the size of buffers exceeding the available non-locked memory space available.

Because R83 systems generally have more free memory space available, this field is not used on R83. All R83 communications port I/O buffers are core locked. Therefore, do not start multiple ports with large I/O buffers to avoid a system lockup problem.

Remote send program name: This field contains the name of the program to execute when using the "A" option with the CP-RECV-PICK TCL command.

Fast user exits enabled (T/Y): This field may contain "Y" for yes if your system supports the fast user exits option. Currently this is only available on AP 6.1.x systems. Refer to your Pick release documentation for details.

Using this option will speed up the runtime execution of user exits at the expense of a small startup delay.

The proper way to use this field is to select "T" first, exit to TCL, run CP-MENU. If it works, it will show this on the screen and automatically change the value from "T" to "Y" and stop. Subsequently the "Y" option will be in effect.

If this is set to "Y" and the feature is not available it may abort. If this occurs you may manually clear the "Y" in the following location:

```
FILE:      LSCP.TBL,CONTROL
ITEM:      GLOBAL
ATTRIBUTE: 13
```

Communications port account,password: This field contains the logon account name and optional password for running the communications port. This field is required for Advanced Pick Operating Systems. It is not required for other versions.

CP-DIR break key setup parameter for Unix systems

Unix break program location: This field contains the name of the UNIX program to shell out from Pick in order to send a break to a TTY device for Unix hosted systems. Details about this field are available starting on page 25.

XModem / YModem

Default send timing parameters: This field contains the default timing parameters for the CP-SEND-XMODEM and CP-SEND-YMODEM commands.

XModem and YModem send timing parameters

#	Default	Explanation
1	3	Number of CTRL-X characters to receive from remote before aborting file transfer.
2	4	Number of CTRL-X characters to transmit to remote when local operator aborts file transfer.
3	60	Number of seconds to wait for the remote to acknowledge a packet before aborting the file transfer.
4	10	Maximum number of successive errors before aborting file transfer.
5	3	Number of seconds to wait for all parameter 1 characters to arrive indicating a real abort sequence.
6	60	Number of seconds to wait for protocol handshake to complete before aborting.
7	0	Number of "beeps" when using screen display mode at the end of a transfer.

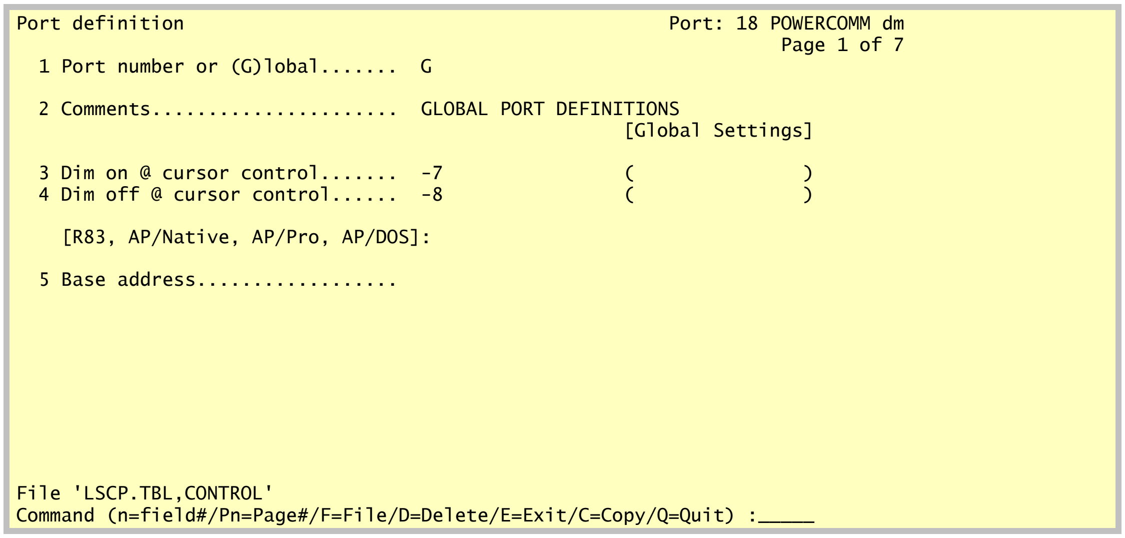
Default receive timing parameters : This field contains the default timing parameters for the CP-RECV-XMODEM and CP-RECV-YMODEM commands.

XModem and YModem receive timing parameters

#	Default	Explanation
1	3	Number of CTRL-X characters to receive from remote before aborting file transfer.
2	4	Number of CTRL-X characters to transmit to remote when local operator aborts file transfer.
3	10	Number of seconds to wait for the remote to begin sending a data packet when not in handshake mode.
4	10	Maximum number of successive errors before aborting file transfer.
5	3	Number of seconds to wait for all parameter 1 characters to arrive indicating a real abort sequence.
6	4	Number of seconds to wait during the handshake phase before sending the next handshake character.
7	0	Number of "beeps" when using screen display mode at the end of a transfer.
8	10	Number of seconds to wait for remaining characters in data packet mode.
9	4	Number of "C"s to send during CRC handshake mode.
10	2	Number of "G"s to send during "stream" handshake mode.
11	15	Maximum number of seconds to wait for input buffer to drain before sending "NAK" to the sender.

Port Definition - Page 1 of 7

The *Port Definition* screen is used to define each port on the system that will be used for communications work. This definition allows you to define the global parameters for all ports by using "G". You may then setup individual ports that can override the global defaults by entering a value for each field you wish to override.



```
Port definition                                            Port: 18 POWERCOMM dm
                                                                     Page 1 of 7
  1 Port number or (G)lobal.......  G

  2 Comments......................  GLOBAL PORT DEFINITIONS
                                                       [Global Settings]

  3 Dim on @ cursor control.......  -7                 (               )
  4 Dim off @ cursor control......  -8                 (               )

    [R83, AP/Native, AP/Pro, AP/DOS]:

  5 Base address..................









File 'LSCP.TBL,CONTROL'
Command (n=field#/Pn=Page#/F=File/D=Delete/E=Exit/C=Copy/Q=Quit) :_____
```

Use CP-MENU PD from TCL to go directly to this menu selection

Cancelling the Global Default

There are 7 places in the *Port Definition* screen where it may be desirable to setup a global default, but for some ports you do not want the global default. The following are the 7 fields where you may enter a backslash "\" to prevent the global default from being used:

Dim on @ cursor control:
Dim off @ cursor control:
Before start port program:
Before start port parameters:
Start port options:
After kill port program:
After kill port parameters:

Port number or (G)lobal: Enter "G" to maintain the global settings for all ports, or enter a port number to define.

Comments: This field contains any comments about this definition.

Dim on @ cursor control number: This field contains the negative number used by Pick to turn on the half intensity mode for the menu displays. If you do not wish to utilize half intensity, leave this field blank.

Dim off @ cursor control number: This field contains the negative number used by Pick to turn off the half intensity mode for the menu displays. If you do not wish to utilize half intensity, leave this field blank.

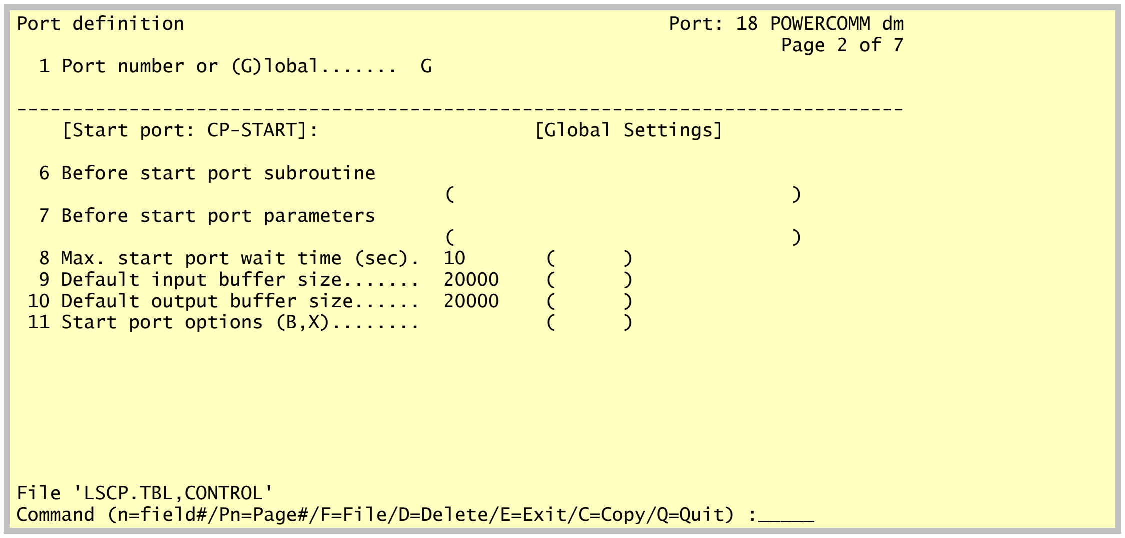
CP-DIR break key setup parameter for non-Unix systems

Base address: This field contains the port's base address in hex notation (for AP/Native, AP/Pro, AP/DOS and R83 versions only). This is used for sending a break during direct mode.

Refer to the *Break Key Setup* section for further information about this field starting on page 25.

Port Definition - Page 2 of 7

This page of the port definition contains fields relevant for starting a port.



```
Port definition                                          Port: 18 POWERCOMM dm
                                                                   Page 2 of 7
  1 Port number or (G)lobal.......  G

-------------------------------------------------------------------------------
    [Start port: CP-START]:                  [Global Settings]

  6 Before start port subroutine
                                      (                              )
  7 Before start port parameters
                                      (                              )
  8 Max. start port wait time (sec).  10       (      )
  9 Default input buffer size.......  20000    (      )
 10 Default output buffer size......  20000    (      )
 11 Start port options (B,X)........           (      )







File 'LSCP.TBL,CONTROL'
Command (n=field#/Pn=Page#/F=File/D=Delete/E=Exit/C=Copy/Q=Quit) :_____
```

CP-START setup parameters

Before start port program: This field contains the name of the program (subroutine) to run before the port is started. Refer to the section, *Using the Before Start and After Kill Port Program Parameters*, starting on page 73. This capability is useful for systems where the port needs to be configured beyond baud rate and flow control which is handled by the CP-START command. On Unix hosted systems it may be desirable to start the Pick process on that TTY line before starting the port. (See also the field named, *After kill port program*, later in this screen description.) The before start port parameters must contain a value.

Before start port parameters: This field contains parameters to pass to the start port program. Each parameter is separated by a space. The subroutine is **only** called if this field contains a value.

Max. start port wait time: This field contains the number of seconds to wait before giving up on the start command.

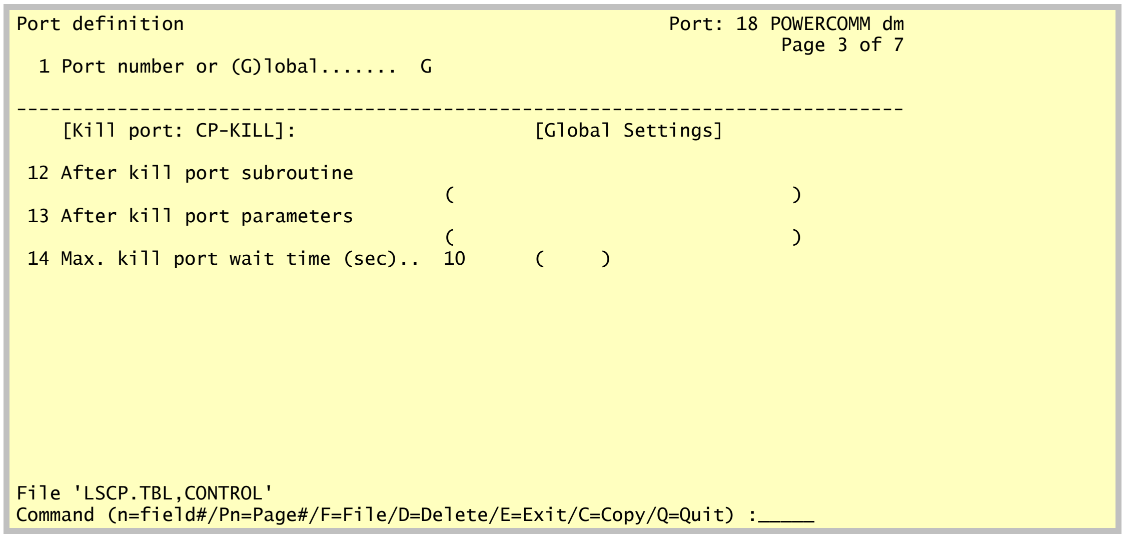
Default input buffer size: This field contains how many bytes to allocate to the communications port input buffer if not specified in the CP-START command.

Default output buffer size: This field contains how many bytes to allocate to the communications port output buffer if not specified in the CP-START command.

Start port options (B,X): This field may contain the CP-START option letters "B" and/or "X". The "B" and "X" option also effect how the port behaves in "ME" mode. In both a communications port and "ME" mode, the "B" option enables 7 bit transfers and the "X" option enables XON/XOFF flow control.

Port Definition - Page 3 of 7

This page of the port definition contains fields relevant for killing a port.



```
Port definition                                           Port: 18 POWERCOMM dm
                                                                    Page 3 of 7
  1 Port number or (G)lobal.......  G

-------------------------------------------------------------------------------
    [Kill port: CP-KILL]:                     [Global Settings]

 12 After kill port subroutine
                                      (                              )
 13 After kill port parameters
                                      (                              )
 14 Max. kill port wait time (sec)..  10      (     )









File 'LSCP.TBL,CONTROL'
Command (n=field#/Pn=Page#/F=File/D=Delete/E=Exit/C=Copy/Q=Quit) :_____
```

CP-KILL setup parameters

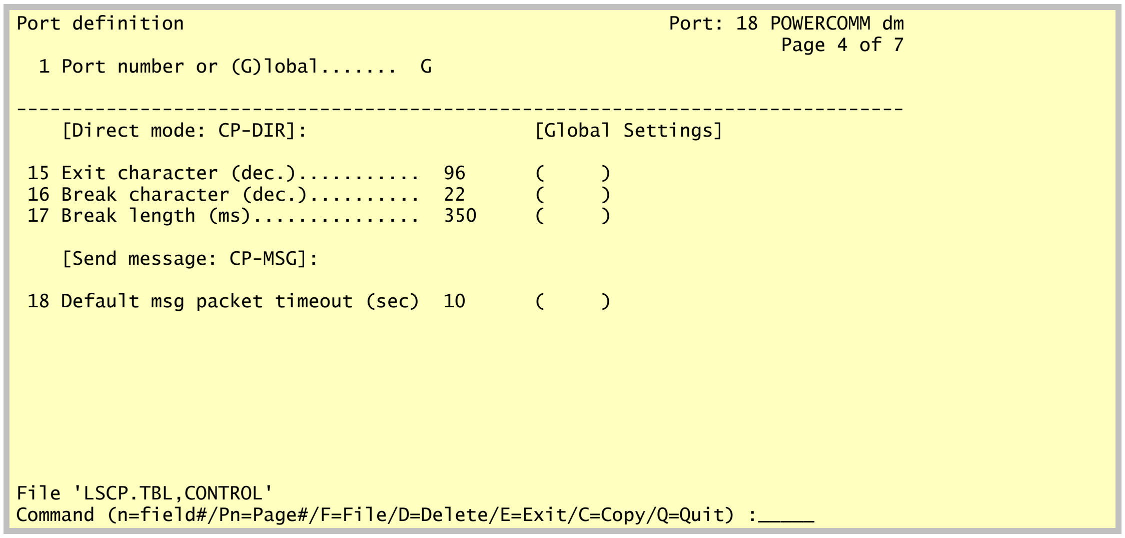
After kill port program: This field contains the name of the program (subroutine) to run after the port is killed. Refer to the section, *Using the Before Start and After Kill Port Program Parameters*, starting on page 73. This capability is useful for systems where the port needs to be re-configured after use. On Unix hosted systems it may be desirable to kill the Pick when complete. (See also the field named, *Before start port program*, earlier in this screen description.) The before start port parameters must contain a value.

After kill port parameters: This field contains parameters to pass to the kill port program. Each parameter is separated by a space. The subroutine is **only** called if this field contains a value.

Max. kill port wait time: This field contains the number of seconds to wait before giving up on the kill command and reporting an error.

Port Definition - Page 4 of 7

This page of the port definition contains fields relevant for using direct mode with CP-DIR and sending messages with the CP-MSG commands.



```
Port definition                                           Port: 18 POWERCOMM dm
                                                                    Page 4 of 7
  1 Port number or (G)lobal.......  G

-------------------------------------------------------------------------------
    [Direct mode: CP-DIR]:                    [Global Settings]

 15 Exit character (dec.)..........  96      (     )
 16 Break character (dec.)..........  22      (     )
 17 Break length (ms)...............  350     (     )

    [Send message: CP-MSG]:

 18 Default msg packet timeout (sec)  10      (     )









File 'LSCP.TBL,CONTROL'
Command (n=field#/Pn=Page#/F=File/D=Delete/E=Exit/C=Copy/Q=Quit) :_____
```

CP-DIR setup parameters

Exit character (decimal): This field contains the decimal number representing the ASCII character to be used to exit out of direct mode when using CP-DIR.

Break character (decimal): This field contains the decimal number representing the ASCII character to be used to initiate a break key sequence in direct mode when using CP-DIR. This feature is used where an alternate break key is desired. Note that the character is not actually transmitted.

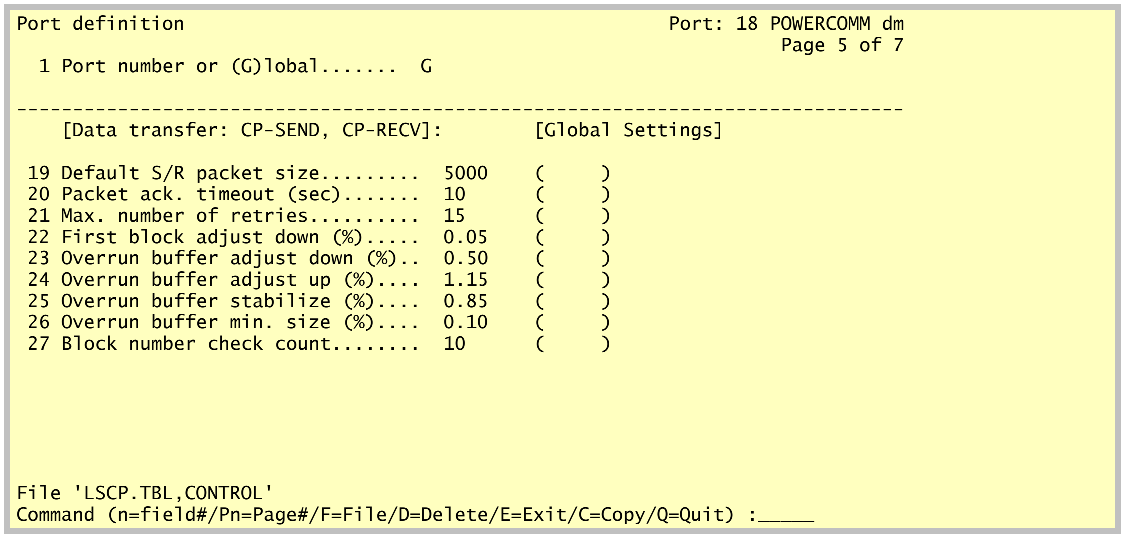
Break length (milliseconds): This field contains the number of milliseconds to send out a break signal to the communications port.

CP-MSG setup parameters

Default message packet timeout: This field contains the default number of seconds to wait for a matching response when using the CP-MSG command.

Port Definition - Page 5 of 7

This page of the port definition contains fields relevant for sending and receiving data using the CP-SEND and CP-RECV commands..



```
Port definition                                           Port: 18 POWERCOMM dm
                                                                    Page 5 of 7
  1 Port number or (G)lobal.......  G

-------------------------------------------------------------------------------
    [Data transfer: CP-SEND, CP-RECV]:        [Global Settings]

 19 Default S/R packet size.........  5000    (     )
 20 Packet ack. timeout (sec).......  10      (     )
 21 Max. number of retries..........  15      (     )
 22 First block adjust down (%).....  0.05    (     )
 23 Overrun buffer adjust down (%)..  0.50    (     )
 24 Overrun buffer adjust up (%)....  1.15    (     )
 25 Overrun buffer stabilize (%)....  0.85    (     )
 26 Overrun buffer min. size (%)....  0.10    (     )
 27 Block number check count........  10      (     )







File 'LSCP.TBL,CONTROL'
Command (n=field#/Pn=Page#/F=File/D=Delete/E=Exit/C=Copy/Q=Quit) :_____
```

CP-SEND, CP-RECV setup parameters

For more details on how these fields effect the protocol, refer to page 71 in the *Technical Information* chapter for more information.

Default S/R packet size: This field contains the default transmission packet size that is used with the CP-SEND TCL command. On the remote system it also contains the maximum number of characters that can be received. You can obtain the receiver's maximum packet size parameter by using the "V" option with the TCL CP-SEND command.

Packet ack. timeout: This field contains the number of seconds of inactivity during communications transfers before a synchronization sequence is issued.

Max. number of retries: This field contains the maximum number of retries when moving data using the CP-SEND and CP-RECV commands before giving up.

First block adjust down: This field contains a number that is multiplied by the packet size. This number is then used as the packet size of the first block transmitted. This feature helps prevent a buffer overrun on the first packet to the remote system.

Overrun buffer adjust down: This field contains a number that is multiplied by the current packet size when a buffer overrun condition occurs. The buffer overrun occurs when a Pick system's input buffer fills up and it transmits a bell character in response.

Overrun buffer adjust up: This field contains a number that is multiplied by the current packet size after a block overrun occurs and the next packet is successful. This feature is used to slowly increase the packet size up until the buffer overrun condition occurs again.

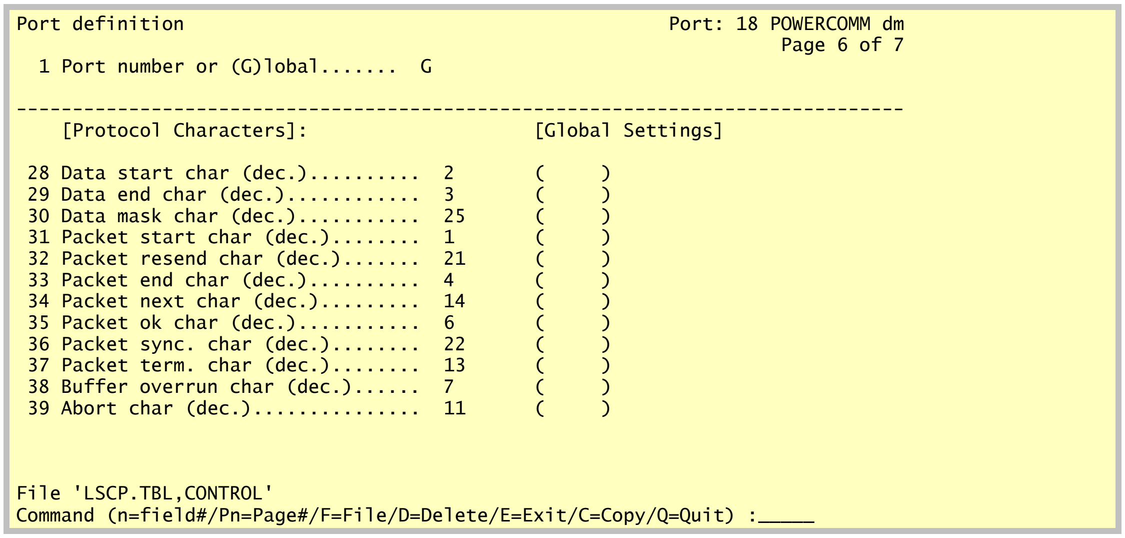
Overrun buffer stabilize: This field contains a number that is multiplied by the current packet size after the packet size has increased after a buffer overrun condition. This number is used to make the packet size slightly less then the last packet size that caused the remote system to exceed its input response capabilities.

Overrun buffer minimum size: This field contains a number that is multiplied by the maximum packet size. This number is the minimum size the packet is allowed to reach. When it reaches this size or less, the transfer will be aborted.

Block number check count: This field contains the number to divide the packet block number as part of the packet protocol. It is provided for backwards compatibility with older versions where this field was not adjustable. This field must match the remote computer you are communicating with. There is no other advantage in changing this number to a lower setting.

Port Definition - Page 6 of 7

This page of the port definition allows you to modify the packet protocol characters in the event you are communicating to a system that will not pass certain characters through to Pick. This may occur, for instance, when the break key for Pick has been assigned to use one of these characters.



```
Port definition                                          Port: 18 POWERCOMM dm
                                                                   Page 6 of 7
  1 Port number or (G)lobal.......  G

-------------------------------------------------------------------------------
    [Protocol Characters]:                  [Global Settings]

 28 Data start char (dec.)..........  2       (     )
 29 Data end char (dec.)............  3       (     )
 30 Data mask char (dec.)...........  25      (     )
 31 Packet start char (dec.)........  1       (     )
 32 Packet resend char (dec.).......  21      (     )
 33 Packet end char (dec.)..........  4       (     )
 34 Packet next char (dec.).........  14      (     )
 35 Packet ok char (dec.)...........  6       (     )
 36 Packet sync. char (dec.)........  22      (     )
 37 Packet term. char (dec.)........  13      (     )
 38 Buffer overrun char (dec.)......  7       (     )
 39 Abort char (dec.)...............  11      (     )



File 'LSCP.TBL,CONTROL'
Command (n=field#/Pn=Page#/F=File/D=Delete/E=Exit/C=Copy/Q=Quit) :_____
```

Protocol parameters

Each of the protocol characters defined in this section and must be unique. Refer to page 70 in the *Technical Information* chapter for more information about protocol characters.

Data start char (dec.): This field contains the decimal ASCII character for the data start portion of the packet.

Data end char (dec.): This field contains the decimal ASCII character for the data end portion of the packet.

Data mask char (dec.): This field contains the decimal ASCII character for the data portion of the packet. This character is used to mask certain characters in the data stream from causing problems with the protocol.

Packet start char (dec.): This field contains the decimal ASCII character for the start portion of the packet.

Packet re-send char (dec.): This field contains the decimal ASCII character to transmit to the other system when the packet was incomplete or invalid.

Packet end char (dec.): This field contains the decimal ASCII character for the end portion of the packet.

Packet next char (dec.): This field contains the decimal ASCII character to transmit to the other system when it is requesting the next packet of data.

Packet ok char (dec.): This field contains the decimal ASCII character to transmit to the other system when the packet received was accepted.

Packet sync. char (dec.): This field contains the decimal ASCII character to transmit to the other system when there has been no activity for a period of time.

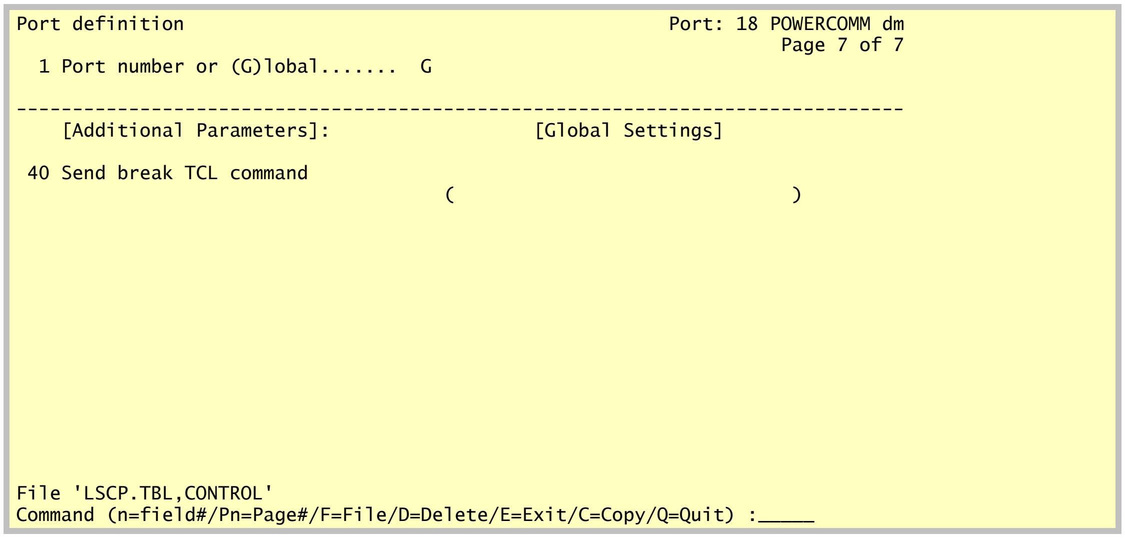
Packet term. char (dec.): This field contains the decimal ASCII character to transmit to the other system after every kind of packet.

Buffer overrun char (dec.): This field contains the decimal ASCII character of the character that is transmitted by the other system when its type-ahead input buffer is filled up.

Abort char (dec.): This field contains the decimal ASCII character to transmit to the other system to abort the current file transfer. This is also used by the CP-ABORT command.

Port Definition - Page 7 of 7

This page of the port definition allows contains additional parameters.



```
Port definition                                           Port: 18 POWERCOMM dm
                                                                    Page 7 of 7
  1 Port number or (G)lobal.......  G

-------------------------------------------------------------------------------
    [Additional Parameters]:                  [Global Settings]

 40 Send break TCL command
                                      (                              )












File 'LSCP.TBL,CONTROL'
Command (n=field#/Pn=Page#/F=File/D=Delete/E=Exit/C=Copy/Q=Quit) :_____
```

Additional parameters

Send break TCL command: This field contains the optional TCL command to execute in order to send a break out the communications port. If the field "Base address" (on screen page 1 of 7) contains a value, it will override this field. You may use "%" in the command syntax to specify the current port number so that it may be made a global definition with the "G" item. Example for recent versions of AP/Pro: SEND-BREAK %

Install On Another Account

This menu selection installs the PowerComm software on another account that you choose. This enables the other account to access all of the PowerComm software utilities and data files.

Enter account name(s) to install or '*' for all (separated by spaces): Enter the account names you want to install PowerComm on separated by spaces or "*" for all accounts on the system.

View Current Version Information

The *View Current Version Information* selection is provided to view details about the current PowerComm version and update status.

Load Software Update

The *Load Software Update* selection is provided to load PowerComm software updates should they become available.

No users or processes should be running any PowerComm routines when using this selection. In addition, the media device must be attached before running this menu selection. Follow the written instructions included with the update and prompts as they appear on the screen.<bm end>

PART 4

Programmer Reference

CHAPTER 10
Using PowerComm in Applications

This chapter describes how to use PowerComm in your application programs.

Note: The new PowerComm Communications Library (PCL) includes a comprehensive set of programs with source code for automating a variety of communications tasks.

Introduction

The following BASIC programs are provided in order to access some of PowerComm features. This is a brief summary of the included programs documented in more detail below.

The following subroutines are used to interface to the communications software.

Program Name	**Brief Description**
CP.BASIC.GET.HANDLE.SUB	Used when interfacing directly to the user modes in order to get the handle of the communications port currently running. See page 142 for additional information.
CP.BASIC.GET.MODES.SUB	Used when interfacing directly to the user modes provided with the software. See page 142 for additional information.

The following programs are examples and are located in the LSCP,BP.EXAMPLES file.

Program Name	**Brief Description**
CP.AP.UNIX.UTY.SUB	Included as an example BASIC program to perform additional functions when starting and killing a communications port. See page 74 for additional information.
CP.TCL.EXAMPLE	Included as an example BASIC program interfacing to the TCL commands. See page 138 for additional information.
CP.UM.EXAMPLE	Included as an example BASIC program interfacing with the user modes. See page 139 for additional information.
CP.TCL.UM.EXAMPLE	Included as an example BASIC program that interfaces to the TCL commands and the user mode interface.

Note: Some of the subroutines require that you use BASIC includes in order to access the parameters of some data structures and files. They are shown where needed.

Using the TCL Command Interface

The TCL commands may be using in procs or BASIC programs. Most all of the TCL commands return an error message number that can be checked within a proc or BASIC program. If you want to ensure the success of a TCL command, it is a good idea to check these error messages. A complete listing of all the error messages is in the appendix, *Error Messages*, starting on page 159.

Note: Some PowerComm TCL commands return multiple error messages.

Using TCL Commands in Procs

Since all of the PowerComm TCL commands are procs, special care must be taken when using the TCL commands from a proc. The following proc is an example of how to call several of the TCL commands from a proc.

Example:

```
PQ
C PROC TO START AND ENTER DIRECT MODE
C
RI
IHCP-START 15,19200
[MD
IF E # CP.008 IF E # CP.011 XNot started
C
RI
IHCP-ATT 15
[MD
IF E # CP.020 XNot attached
C
RI
IHCP-DIR
[MD
XTCL
```

You cannot call a proc from another proc using the "H" and "P" commands and still return to the proc that called it. Using the technique in the example gets around the limitation. However, it does so at the expense of using the entire primary input buffer. Therefore, you have no proc variables available.

A better way is to use a BASIC program.

Using TCL Commands In BASIC Programs

The following simple program is equivalent to the preceding proc. Since it is written in BASIC, you have more error control, program flow logic and variables available. In addition, it is not as cryptic and therefore easier to read and maintain.

Example:

```
* program to start and enter direct mode
*
TCL = "CP-START 15,19200"
EXECUTE TCL RETURNING ERRORS
*
IF INDEX(ERRORS, "CP.008", 1) OR INDEX(ERRORS, "CP.011", 1) ELSE
  CRT "Not started   "
  STOP
END
*
TCL = "CP-ATT 15"
EXECUTE TCL RETURNING ERRORS
*
IF INDEX(ERRORS, "CP.020", 1) ELSE
  CRT "Not attached  "
  STOP
END
*
TCL = "CP-DIR"
EXECUTE TCL RETURNING ERRORS
*
STOP
```

In BASIC, error messages from a TCL command can be obtained with the RETURNING clause on the EXECUTE command as well as the SYSTEM(17) function. The RETURNING clause separates error messages by spaces marks and SYSTEM(17) function separates error messages by attribute marks.

Keep in mind that several TCL commands return multiple error messages.

TCL Command Interface Example

The following program demonstrates the use of the TCL commands in a BASIC program. This program asks the user for a port to use, starts the port, enters direct mode, then kills the port when the user exits direct mode.

The following program CP.TCL.EXAMPLE is located in the LSCP,BP.EXAMPLES file.

Program listing

```
CP.TCL.EXAMPLE
* Example communications program
* Starts port, goes into direct mode and kills port
*
* +---------------------------------------------------------------------+
* | (C) Copyright 1991 Logitek Systems, San Marcos, CA USA             |
* |                                                                     |
* | This unpublished work is the property of and embodies trade secrets |
* | and confidential information proprietary to Logitek Systems.        |
* | adapted, or modified without the express written approval of        |
* | Logitek Systems.  ALL RIGHTS RESERVED.                              |
* +---------------------------------------------------------------------+
*
* TDB 09-10-91 1.00  Implemented
*
* constants
*
EQU AM                              TO CHAR(254)
EQU VM                              TO CHAR(253)
EQU SVM                             TO CHAR(252)
EQU SP                              TO " "
EQU NUL                             TO ""
EQU MAX                             TO 32767
*
EQU C.CR                            TO CHAR(13)
*
*************************************************************************
*
PROMPT NUL
*
CRT "Enter communications port# to use: ":
INPUT CommPort
CommPort = CommPort "CN"
IF CommPort = NUL THEN STOP
```

The following code executes the TCL command CP-START and checks for its success by testing the error message number returned from the execute statement.

```
*
* start port
*
TCL = "CP-START ":CommPort:",,DIR-":CommPort:",5000,5000"
EXECUTE TCL RETURNING Errors
IF Errors = "CP.008" OR Errors = "CP.011" ELSE STOP
```

The following code attaches the port using CP-ATT and checks for its success.

```
*
* attach port
*
TCL = "CP-ATT ":CommPort
EXECUTE TCL RETURNING Errors
IF Errors = "CP.020" ELSE STOP
*
CRT "Press exit character to exit direct mode."
```

The following code runs the CP-DIR command.

```
*
* enter direct mode
*
TCL = "CP-DIR"
EXECUTE TCL RETURNING Errors
```

Then the port is killed using the CP-KILL command.

```
*
* kill port
*
```

```
CRT "Killing communications port..."
TCL = "CP-KILL ":CommPort
EXECUTE TCL
*
STOP
*
```

Using the User Modes Interface

The User Mode Interface can be used when you want complete control over input and output of a communications port. These routines can be used to send data and receive data from the currently attached communications port. In order to use the User Modes, you must load the names of the user modes into an equated array using the following lines of code:

DIM LSCP.MODES(200)
INCLUDE LSCP,EQU LSCP.MODES
CALL CP.BASIC.GET.MODES.SUB(MAT LSCP.MODES, bError)

This allows you to reference the user mode by name instead of an address.

Most user modes that interface to the communications port routines require a handle. This handle may be obtained for the currently attached communications port with the following code:

CALL CP.BASIC.GET.HANDLE.SUB(MAT LSCP.MODES, Handle, PortNo, bError, Err, PrintFlag)

This handle is used to reference the appropriate table entries in the assembly code.

For more details about using the User Mode API, refer to the chapter, *User Mode API Reference,* starting on page 141. This chapter also contains an alphabetical listing of the available user modes.

User Mode Interface Example

The following program demonstrates the use of the User Mode API. This program retrieves and displays on the screen the US Robotics configuration profile.

The following program CP.UM.EXAMPLE is located in the LSCP,BP.EXAMPLES file.

Program listing

```
CP.UM.EXAMPLE
* Example communications program
* Returns configuration profile on US ROBOTICS modem
*
* +---------------------------------------------------------------------+
* | (C) Copyright 1991 Logitek Systems, San Marcos, CA USA             |
* |                                                                     |
* | This unpublished work is the property of and embodies trade secrets |
* | and confidential information proprietary to Logitek Systems.       |
* | adapted, or modified without the express written approval of       |
* | Logitek Systems.  ALL RIGHTS RESERVED.                              |
* +---------------------------------------------------------------------+
*
* constants
*
EQU AM                               TO CHAR(254)
EQU VM                               TO CHAR(253)
EQU SVM                              TO CHAR(252)
EQU SP                               TO " "
EQU NUL                              TO ""
EQU MAX                              TO 32767
*
EQU C.CR                             TO CHAR(13)
*
```

The following line must be included in order to reference the user mode definitions

```
INCLUDE LSCP,EQU LSCP.MODES
*
*************************************************************************
*
PROMPT NUL
```

The following lines call a program to load the user mode definitions for the system type you are running on.

```
*
* get user modes
*
```

```
DIM LSCP.MODES(200)
CALL CP.BASIC.GET.MODES.SUB(MAT LSCP.MODES, bError)
IF bError THEN
  CRT "Modes not defined for this system   "
  ABORT
END
```

The following lines call a program to retrieve the handle of the currently attached communications port running.

```
*
* get handle of comm port
*
CALL CP.BASIC.GET.HANDLE.SUB(MAT LSCP.MODES, Handle, PortNo, bError, Err, "1")
IF bError THEN STOP
*
*************************************************************************
```

The following line clears the current input buffer for the communications port.

```
*
* reset all input on port
*
z = OCONV(Handle, uResetInputPort)
```

The following lines send the command to the modem.

```
*
* send inquiry command
*
String = "ATE1Q0I7":C.CR
z = OCONV(Handle:String, uWritePort)
```

The following lines wait for input for up to 2 seconds.

```
*
* get data until 2 second timeout occurs or CHAR(254) received
*
String = OCONV(Handle:AM:"2", uReadPortCharTimeout)
*
* display on screen
*
CRT String
*
STOP
*
```

CHAPTER 11
User Mode API Reference

This chapter documents the programs for accessing the user mode interface capabilities of PowerComm.

Note: The new PowerComm Communications Library (PCL) includes a comprehensive set of programs with source code for automating a variety of communications tasks.

Introduction

User modes may be used when you want to have complete control over the I/O on the communications port. This chapter documents all of the user modes available and their interfaces.

Note: When using the user exits, it is not possible to send or receive the hex "FF" character. Upon receipt of an "FF" it is converted to "00". A future release may address this limitation.

User Modes by Category

These are the PowerComm User Mode Routines organized by type of function. Use this list as a quick index to the user modes reference section following. Page numbers are indicated after the short descriptions.

Communications Port I/O Routines

The following routines are used to manage the communications port. There are routines for reading data from the port and writing to the port. In addition, there are several routines for resetting and counting the number of characters in the communications port buffers.

uCountInputPort	Used to count how many characters are currently in the comm ports input buffer, 145
uCountOutputPort	This user mode is used to count how many characters are currently in the comm ports output buffer, 141
uReadPortCharTimeout	Used to get characters from comm port up to 'char' is found or timeout occurs, 147
uReadPort	Used to get all available input characters from a comm port, 146
uReadPortCount	Used to get up to 'count' number of characters from comm port, 147
uReadPortChar	Used to get input from comm port up to 'char' found in input stream, 147
uResetInputPort	Used to reset comm port input buffer to null, 147
uWritePort	This user mode is used to send output to a comm port, 148

Local Port I/O Routines

The following routines are for accessing the local port and not the communications port. These are for inputting and outputting to the local port in raw mode.

uInputAll	Used to input characters from keyboard until no characters remaining in input buffer, 145
uInputChar7	Used to get one character from keyboard with no 8th bit, 145
uInputLen	Used to input "n" characters from keyboard, 146
uInputToChar	Used to input characters from keyboard until 'char' is pressed, 146
uInputToCharTimeout	Used to input characters from keyboard until 'char' is pressed or timeout occurs, 146
uOutput	Used to print all characters in string to screen without interpretation by BASIC PRINT or CRT statements, 146

Miscellaneous Routines

The following routines are used in the communications software to perform a few tasks that would be slow or unavailable in Pick Basic.

uBitWise	Used to perform bitwise operations on 2 numbers or characters, 143
uConvBits	Used to convert data from hex to bits (one's & zero's) and from bits back to hex, 144
uConvMask	Used to mask undesirable control characters from a string for communications transfers, 144
uGetPort	Used to get current port's port number, 145

User Mode Interface Routines

The following BASIC subroutines are required in order to use any of the User Mode routines provided with PowerComm.

All PowerComm subroutines use the default precision and have **not** been compiled using any BASIC PRECISION statements.

CP.BASIC.GET.MODES.SUB

CP.BASIC.GET.MODES.SUB is a BASIC subroutine that may be called from your application program as follows:

CALL CP.BASIC.GET.MODES.SUB(MAT LSCP.MODES, bError)

This subroutine is used to load the user mode definitions for your system type. This routine only needs to be used if you plan on calling any of the user modes.

The following lines of code must proceed the call to CP.BASIC.GET. MODES.SUB:

DIM LSCP.MODES(200)
INCLUDE LSCP,EQU LSCP.MODES

Parameters Passed	
(none)	

Parameters Returned	
MAT LSCP.MODES	Required. Used for returning the array of user mode definitions that the equated names relate to.
bError	Set to "1" if an error occurred and the user modes are not defined.

CP.BASIC.GET.HANDLE.SUB

CP.BASIC.GET.HANDLE.SUB is a BASIC subroutine that may be called from your application program as follows:

CALL CP.BASIC.GET.HANDLE.SUB(MAT LSCP.MODES, Handle, PortNo, bError, Err, PrintFlag)

This subroutine returns the handle of the communications port currently attached.

This is used when using the user modes to interface to the communications port. It returns the handle of the communications port that is currently attached. Many of the user modes require this handle to access the correct communications port table entry. You must call the program CP.BASIC.GET.MODES.SUB before calling this routine and before using any of the user modes.

To get the handle of another communications port, use the TCL CP-ATT command in an execute statement, then call this subroutine to again to get the handle of the new current attached communications port.

The following lines of code must proceed the call to CP.BASIC.GET.HANDLE.SUB:

DIM LSCP.MODES(200)
INCLUDE LSCP,EQU LSCP.MODES
CALL CP.BASIC.GET.MODES.SUB(MAT LSCP.MODES, bError)

Parameters Passed	
MAT LSCP.MODES	Required. Used in the subroutine.
PrintFlag	Set to "1" to print the error messages on the screen.

Parameters Returned	
Handle	One byte character representing the entry number in the communications port table.
PortNo	Your port number, not the communications port.
bError	Set to "1" if an error occurred.
Err	Error message numbers and parameters. Each error is separated by attribute marks and parameters by sub-value marks.

User Mode Routines (In Alphabetical Order)

uBitWise

uBitWise is a user exit that may be called from your BASIC program as follows:

result = OCONV("I" : op : n1 : "." : n2, uBitWise)

Used to perform bitwise operations on two numbers.

2 different modes and 3 bitwise operations are supported:

The first form using "I" performs the operations on 2 integer numbers and returns the results as a integer number.

The following user mode parameters are required:

"I" indicates "I"nteger mode

op the operation to perform:

O Or the 2 numbers
A And the 2 numbers
X Exclusive Or the 2 numbers

n1 the first number
n2 the second number

Each number, n1 and n2, must be an integer in the range 0 through 255.

The following parameters are returned:

result an integer between 0 and 255

Examples:

OCONV("IO127.3", uBitWise) = 127
OCONV("IA127.3", uBitWise) = 3
OCONV("IX127.3", uBitWise) = 124

The second form using "C" performs the operations on 2 characters and returns the results as a character. This form is slightly faster than the first form because several internal conversions are avoided.

result = OCONV("C" : op : c1 : c2, uBitWise)

The following user mode parameters are required:

"C" indicates "C"haracter mode

op the operation to perform:

O Or the 2 numbers
A And the 2 numbers
X Exclusive Or the 2 numbers

c1 the first character
c2 the second character

Character 'c1' must be an ASCII character between 0 and 254. Character 'c2' must be an ASCII character between 0 and 254. Character 'c2' may be NULL to represent CHAR(255).

The following parameters are returned:

result a character with an ASCII value between 0 and 254. A null character will be returned if the result was ASCII 255.

Examples:

C1 = CHAR(127) ; C2 = CHAR(3)
SEQ(OCONV("CO":C1:C2, uBitWise)) = 127
SEQ(OCONV("CA":C1:C2, uBitWise)) = 3
SEQ(OCONV("CX":C1:C2, uBitWise)) = 124

uConvBits

uConvBits is a user exit that may be called from your BASIC program as follows:

result = OCONV(hex, uConvBits)

Used to convert data from hex to bits (one's & zero's) and from bits back to hex.

The first form using OCONV, converts from hex to binary.

The following user mode parameters are required:

hex hex data to convert

The following parameters are returned:

result string of 1's and 0's from hex data:
"F4" = "11110100"

result = ICONV(bits, uConvBits)

The second form using ICONV, converts from binary to hex.

The following user mode parameters are required:

bits string of 1's and 0's to convert

The following parameters are returned:

result hex data from bit data:
"11110100" = "F4"

uConvMask

uConvMask is a user exit that may be called from your BASIC program as follows:

results = OCONV(mask.char : bit.options : string, uConvMask)

Used to mask undesirable control characters from a string for communications transfers.

The first form using OCONV, converts from raw data to masked format.

The following user mode parameters are required:

mask.char character to flag characters masked

bit.options character:
if CHAR(00):
  converts to 7 bit ASCII
  masks chars < 32 (space)
  and > 126
if CHAR(01):
  1 = converts to 8 bit ASCII
  masks chars < 32 (space)
  and char(127)

The following parameters are returned:

results converted string

results = ICONV(mask.char : bit.options : string, uConvMask)

The second form using ICONV, converts from masked format to raw data format.

The following user mode parameters are required:

mask.char character to designate masked characters

bit.options character: (must be a single character but not used)

The following parameters are returned:

results converted string

uCountInputPort

uCountInputPort is a user exit that may be called from your BASIC program as follows:

count = OCONV(handle, uCountInputPort)

Used to count how many characters are currently in the comm ports input buffer.

The following user mode parameters are required:

handle 1 byte handle

The following parameters are returned:

count number of bytes in input buffer

uCountOutputPort

uCountOutputPort is a user exit that may be called from your BASIC program as follows:

count = OCONV(handle, uCountOutputPort)

This user mode is used to count how many characters are currently in the comm ports output buffer.

The following user mode parameters are required:

handle 1 byte handle.

The following parameters are returned:

count Number of bytes in output buffer.

uGetPort

uGetPort is a user exit that may be called from your BASIC program as follows:

portno = OCONV("", uGetPort)

Used to get current port's port number.

The following user mode parameters are required:

"" null

The following parameters are returned:

portno this port number

uInputAll

uInputAll is a user exit that may be called from your BASIC program as follows:

string = OCONV("", uInputAll)

Used to input characters from keyboard until no characters remaining in input buffer.

The following user mode parameters are required:

"" null

The following parameters are returned:

string data input

uInputChar7

uInputChar7 is a user exit that may be called from your BASIC program as follows:

char = OCONV("", uInputChar7)

Used to get one character from keyboard with no 8th bit.

The following user mode parameters are required:

"" null

The following parameters are returned:

char single character input

uInputLen

uInputLen is a user exit that may be called from your BASIC program as follows:

string = OCONV(count, uInputLen)

Used to input "n" characters from keyboard.

The following user mode parameters are required:

count	number of characters to input

The following parameters are returned:

string	characters entered

uInputToChar

uInputToChar is a user exit that may be called from your BASIC program as follows:

string = OCONV(char, uInputToChar)

Used to input characters from keyboard until 'char' is pressed.

The following user mode parameters are required:

char	character to terminate input

The following parameters are returned:

string	characters entered (including 'char')

uInputToCharTimeout

uInputToCharTimeout is a user exit that may be called from your BASIC program as follows:

string = OCONV(char : timeout, uInputToCharTimeout)

Used to input characters from keyboard until 'char' is pressed or timeout occurs.

The following user mode parameters are required:

char	character to terminate input
timeout	timeout value in seconds

The following parameters are returned:

string	characters entered (including 'char')

uOutput

uOutput is a user exit that may be called from your BASIC program as follows:

results = OCONV(string, uOutput)

Used to print all characters in string to screen without interpretation by BASIC PRINT or CRT statements.

The following user mode parameters are required:

string	string to output

The following parameters are returned:

results	null

uReadPort

uReadPort is a user exit that may be called from your BASIC program as follows:

string = OCONV(handle, uReadPort)

Used to get all available input characters from a comm port.

The following user mode parameters are required:

handle	1 byte handle

The following parameters are returned:

string	data from comm port

uReadPortChar

uReadPortChar is a user exit that may be called from your BASIC program as follows:

string = OCONV(handle : char, uReadPortChar)

Used to get input from comm port up to 'char' found in input stream. This will not wait for the character.

The following user mode parameters are required:

handle	1 byte handle
char	character to terminate input

The following parameters are returned:

string	data from comm port, including 'char' if found

uReadPortCharTimeout

uReadPortCharTimeout is a user exit that may be called from your BASIC program as follows:

string = OCONV(handle : char : timeout, uReadPortCharTimeout)

Used to get characters from comm port up to 'char' is found or timeout occurs.

The following user mode parameters are required:

handle	1 byte handle
char	character to terminate input
timeout	timeout value in seconds

The following parameters are returned:

string	data from comm port (including 'char')

uReadPortCount

uReadPortCount is a user exit that may be called from your BASIC program as follows:

string = OCONV(handle : count, uReadPortCount)

Used to get up to 'count' number of characters from comm port. If no characters present, returns immediately.

The following user mode parameters are required:

handle	1 byte handle
count	maximum number of characters to return

The following parameters are returned:

string	data from comm port

uResetInputPort

uResetInputPort is a user exit that may be called from your BASIC program as follows:

z = OCONV(handle, uResetInputPort)

Used to reset comm port input buffer to null.

The following user mode parameters are required:

handle	1 byte handle

The following parameters are returned:

z	null

uWritePort

uWritePort is a user exit that may be called from your BASIC program as follows:

z = OCONV(handle : string, uWritePort

This user mode is used to send output to a comm port.

The following user mode parameters are required:

handle	1 byte handle.
string	string to output to comm port

The following parameters are returned:

z	null

Appendices

APPENDIX A
PowerComm's Script Language

This appendix describes PowerComm's script language. Scripts are used with the CP-SCRIPT command. Each line of the script file is a separate command or comment. Comment lines are designated by "!" or "*" as the first non-space character in a line.

Refer to page 113 for more information on CP-SCRIPT.

Note: The new PowerComm Communications Library (PCL) includes a more comprehensive script language including source code that you can expand upon.

Script Command Format

Script command lines are in the following format:

```
command parameters ...
```

- Script commands may be upper or lower case.
- Parameters may be numeric, or strings enclosed in double (") or single (') quotes.

Runtime Considerations

During the execution of a script, the user may press a key with the following prompt displayed:

```
Script interrupted by keystroke!
Select action (C=continue, A=abort, S=skip command, D=direct mode):
```

The following actions are available:

C	Continue with the script
A	Abort the script
S	Skip the current command in the script
D	Enter direct mode

Script Commands

Script commands are listed below in alphabetical order.

PAUSE n

Pauses for *n* seconds.

Example:

```
PAUSE 5
```

RECVCLEAR

Clears the input buffer of any data. Useful for subsequent *waitfor* commands to insure the *waitfor* commands are in sync with the *transmit* commands.

STOP

Stops execution of this script.

TCL "command" "stack1" "stack2" ...

Execute the TCL command *command* passing stacked data *stack1*, *stack2*, etc.

Example:

```
TCL "CP-DIR"
```

Example:

```
TCL "COPY BP *" "(BP.UPLOAD"
```

TRANSMIT "string"

Transmits the string in quotes. Strings may contain special characters as described on page 17.

Example:

```
TRANSMIT "TERM W^M"
```

WAITFOR "string" n

Waits for the remote system to send the string in quotes up to *n* seconds before terminating. If *n* is omitted, 10 minutes is the maximum wait time. Strings may contain special characters as described on page 17.

Example:

```
WAITFOR "LOGON PLEASE:" 10
```

WAITQUIET n m

Waits up to *m* seconds for the received input data to be inactive for *n* seconds. If *m* is omitted, 10 minutes is the maximum wait time.

Example:

```
WAITQUIET 10 120
```

Script Example

This simple script calls up a system, logs on, downloads a file from SYSPROG and hangs up.

```
*
* call client system, logon to sysprog download file and hang up
*
TRANSMIT "ATE1Q^M"
PAUSE 2
TRANSMIT "ATDT555-1111>^M"
WAITFOR "CONNECT"
PAUSE 1
TRANSMIT "  ^M"
PAUSE 1
TRANSMIT "SYSPROG^M"
PAUSE 5
TRANSMIT "CP-SEND SYSPROG-BP FORMAT (M^M"
PAUSE 3
TCL "CP-RECV BP"
PAUSE 1
TRANSMIT "OFF^M"
PAUSE 5
TRANSMIT "!!!"
PAUSE 2
TRANSMIT "ATZ^M"
PAUSE 2
STOP
```

APPENDIX B
TCL Command Message Control Table

The following table of TCL command messages is provided so that you may use additional options with the TCL commands to suppress or add information to the execution of these commands.

The "#" and "$" options are cumulative. If an action is suppressed by "##" then the actions suppressed by one "#" is also suppressed. Likewise, if additional information is obtained with "$$" then all information with a single "$" is also shown. These options may be combined to suppress normal information and show additional information at the same time.

- Using "#" multiple times *suppresses* more and more information than is normally shown with using a command.
- Using "$" multiple times *displays* more and more information than is normally shown with using a command.

Column	Description
Command	This column shows the command name in boldface followed by a list of possible messages or functions. If a function is preceded by an option in parentheses "()", then that message or function is applicable only when the indicated option is used with the command.
Suppressed	This column will suppress the associated message or function by using the number of "#" characters indicated. A special case of 5 "#" will suppress all PowerComm error messages that are in the format of "[CP.nnn] message".
Extra	This column will add the associated messages or behavior to the command execution by using the number of "$" characters indicated.

The "#" and "$" options may be combined with other options and used with any PowerComm TCL command as shown in the following examples with optional commas shown for clarity:

Examples:

```
CP-SEND BP * ($
CP-DIR (##,$$
CP-ABORT (#
CP-MENU ($$$
CP-MSG (##
```

TCL Command Message Control Table

Command	Suppressed	Extra
CP-ABORT		
(C) <n characters lost>	#	
CP-DIR / CP-CAPTURE		
capture file opened	# #	
capture file created	# #	
direct mode started	# #	
no break key option	# #	
(A) character entered was ...	#	
escape key is ...		$
break key is ...		$
(C) <n characters lost>	#	
(Q) <control Q sent>	#	
(R) <reset port n>	#	
(R) <reset port not available on this platform>	#	
<break> (when break pressed)		$ $
<n characters lost> (when break pressed)	#	
<no break option configured> (when break pressed and no break key)	#	
direct mode stopped	# #	
disconnect / detach by other user	# # #	
capture file closed	# #	
CP-LIST		
ports listed messages	# #	
comm table frame is		$
CP-MENU		
executing TCL command ...		$ $
CP-MSG		
message sent	# #	
message timeout	# #	
message mismatch	# #	
CP-PACKAGE		
(S) Suppress "*"'s	#	
View individual commands		$ $
CP-PLAYBACK		
playback of capture file completed	# #	
capture item(s) deleted	# #	
capture stats	#	
CP-RECV-ASCII		
ASCII download started	# #	
ASCII download completed	# #	
ASCII receive is now starting...	# # #	
ASCII download aborted	# #	
ASCII download stats	#	
ASCII download progress dots		$

TCL Command Message Control Table

Command	Suppressed	Extra
CP-RECV-PICK		
ASCII download started	# #	
ASCII download completed	# #	
ASCII download aborted	# #	
ASCII download stats	#	
Remote file =		$
Hex packet detected		$
Retries <R>	#	
ASCII download progress dots		$
CP-RECV-XMODEM / CP-RECV-YMODEM		
result code/comm protocol message	##	
error stats message	#	
<nak> message	#	
CP-SCRIPT		
script not on file	# # #	
script started	# #	
disconnect / detach by other user	# # #	
script completed	# #	
script aborted	# #	
script syntax error ...	# # # #	
CP-SEND / CP-RECV		
communications statistics; overruns, timeouts, syncs, retries		$
(A) Last filed ID =	# #	
(V) remote version =	# #	
(V) remote max packet size =	# #	
suppress communications errors; <R>, <B>, <S>, %	# # #	
view communications error details		$ $ $
view communications error details and block size adjustments details		$ $ $ $
CP-SEND-ASCII		
ASCII upload started	# #	
ASCII upload completed	# #	
ASCII send will start in "n" seconds	# # #	
ASCII upload aborted	# #	
ASCII upload stats	#	
ASCII upload progress dots		$
CP-SEND-PICK		
ASCII upload started	# #	
ASCII upload completed	# #	
ASCII send will start in "n" seconds	# # #	
ASCII upload aborted	# #	
ASCII upload stats	#	
ASCII upload progress dots		$
CP-SEND-PORT		
break sent	# #	
no break option available	# #	
message ... sent	# #	
<break sent>		$ $

TCL Command Message Control Table

Command	Suppressed	Extra
CP-SEND-XMODEM / CP-SEND-YMODEM		
result code/comm protocol message	##	
error stats message	#	
<nak> message	#	
CP-START		
Baud rate, parity, bits, word length, buffer sizes		$
Runs CP-LIST port#		$ $
Runs CP-LIST port# (C,T		$ $ $
Runs CP-STATUS-PORT port# ($		$ $ $ $
CP-STAT-FILE		
display min, average, max statistics		$
suppress statistics message at end	# #	
CP-STATUS-PORT		
suppress some column headings	#	
suppress all column headings	# #	
for CP-LIST, use the "C,T" options		$
CP-VIEW-TCL		
suppress messages when changing display characteristics	#	
suppress command counter	# #	
show delay countdown		$

APPENDIX C
PowerComm Specifications

Advanced Pick Systems

- Up to 16 simultaneous sessions
- Baud rates to 38400
- User-definable ABS file usage during installation

Pick R83 Systems

- Up to 15 simultaneous sessions
- 32,000 bytes maximum item size
- Baud rates to 19200
- User-definable ABS frame usage during installation

APPENDIX D
Removing PowerComm

The following PowerComm removal instructions are by system type. Refer to the appropriate section for your system type.

Refer to *Terminology* starting on page 15 if you are unfamiliar with any terms used in this section.

PowerComm for Advanced Pick Removal Instructions

▸ **Follow these steps to remove PowerComm from the system:**

1. Logon to the DM account if you are not logged on to DM already.
2. Make sure no communications ports are running.

   Use the TCL command "CP-LIST" to check.

   ```
   CP-LIST
   ```

3. Release used buffer space back to overflow.

   Use the PowerComm TCL command "CP-CLEAR" to release all available buffer frames to overflow.

   ```
   CP-CLEAR (A,R,U
   ```

4. Remove the ABS modes from the ABS file. (**Optional Step**)

   This step is optional and utilizes a currently undocumented feature of Advanced Pick's MLOAD command and may not work on all systems. You may only need to do this if you are loading other modes and you need the space in the ABS area.

   First type in "SSELECT LSCP,MODES", then type in "MLOAD LSCP,MODES (U)" and when prompted for the ABS file name, enter the ABS file name you originally loaded the ABS modes into.

   ```
   SSELECT LSCP,MODES
   MLOAD LSCP,MODES (U
   ```

5. Remove the account from the system.

   Use the TCL command "DELETE-ACCOUNT acctname" substituting "acctname" for the name of the account you installed PowerComm to earlier.

   ```
   DELETE-ACCOUNT POWERCOMM
   ```

PowerComm for R83 Removal Instructions

▸ **Follow these steps to remove PowerComm from the system:**

1. Logon to the SYSPROG account if you are not logged on to SYSPROG already.
2. Make sure no communications ports are running.

   Use the TCL command "CP-LIST" to check.

   ```
   CP-LIST
   ```

3. Release used buffer space back to overflow.

   Use the PowerComm TCL command "CP-CLEAR" to release all available buffer frames to overflow.

   ```
   CP-CLEAR (A,R,U
   ```

4. Remove the account from the system.

   Use the TCL command "DELETE-ACCOUNT acctname" substituting "acctname" for the name of the account you installed PowerComm to earlier.

   ```
   DELETE-ACCOUNT POWERCOMM
   ```

APPENDIX E
Error Messages

The following is a numerical listing of all of the error messages in PowerComm.

[CP.002] Maximum of A retries exceeded!

[CP.003] Communication ports are still active!

[CP.004] Communication ports not cleared!

[CP.005] Communication ports cleared!

[CP.006] A communications ports listed.

[CP.007] No communications ports running!

[CP.008] Communications port number A started, named B.

[CP.009] Communications port number A not started!

[CP.010] Communications port number A is not valid!

[CP.011] Communications port number A is already running!

[CP.012] Communications port A not available!

[CP.013] Communications port number A reports no disk space!

[CP.014] Communications port number A reports no available entries!

[CP.015] Communications port number A start timed out!

[CP.016] Communications port number A did not come ready!

[CP.017] Invalid communications port number A I/O buffer size!

[CP.018] Communications port name A not found!

[CP.019] Communications port A not attached!

[CP.020] Communications port A attached.

[CP.021] Port number A is currently attached to communications port B.

[CP.022] Communications port name required!

[CP.023] Communications port A not killed!

[CP.024] Communications port A not active!

[CP.025] Communications port number A killed, named B.

[CP.026] Communications port A not found.

[CP.027] Port A is attached to B!

[CP.028] Not attached to communications port A!

[CP.029] Communications port A detached.

[CP.030] None attached!

[CP.031] Direct connected to communications port A.

```
[CP.032] Direct connect exited from communications port A.

[CP.033] Unable to open file "A"!

[CP.034] Capture item "A" in file "B" opened.
         Containing C items and D bytes.

[CP.035] Capture item "A" in file "B" updated.
         Added C bytes for a total of D bytes.

[CP.036] Warning: No <BREAK> key option!

[CP.037] Port number A detached from communications port B!

[CP.038] Cannot detach, port number A not in a detachable mode
         with communications port B.

[CP.039] Message sent to communications port A.

[CP.040] Message timeout of A seconds occured!

[CP.041] Message results mismatch!

[CP.042] No active communications ports found!

[CP.043] Pause of A seconds on communications ports started...

[CP.044] Pause of communications ports completed.

[CP.045] Active communications ports found, pause aborted!

[CP.046] Capture item "A" in file "B" created.


[CP.047] Playback of "A" in file "B" completed.

[CP.048] Send of A completed.
         B items, C bytes, D blocks, E minutes, F chars/sec.

[CP.049] Receive of A completed.
         B items, C bytes, D blocks, E minutes, F chars/sec.

[CP.050] Transfer aborted!

[CP.051] A 'B' frames released to overflow.

[CP.052] Aborted ID unknown!

[CP.053] Information request unknown!

[CP.054] Information request response invalid!

[CP.055] A overruns, B timeouts, C resends, D sync losses.

[CP.056] Unable to read item "A" from file "B"!

[CP.057] Script "A" in file "B" started.

[CP.058] Missing port table for port A from the "B" file!

[CP.059] Packet size A is invalid!
         Port B input buffer size is C,
         and output buffer size is D.

[CP.060] The following verb is missing from the Master Dictionary (MD):
         A
         Port B may be in an ambiguous state!

[CP.061] Expiration date and time has been reached for this transfer!

[CP.074] Transfer timeout error!

[CP.075] Port name A is being used on port number B.

[CP.078] Baud rate "A" is invalid!
         Allowable baud rates are:
         110 150 300 600 1200 2400 4800 9600 19200 38400
```

```
[CP.085] Item "A" is not in file "B"!


[CP.086] Script "A" in file "B" completed.

[CP.090] Communications port number A kill timed out!

[CP.091] Missing from port A data element:
         B

[CP.092] Communications port number A being started by port B,
         it is not available!

[CP.093] Insufficient disk space to create buffer of A frames.

[CP.094] Maximum packet size is A.

[CP.095] Parity "A" is invalid!
         Allowable parity values are:
         NONE ODD EVEN MARK SPACE

[CP.096] Stop bits "A" is invalid!
         Allowable stop bit values are:
         1 2

[CP.097] Word length "A" is invalid!
         Allowable word length values are:
         7 8

[CP.098] Communications table frame is A.

[CP.099] Invalid port A data element:
         B
         Data found:
         C
         Correction:
         D

[CP.100] Remote aborted with the following error:
         A

[CP.101] The abort command was used.

[CP.102] Packet segment size invalid "A".

[CP.103] Script syntax error:
         A
         B

[CP.104] Escape key is "A".

[CP.105] Alternate break key is "A".


[CP.106] Script "A" in file "B" aborted by user!


[CP.107] ASCII upload of A items in file "B" completed.


[CP.108] ASCII upload of file "A" aborted!
         B items successfully uploaded.
         Aborted ID was "C".

[CP.109] ASCII upload of file "A" started.

[CP.110] A bytes, B minutes, C chars/sec.

[CP.111] Creating package "A" in file "B".

[CP.112] Updating package "A" in file "B".

[CP.113] Viewing package "A" in file "B".

[CP.114] Extracting from package "A" in file "B".
```

```
[CP.115] Package "A" in file "B" updated.

[CP.116] Package "A" in file "B" added.

[CP.117] Package "A" in file "B" completed.

[CP.118] A commands, B entries, C bytes, D package items.

[CP.119] Unable to read package "A" from file "B"!

[CP.120] Package error:
         A

[CP.121] Direct mode active for A minutes.

[CP.122] ASCII download of A items in file "B" completed.


[CP.123] ASCII download of file "A" aborted!
         B items successfully downloaded.
         Last saved ID was "C".

[CP.124] ASCII download of file "A" started.

[CP.125] A bytes, B minutes, C chars/sec.

[CP.126] Capture item(s) "A" in file "B" deleted.

[CP.127] Break sent to communications port A.

[CP.128] Break not available for communications port A.

[CP.129] A 'B' frames remaining in use.

[CP.130] Communication ports are inactive.

[CP.136] A sent to communications port B.

[CP.137] A capture items, B bytes, created C
         by D on port E, last updated F.

[CP.138] Stats of file "A":
         B items; C bytes; D bit checksum total = E.

[CP.139] A minimum size; B average size; C maximum size.

[CP.140] Reset port executed on communications port A.

[CP.141] Minimum packet size reached; packet size was A bytes!

[CP.142] Baud rate: A, parity: B, stop bits: C, word length: D.
         Input buffer: E bytes, output buffer: F bytes, entry: G.

[CP.143] Transfer aborted by remote!

[CP.144] Spooler send completed.
         A entries, B bytes, C blocks, D minutes, E chars/sec.

[CP.145] Spooler receive completed.
         A entries, B bytes, C blocks, D minutes, E chars/sec.

[CP.146] XModem send of A completed.
         B items, C bytes, D blocks, E minutes, F chars/sec.

[CP.147] XModem send of A aborted.
         B items, C bytes, D blocks, E minutes, F chars/sec.

[CP.148] Result code: A, B
         Protocol: C

[CP.149] YModem send of A completed.
         B items, C bytes, D blocks, E minutes, F chars/sec.

[CP.150] YModem send of A aborted.
         B items, C bytes, D blocks, E minutes, F chars/sec.

[CP.151] XModem receive of A completed.
```

```
         B items, C bytes, D blocks, E minutes, F chars/sec.

[CP.152] XModem receive of A aborted.
         B items, C bytes, D blocks, E minutes, F chars/sec.

[CP.153] YModem receive of A completed.
         B items, C bytes, D blocks, E minutes, F chars/sec.

[CP.154] YModem receive of A aborted.
         B items, C bytes, D blocks, E minutes, F chars/sec.

[CP.155] Error stats:
           Timeouts              A
           Sequence errors       B
           Checksum/CRC errors   C
           Short data errors     D
           Nak'd packets         E
           Invalid packets       F
```

APPENDIX F
File Contents

The following tables describes some of the file names that are used in the communications software and their contents.

PowerComm Data Files (Read / Write)

File Subfile Record	Description
LSCP.TBL,	
BUFFER	Must be a DY file (it is never to be backed up or restored) with the following dynamic contents:
BUFFER.TBL	Contains workspace buffers
COMM.TBL	Contains active communications port information
EMSG*n	Contains the last PowerComm TCL command error messages and parameters, where "n" is the port number of the port executing the command
CONTROL	Control file with the following static contents:
GLOBAL	Global PowerComm parameters
PORT*G	Global default port definition item
PORT*n	Port definition items, where "n" is the port number

PowerComm Software Files (Read Only)

File Subfile Record	Description
LSCP,	
BP.EXAMPLES	Contains the included sample program examples
CATALOG	Item names to catalog for LSCP,BP
CONTROL	Control file with the following contents:
SCREEN*name	Entry screen "name"
MENU*name	Menu "name"
VERBS	Contains a list of verbs for this computer
COMPUTER	Contains information about the type of computer
MODES.TABLE	Contains the user mode definitions for this computer
EQU	Contains the standard equate definitions for files and user modes
ERRMSG	Contains the PowerComm error messages that are copied to the MESSAGES or the ERRMSG file during installation
HELP	Help for TCL commands
NEW.MD	Items to copy to the MD during installation and when installing PowerComm to a user account
MODES	Contains the assembly object code for computer
PROC	Procs for PowerComm commands at TCL

Index