



*Advanced
User's Guide*

PERCON

**Falcon DOS Portable Terminals
Advanced User's Guide
©1998**

PERCON

**1800 Millrace Drive
Eugene, OR 97403**

Voice: (541) 344-1189

Fax: (541) 344-1399

E-mail: info@percon.com

Web: <http://www.percon.com>

All rights reserved. No part of this work may be reproduced, transmitted, or stored in any form or by any means without prior written consent, except by a reviewer, who may quote brief passages in a review, or as provided for in the Copyright Act of 1976.

Percon® is a registered trademark of Percon Incorporated. Falcon™ is a trademark of Percon Incorporated.

Datalight® is a registered trademark of Datalight, Inc., and ROM-DOS™ is a trademark of Datalight, Inc.

PhoenixCARD Manager Plus ©1993, 1994 Phoenix Technologies Ltd.

Windows® is a registered trademark of Microsoft Corporation.

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where these designations appear here and the authors were aware of a trademark claim, the designations have been printed with a trademark (™) symbol.

The information contained in this document is subject to change without notice.

Introduction

This book is a supplement to the *Falcon DOS Portable Terminals User's Guide*. It contains technical information about the Falcon's system configuration, disk drives, utilities, and DOS commands. It also provides information about using PC cards with the Falcon, resetting the Falcon, and programming the Falcon 4-Slot Dock. Included in appendixes are a table of programming parameters, bar codes for setting those parameters, and connector configurations for the Falcon portables, Falcon Dock, and Falcon 4-Slot Dock.

This book is provided as a reference guide for system administrators, developers, and programmers who want to create end-user solutions for Falcon DOS portable terminals. It is not intended for use by first-time Falcon users.

NOTE: *When used in this book, the word "Falcon" generally refers to any or all of the DOS portable models (310, 315, 320, and 325). Where information applies to specific models, those models are clearly identified, as in the following example:*



Getting Help

The most comprehensive source for technical support and information for Percon products is the Percon web site, at www.percon.com. The site offers answers to frequently asked questions (FAQs), software updates, patches, demos, product documentation, and instructions for returning products for repair.

Another excellent source for technical assistance and information is your authorized Percon reseller. Your reseller is directly acquainted with your type of business, application software, and computer system and, therefore, is in the best position to provide individualized assistance.

If you cannot find a solution to your technical support questions through our Web site or your reseller, you may contact Percon technical support directly via e-mail at tech@percon.com.

Contents

<i>Introduction</i>	iii
-------------------------------	-----

<i>Getting Help</i>	iii
-------------------------------	-----

Chapter One:

<i>Using Falcon DOS Terminals</i>	1
Programming the Laser Triggers	2
Using PC Cards	2
Opening the PC Card Slot Cover	3
Inserting a PC Card	4
Card Recognition and Configuration	5
Removing a PC Card	6
The Serial Port	6
The Disk Drives	7
Configuring the Falcon	8
Transferring Files	8
An Example	9

Chapter Two:

<i>The Falcon Configuration Utility</i>	11
Installing and Starting the Utility on Your PC	12
The Main Menu	13
The Custom Configuration Menu	15
The File Configuration Windows	18
The Program Settings Windows	26
The Comm Settings Dialog Box	29
The File Transfer Window	30

Chapter Three:

<i>Using XFER</i>	35
About XFER	36
Syntax and Parameters	36
The XFER_ARGS Environment Variable	50
Multiple-Option Blocks	51
The Modem-Initialization File	53
Format	53
Keywords	53
Sample Modem-Initialization File	56
Performance	56
Error Codes	57

Chapter Four:

<i>The Falcon 4-Slot Dock</i>	59
Configuring the Falcon 4-Slot Dock.	60
Baud Rate	60
Operating Modes	61
Basic Communications with the Falcon 4-Slot Dock	61
Loading 4SLOT.SYS	61
Transferring Files with XFER	62

Chapter Five:

<i>Default Disk Drive Organization and Unit Configurations</i>	63
File System Format	64
Overview	64
Drive A	64
Drive B	66
Drive C	68
Drive D	69
Drive E.	69

System Software	69
BIOS and DOS	69
Device Drivers and Utilities	70
PC Card and RF Networking Software	70
Unit Configurations	70
Default Configuration	70
I/O PC Card Support	71

Chapter Six:

<i>Resetting the Falcon</i>	73
Warm Boot	74
Cold Boot	74
Safe Boot	75
Hardware Reset	76
Recommended Programming Practices	77

Chapter Seven:

<i>Using PC Cards</i>	79
Overview	80
Configuring Your PC Card	80
PhoenixCARD Manager Plus (PCM+)	80
ATA Cards	81
Fax/Modem Cards	81
LAN Cards	82
RF Cards	82

Chapter Eight:

***System Utilities* 83**

CFGDEV.SYS 84

DECODE.SYS 85

FLASHDSK.SYS 85

FORMAT.COM 85

LOCK.COM 86

ORGANIZE.COM 86

PM.COM 87

VDISK.SYS 88

XFER.EXE 88

Chapter Nine:

***ROM-DOS Commands* 89**

Command Overview 90

ROM-DOS vs. MS-DOS 92

Command Descriptions 93

 ATTRIB 93

 BUFFERS 94

 CHKDSK 94

 COMMAND 96

 DIR 97

 FCBS 100

 FIND 100

 HELP 101

 NEWFILE 102

 PRINT 103

 SHARE 104

 SWITCHES 105

 TREE 106

 VER 106

 XCOPY 107

 XDEL 108

Appendix A:

<i>Programming Parameters</i>	111
Code 39.	113
Interleaved 2 of 5	113
Matrix 2 of 5	113
Standard 2 of 5	113
Code 11.	113
Codabar/Ames.	114
MSI	114
Code 93.	114
Universal Product Code-A (UPC-A).	114
Universal Product Code-E (UPC-E)	114
European Article Numbering (EAN)/Japan Article Numbering (JAN)	114
UPC, EAN, JAN Extensions	115
Code 128.	115
Labelcode 4/5	115
Other Controls.	115

Appendix B:

<i>Connector Configurations</i>	117
Falcon Portable Terminals	118
Laser-Scanner Connector.	118
Serial Port Jack	118
Falcon Dock.	119
25-Pin Connector	119
Cradle Jack	120
Falcon 4-Slot Dock.	120
Host-Interface Cable	121
Dock-Network Cable Jacks	121

Appendix C:

***Bar Codes for Configuring Falcon* 123**

Predefined Defaults 124

Code 39. 124

Interleaved 2 of 5 125

Matrix 2 of 5 126

Standard 2 of 5 127

Code 11. 128

Codabar/Ames. 128

MSI 129

Code 93. 130

Code 128. 131

Labelcode 4/5 132

UPC-A 132

UPC-E 132

EAN/JAN 133

UPC/EAN/JAN Extensions 133

Other Controls. 134

***Index* 137**

>> Chapter One

Using Falcon DOS Terminals

This chapter provides information about advanced features of Falcon DOS portable terminals. It does not cover Falcon basics, such as use of the keypad and viewport.

Finding Out More: For basic information about the Falcon, see the Falcon DOS Portable Terminals **User's Guide**.

CHAPTER CONTENTS

<i>Programming the Laser Triggers</i>	<2>
<i>Using PC Cards</i>	<2>
<i>Opening the PC Card Slot Cover</i>	
<i>Inserting a PC Card</i>	
<i>Card Recognition and Configuration</i>	
<i>Removing a PC Card</i>	
<i>The Serial Port</i>	<6>
<i>The Disk Drives</i>	<7>
<i>Configuring the Falcon</i>	<8>
<i>Transferring Files</i>	<8>
<i>An Example</i>	

>> Programming the Laser Triggers

Normally, the left trigger operates the Falcon laser or another bar code reader attached to the Falcon, and the right trigger toggles the Falcon in and out of Alpha mode (models **310** and **315**) or Function mode (models **320** and **325**). You can reprogram one or both of the laser triggers to act as equivalents, or “aliases,” of keypad keys. To use a trigger as an alias for a keypad key:



First, hold down the **FN 1** and **FN 2** keys and press the **SWAP** key. The program-trigger cursor (P) should appear in the viewport. Press the trigger that you want to change, and then press the keypad key that you want to assign to the trigger.



First, hold down the **FN** key and press the **SWP** key. The program-trigger icon (⊗) should appear in the viewport. Next, press the trigger that you want to change, and then press the keypad key that you want to assign to the trigger.

For example, to turn the right trigger on a model **320** into an alias for the **ENTER** key, hold down the **FN** key and press the **SWP** key to enter trigger-program mode. Then press and release the right trigger, and press the **ENTER** key. The right trigger will now work as a second **ENTER** key.

To change a reassigned trigger back to a laser trigger, put the Falcon into trigger-program mode (see above) and press the trigger twice.

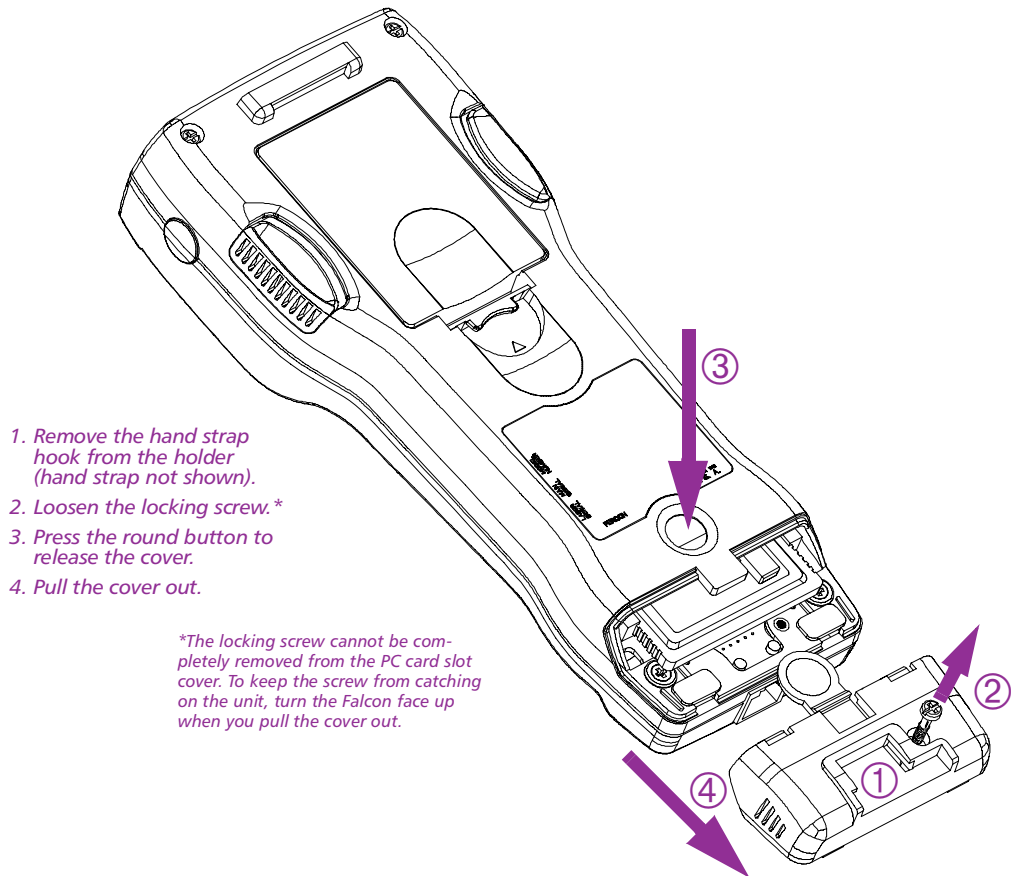
>> Using PC Cards

The Falcon has a slot for plugging in PC cards, and each unit is factory-equipped with PhoenixCARD Manager Plus PC card drivers. PC cards provide such features as network connectivity, modem connectivity, and wireless capability. Their primary purpose in the Falcon is to provide additional memory storage by functioning as a disk drive. (See page [7](#) for information about the Falcon’s drives.)

Opening the PC Card Slot Cover

The PC card slot is located near the bottom on the back of the Falcon (see figure 1-1). The slot is protected by a cover. Detach the elastic hand strap on the back of the Falcon by pulling its hook out of the holder near the base. If the slot cover is secured by a screw, loosen the screw. Then, while pressing the round button above the slot cover, slide the cover out and away from the Falcon.

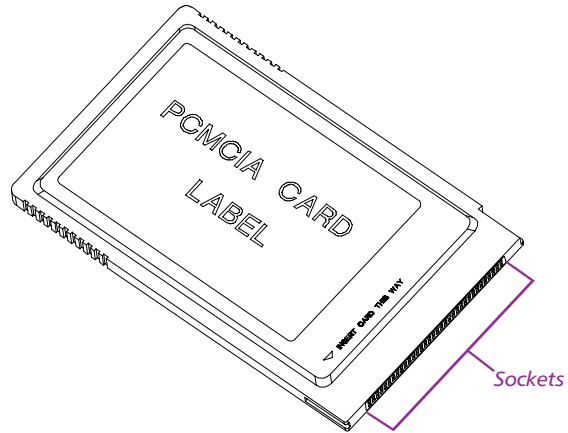
Figure 1-1: Removing the PC Card Slot Cover



Inserting a PC Card

All PC cards have two rows of small sockets on one end (see figure 1-2). The cards also have face-up and face-down sides. The card manufacturer's label is usually on the face-up side.

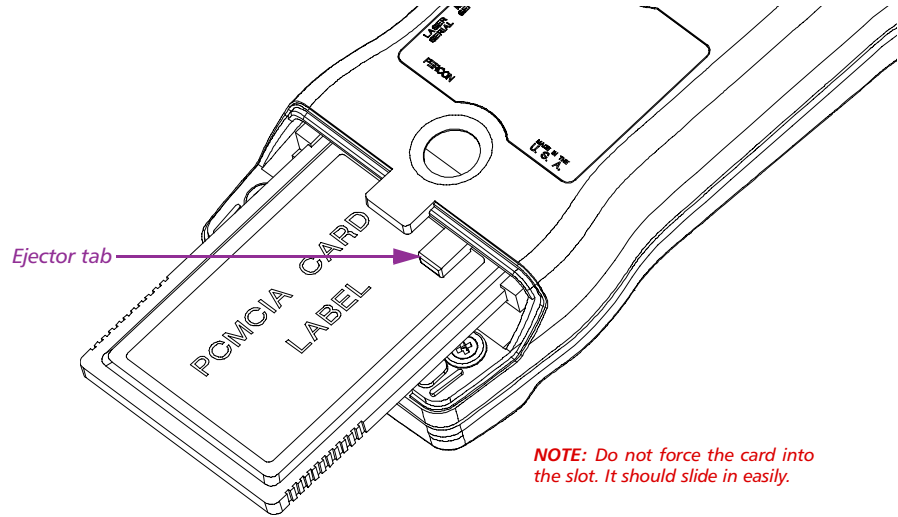
Figure 1-2: A Typical PC Card



NOTE: Turn the Falcon unit off before inserting or removing a PC card.

With the Falcon face down and the PC card face up, insert the end of the card with the sockets into the card slot (see figure 1-3). There are two tracks inside the slot to help you guide the card. Push the card firmly into the slot until the ejector tab slides out. Then replace the PC card slot cover.

Figure 1-3: Inserting a PC Card into the Slot



NOTE: The PC card slot on the Falcon is designed so that you cannot insert a card upside down or backward. If you cannot push the card all the way into the slot easily, you probably do not have the card positioned properly. Make sure you are putting the end with the holes into the slot first. Then flip the card upside down and try to insert it again.

Card Recognition and Configuration

Once you have inserted the card in the slot, turn the Falcon on. The Falcon will try to recognize and configure the card.

If the Falcon responds with one beep when you turn it on, the PhoenixCARD Manager Plus drivers successfully recognized and configured the card. If the Falcon does not beep, the drivers might not have successfully configured the card. The Phoenix drivers may not be loaded in the Falcon, or the beeper may be disabled. Check with your system administrator.

NOTE: In some cases, drivers provided by a specific card's vendor are responsible for configuring the card. If you are using one of these cards, you might receive no audio signals for card configuration. See the configuration instructions that came with the card.

Removing a PC Card

Inside the PC card slot is a tab that ejects the installed card (see figure 1-3). Push the end of the ejector tab into the Falcon. As you do so, the PC card should slide partway out of the slot. Hold the card by the edges and pull it the rest of the way out.

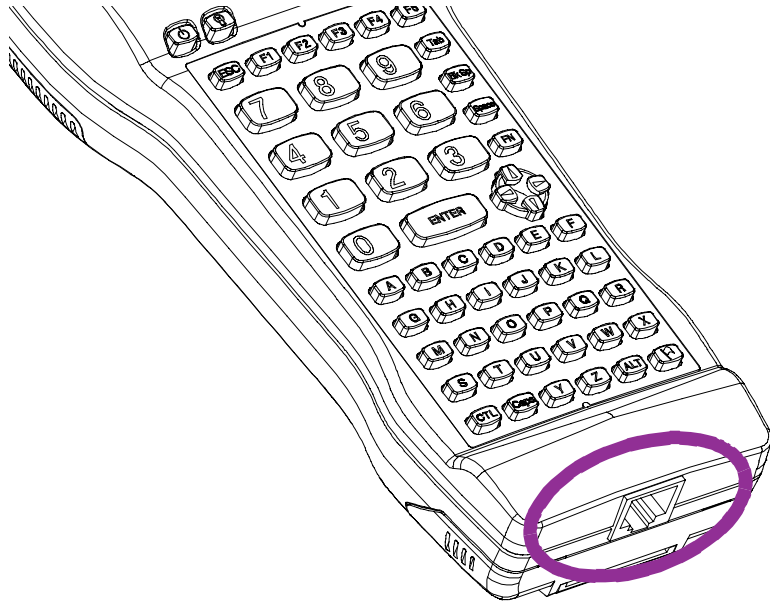
>> The Serial Port

The Falcon has a port for serial communications with a PC. The port is located at the base of the Falcon unit (see figure 1-4). Designated as COM1, it is a 10-pin telephone-style jack providing a standard RS-232 connection.

With a serial cable connected to it, the port allows communications with a host computer or any serial device, such as a printer or modem. The serial port also provides a connection for communications and battery recharging in the Falcon Dock and Falcon 4-Slot Dock.

For the wiring configuration of the serial port, see appendix B.

Figure 1-4: The Serial Port



>> The Disk Drives

The Falcon contains four logical disk drives that provide storage for system files, applications, and data.

Drive A Drive **A** is a read-only drive. Its contents cannot be changed.

Drive B Drive **B** is a read-only drive. It is used to store system utilities and to initialize the boot process. Its contents cannot be changed.

Drive C Drive **C** is a 700-kilobyte flash disk drive with full read and write access. This drive contains DOS command files, PC card drivers, Falcon utilities, and executable files and associated files for applications. It may also contain additional `config.sys` and `autoexec.bat` files to configure your Falcon to run applications.

Drive D Drive **D** is a RAM disk drive. The RAM disk is used primarily for data storage. Programs that need to be loaded into memory and then quickly removed from memory can also be placed here. Drive **D** can also be used for scratch disk space or temporary files.

NOTE: *As with any RAM drive, data on drive **D** can be lost if the Falcon has a power failure caused by low batteries or a system reset. For truly secure data collection, store your data on drive **C** or on an ATA flash card.*

Additional Drives

Drive **E** exists only if your Falcon has been configured to use PC ATA flash cards. The PC card looks like a hard disk drive to the operating system. You can use drive **E** for safe and permanent storage of data.

Alternately, drive **E** could be a peer-to-peer or client-server network drive that is accessed through a wireless access point or Ethernet network card link.

Your Falcon may have other additional logical drives. These might be RAM drives, ATA flash cards, or network drives accessed via wireless access points.

>> *Configuring the Falcon*

If your Falcon is not already configured for use when you get it, you can use the Falcon Configuration Utility to install applications and set options for bar code scanning. The Falcon Configuration Utility operates under Windows 3.1, Windows 95, and Windows NT. For instructions on using the utility, see chapter [2](#).

You can also use bar codes to change settings in your Falcon. Appendix [C](#) has bar codes for many common settings.

>> *Transferring Files*

To transfer data or program files, connect your Falcon unit to the host computer with any of the following accessories:

- Falcon serial cable
- Falcon Dock
- Falcon 4-Slot Dock

Your Falcon application may have simple file-transfer options, or you can use the Falcon XFER utility.

When you use the Falcon Configuration Utility to transfer files (see chapter [2](#)), the configuration utility runs XFER automatically for you. If you are not using the configuration utility, using XFER involves entering commands at the DOS command line on both the Falcon and the PC.

The XFER utility is loaded into the Falcon at the factory and placed on drive **B**. If your PATH statement has not been changed, you can run XFER from any drive on the Falcon. However, before you can run XFER on your PC, you must install the Falcon Configuration Utility onto your hard drive. By default, `xfer.exe` is placed in the `\percon\cfgutil\program` directory.

You can transfer a single file by using XFER with the Xmodem protocol (the default protocol) or transfer multiple files with the Zmodem protocol.

The command line syntax for XFER is as follows:

```
XFER [/option1 [/option2] . . .] filename
```

You can use a slash (/) or a hyphen (-) to denote options, and you can use uppercase or lowercase letters for them. Options can be placed before or after the filename on the command line. A sample command line appears at the end of this section.

Basic options for Xmodem protocol are listed and described in table 2-1. The “Default” column indicates whether the option is used (“On”) or ignored (“Off”) if you do not include it in the command line. For options that have two or more possible values, the default value is given.

NOTE: For more information about XFER, including Zmodem options, see chapter 3.

Table 2-1: XFER Options (Xmodem Protocol)

Option	What It Does	Default
<i>filename</i>	Identifies the file to be transferred or received.	<i>None</i>
#	Specifies the communication port to use. Replace the # symbol with the desired setting: 1 = COM1 2 = COM2	1
B#	Specifies the baud rate. Replace the # symbol with the desired setting: 2400 4800 9600 19200 38400 57600 115200	19200
D#	Specifies the number of seconds for XFER to wait for activity before cancelling the transfer. Replace the # symbol with the desired number of seconds for the timeout delay. Acceptable values are 0 (no timeout) through 65,535.	60
H or ?	Displays help for the XFER command.	Off
o	Overwrites an existing file with a new file having the same name.	Off
R	Receives the specified file.	Off
T	Transmits the specified file.	On

An Example To transfer a file named `foo` from a PC to a Falcon using Xmodem protocol, you could use the following command lines.

On the PC: `xfer foo`. This command causes the computer to send the specified file using XFER’s default settings.

On the Falcon: `xfer /r foo`. This command causes the Falcon to receive the specified file transmitted from the PC.

>> Chapter Two

The Falcon Configuration Utility

The Falcon Configuration Utility provides a simple way to change the Falcon's settings for bar code symbologies and serial communications. You can also use it to load programs and files into the Falcon. The utility runs under Windows 3.1x, Windows 95, and Windows NT. This chapter describes how to install the utility and use it to configure your Falcon.

CHAPTER CONTENTS

<i>Installing and Starting the Utility on Your PC</i>	<12>
<i>The Main Menu</i>	<13>
<i>The Custom Configuration Menu</i>	<15>
<i>The File Configuration Windows</i>	<18>
<i>The Program Settings Windows</i>	<26>
<i>The Comm Settings Dialog Box</i>	<29>
<i>The File Transfer Window</i>	<30>

>> *Installing and Starting the Utility on Your PC*

To install the configuration utility, insert the disk labeled “Falcon Configuration Utility Disk 1” into your computer’s floppy drive. Then complete the following steps:

1. In Windows, run the `setup.exe` file from the disk in drive **A**.
2. In the “Installation Options” window, select the radio components you’d like to install, if any. Select **Next** to move on.
3. In the “RF Installation Options” window, uncheck the check boxes for any components you do not want to install. Then select **Next** to move on.
4. In the “Select a Group Name” window, select a program group in which to place the Falcon Configuration Utility icons. Select **Next** to move on.
5. In the next window, specify the directory in which to place the Falcon Configuration Utility files. Select **Next** to move on.
6. When the installation is completed, select **Finish** in the final window.



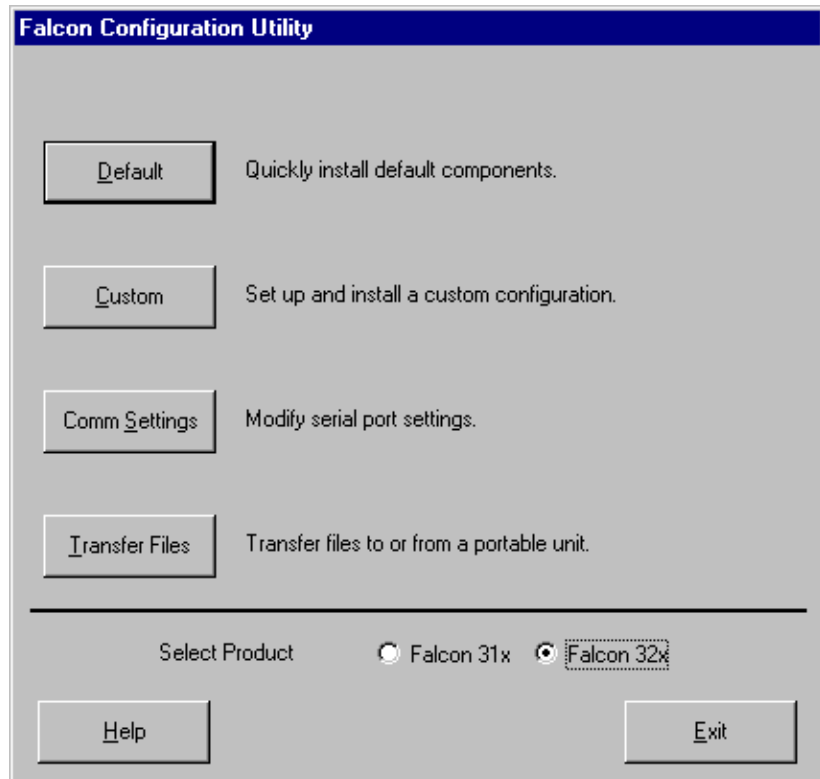
To start the Falcon Configuration Utility, double-click on the FALCON icon in the program group.

The first screen that appears is the main menu.

>> The Main Menu

This menu gives you access to all the configuration settings for the Falcon.

Figure 2-1: The Falcon Configuration Utility Main Menu

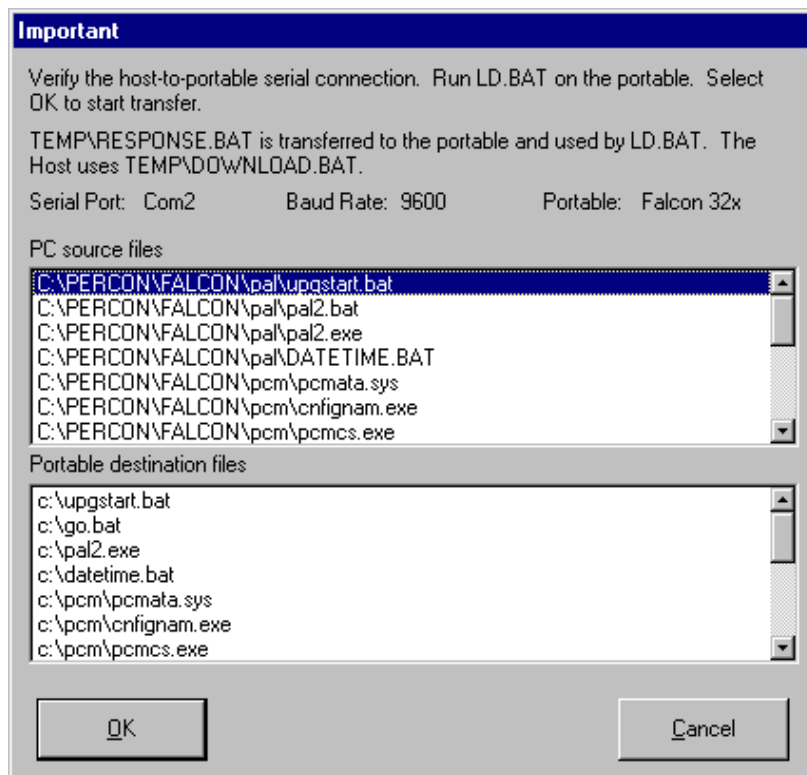


Default Select this option to load the original factory configuration into your Falcon unit. The configuration utility prepares files to be transferred to the Falcon and opens a dialog box titled “Important.” Check the file lists in both sections of the dialog box to see if they are correct and complete. If you need to add, delete, or rename files, select **Cancel** from the dialog box. Then use the **Custom** selection in the main menu to build your own lists. (See pages [15–18](#) for information about creating custom configurations.)

The Falcon Configuration Utility

If the file lists in the dialog box are correct, make sure your Falcon unit is properly connected to the serial port specified in the Comm Settings dialog box (see page 29). Then run the `ld.bat` file on the Falcon, and select **OK** in the Important dialog box on the computer.

Figure 2-2: The Important Dialog Box



Custom Select this option to select or modify configuration files or program files to be loaded into your Falcon unit. See the next section for information on the Custom Configuration menu.

Comm Settings Select this option to modify settings for your computer's serial port. See page 29 for information on the Comm Settings dialog box.

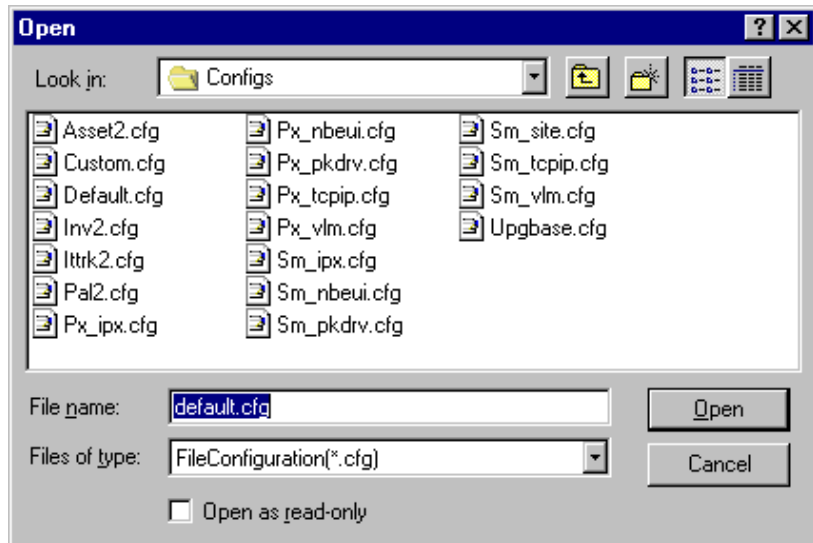
Transfer Files Select this option to transfer data files between the Falcon and the host computer. See pages 30–33 for information on the File Transfer window.

Select Product Select **Falcon 31x** if you are configuring Falcon model 310 or 315. Select **Falcon 32x** if you are configuring model 320 or 325.

>> The Custom Configuration Menu

When you select **Custom** from the main menu, an Open dialog box appears. Use this dialog box to select a configuration file from the Configs folder. The configuration file contains the information that will be presented when you select the File Configuration button on the Custom Configuration menu.

Figure 2-3: The Open Dialog Box for Selecting a Configuration File

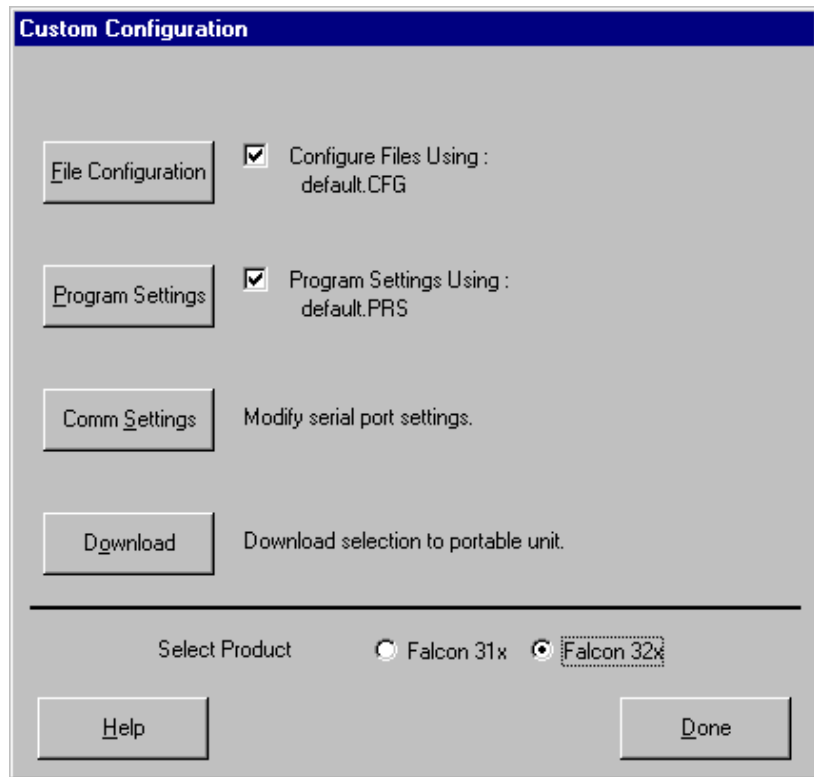


NOTE: If you select **Open as read-only** and make changes to the configuration settings, you will need to use a new name for the file to save the changes.

When the configuration file is finished loading, a second Open dialog box appears. Use this dialog box to select a program-settings file from the Progsets folder. The program-settings file contains the information that will be presented when you select the Program Settings button on the Custom Configuration menu.

After the program-settings file is loaded, the Custom Configuration menu appears.

Figure 2-4: The Custom Configuration Menu



File Configuration

Select this option to choose application files to be loaded into the Falcon. See pages [18–25](#) for information about the File Configuration windows.

Configure Files Using

This field shows the configuration file that will be used to specify the files that will be loaded into the Falcon. If you do not want to load any files, turn the switch off.

Program Settings

Select this option to view or change settings for bar code symbologies and other programmable Falcon options. See pages [26–29](#) for information about the Program Settings windows.

Program Settings Using

This field shows the program-settings file that will be used. If you do not want to load any program settings, turn the switch off.

- Comm Settings** Select this option to view or change serial communications settings for the Falcon. See page [29](#) for information about the Comm Settings dialog box.
- Download** When you finish customizing the Falcon configuration, select this option to load the custom configuration into your Falcon. When you select it, the configuration utility prepares files to be transferred to the Falcon and opens the Important dialog box (see “[Default](#)” on page [13](#)). Make sure your Falcon unit is properly connected to the serial port specified in the Comm Settings dialog box (see page [29](#)). Then run the `ld.bat` file on the Falcon, and select **OK** in the Important dialog box on the computer.
- Select Product** Select **Falcon 31x** if you are configuring Falcon model 310 or 315. Select **Falcon 32x** if you are configuring model 320 or 325.
- Done** Select this option to return to the main menu. If you made any changes to the file configuration or program settings, one or both of the following prompts appear:

Figure 2-5: The Prompt for Saving Changes to the Current File Configuration

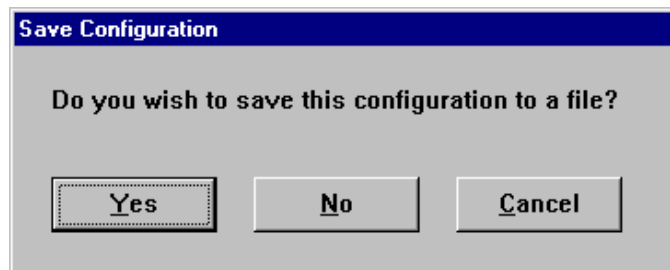
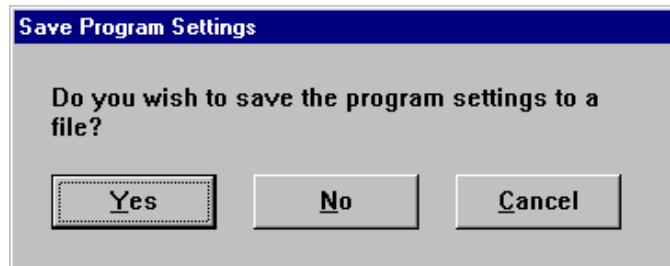


Figure 2-6: The Prompt for Saving Changes to the Current Program Settings



Yes Select this option to save the changes. A Save As dialog box will open. Use the dialog box to specify the location and name of the new configuration or program-settings file.

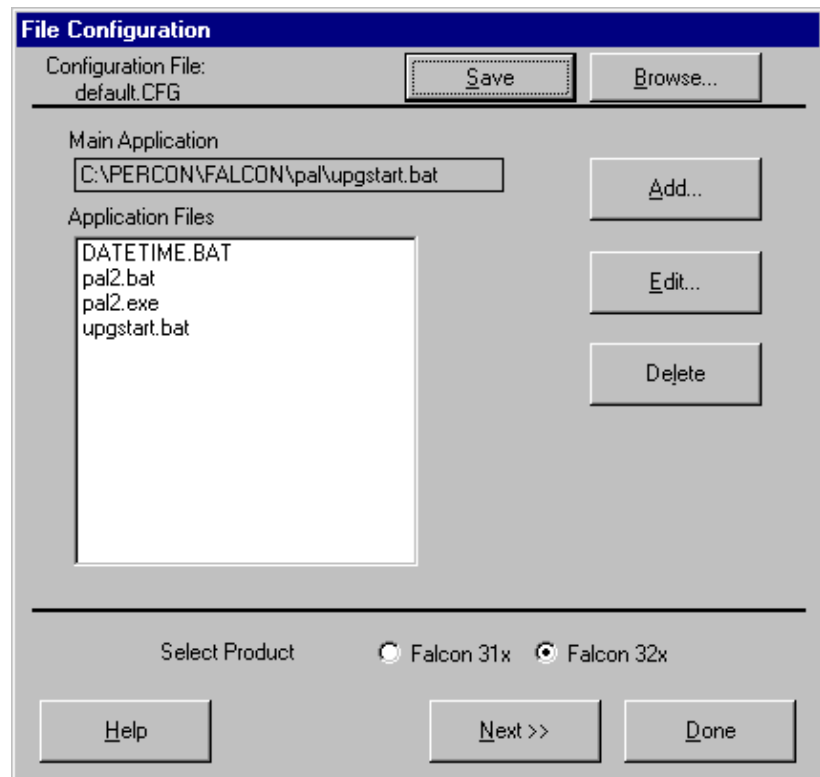
No Select this option to discard the changes.

Cancel Select this option to return to the Custom Configuration menu without saving or discarding the changes.

>> The File Configuration Windows

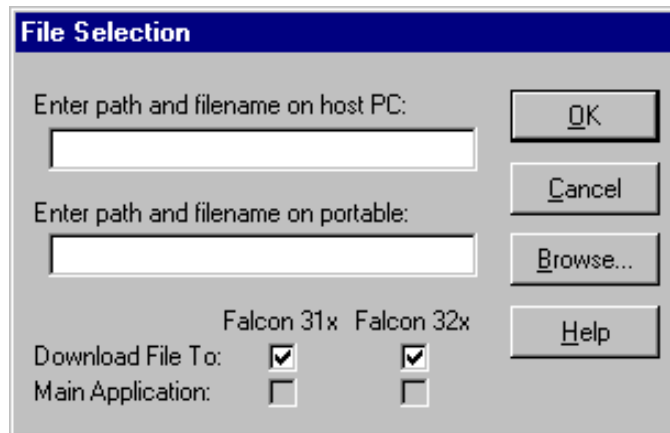
When you select **File Configuration** from the Custom Configuration Menu, the first of three File Configuration windows appears. Use these windows to choose application files to be loaded into the Falcon.

Figure 2-7: The First File Configuration Window



- Configuration File** This field shows the configuration file used to specify the files that will be loaded into the Falcon.
- Save** After adding, editing, or deleting files in the Application Files list, select **Save** to save the revised list in the current configuration file or in a new one.
- Browse** Select **Browse** to use a different configuration file. An Open dialog box will appear. Use the dialog box to choose a configuration file from the Configs folder. (If you select Browse after making changes in this or any other File Configuration window without saving the changes, the prompt shown in figure 2-5 will appear.)
- Main Application** This field identifies the default application that will run on Falcon after you complete the installation.
- Application Files** This field lists the files associated with the main application.
- Add** Select this option to include other files to be installed on your Falcon unit. The File Selection dialog box opens.

Figure 2-8: The File Selection Dialog Box for Adding an Application File



Enter path and filename on host PC Use this field to specify the file you want to transfer to the Falcon.

Enter path and filename on portable Use this field to specify the location and name you want the transferred file to have on the Falcon. The name can be the same as the original file or you can give the file a new name.

Download File To Turn this switch on if you want the specified file to be downloaded to the specified Falcon models.

Main Application Turn this switch on if you want the specified file to be the main application on the specified Falcon models.

***NOTE:** Only one file can be selected as the main application for a model series. To select another file as the main application, you must first highlight the current one in the Application Files list in the File Configuration window, select **Edit**, and turn off the Main Application switch for that file.*

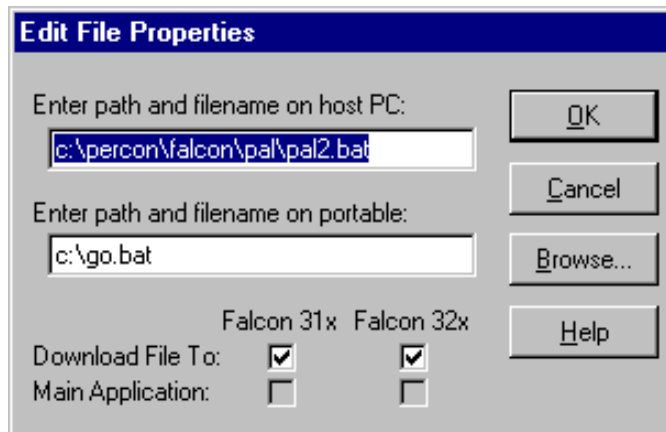
OK Select **OK** to return to the File Configuration window. The specified source file will appear in the Application Files list.

Cancel Select **Cancel** to return to the File Configuration window without adding a file to the Application Files list.

Browse Select **Browse** to view the files on your computer. An Open dialog box will appear. Use the dialog box to choose a source file to be included in the Custom configuration.

Edit Select a file in the Application Files list and then select **Edit** to change the source path or destination path for the file. The Edit File Properties dialog box opens.

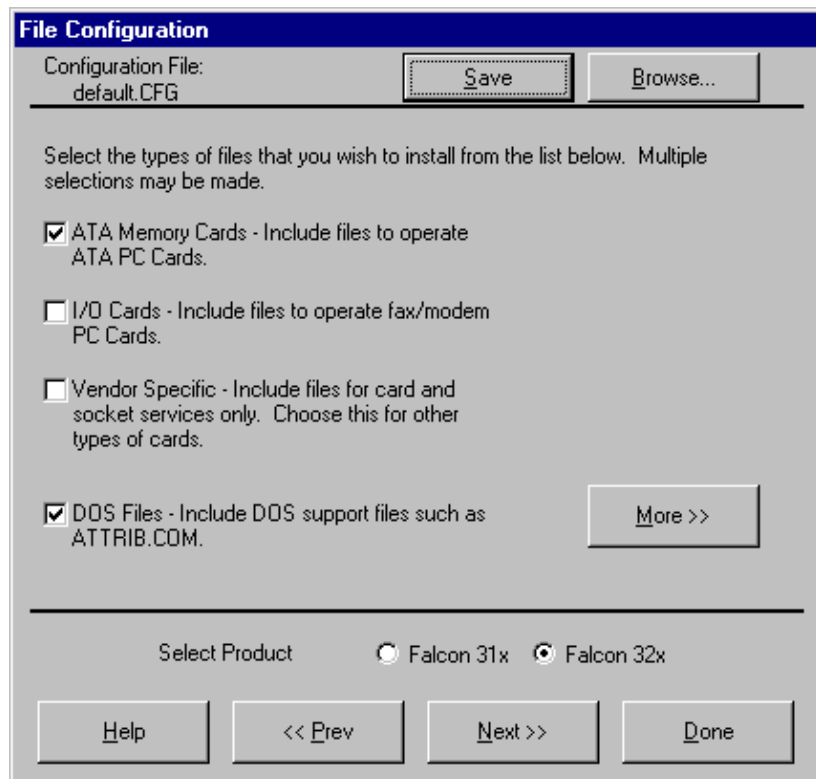
Figure 2-9: The Edit File Properties Dialog Box for an Application File



For information on using this dialog box, see [“Add”](#) on page [19](#).

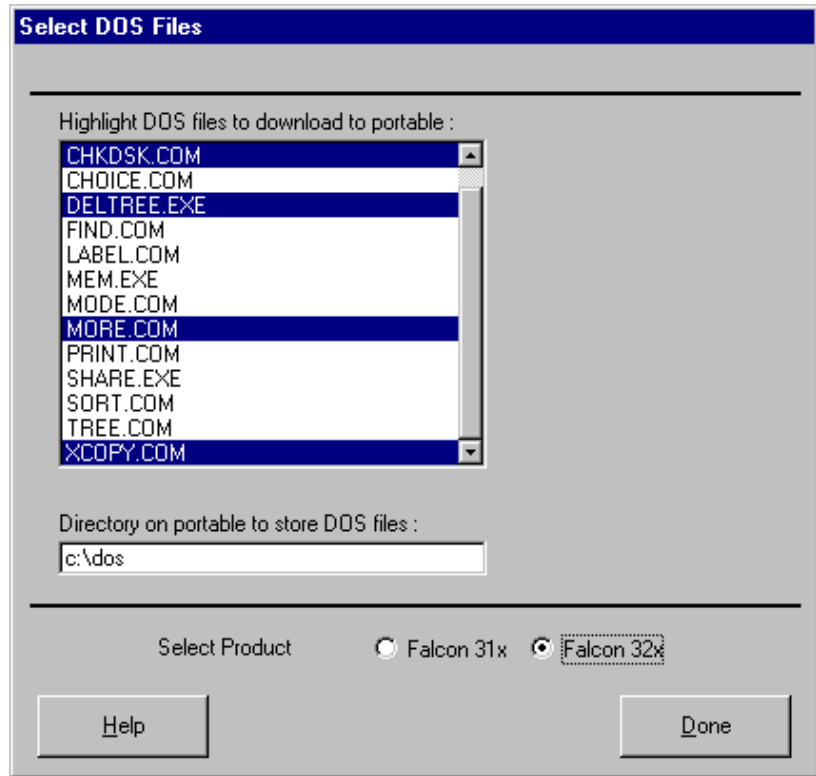
- Delete** To delete a file from the Application Files list, select the file in the list and then select **Delete**.
- Select Product** Select **Falcon 31x** if you are configuring Falcon model 310 or 315. Select **Falcon 32x** if you are configuring model 320 or 325.
- Next** Select **Next** to view or change additional file-configuration options for the custom installation.
- Done** Select **Done** if you are finished setting file-configuration options for the custom installation. If you have made any changes
The second File Configuration window appears if you select **Next** in the first window.

Figure 2-10: The Second File Configuration Window



- Configuration File** This field shows the configuration file used to specify the files that will be loaded into the Falcon.
- Save** After changing selections in this window, select **Save** to save the revisions in the current configuration file or in a new one.
- Browse** Select **Browse** to use a different configuration file. An Open dialog box will appear. Use the dialog box to choose a configuration file from the Configs folder. (If you select Browse after making changes in this or any other File Configuration window without saving the changes, the prompt shown in figure [2-5](#) will appear.)
- ATA Memory Cards** Turn this switch on to transfer drivers for ATA memory cards to the Falcon.
- I/O Cards** Turn this switch on to transfer drivers for fax/modem cards to the Falcon.
- Vendor Specific** Turn this switch on to transfer drivers for other types of PC cards.
- DOS Files** Turn this switch on to transfer files for DOS commands and utilities to the Falcon.
- More** Select **More** to add or remove DOS files from the custom installation. The Select DOS Files dialog box appears.

Figure 2-11: The Select DOS Files Dialog Box



Highlight DOS files to download to portable This field lists DOS files that are available. Files that are highlighted are currently selected to be included in the custom installation. Click on a file to select it or deselect it.

Directory on portable to store DOS files Use this field to specify where the DOS files should be placed in the Falcon unit.

Done Select **Done** to return to the File Configuration window.

Select Product

Select **Falcon 31x** if you are configuring Falcon model 310 or 315. Select **Falcon 32x** if you are configuring model 320 or 325.

Prev

Select **Prev** to return to the previous file-configuration window.

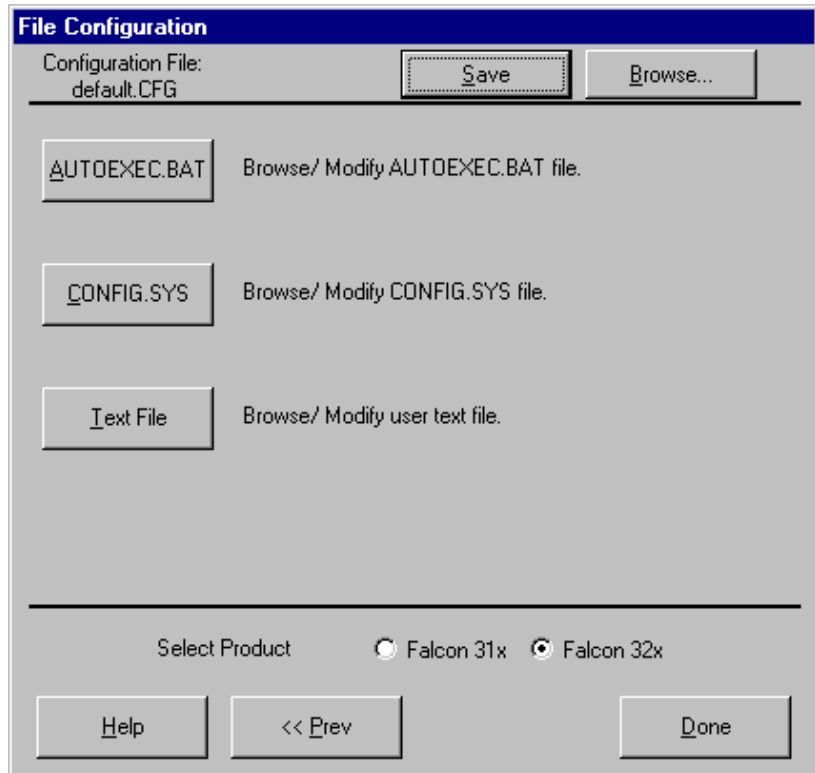
Next

Select **Next** to move on to the next file-configuration window.

Done Select **Done** if you are finished setting file-configuration options for the custom installation.

The third File Configuration window appears if you select **Next** in the second window.

Figure 2-12: The Third File Configuration Window



Configuration File	This field shows the configuration file used to specify the files that will be loaded into the Falcon.
Save	After selecting options in this window, select Save to save the revisions in the current configuration file or in a new one.
Browse	Select Browse to use a different configuration file. An Open dialog box will appear. Use the dialog box to choose a configuration file from the Configs folder. (If you select Browse after making changes in this or any other File Configuration window without saving the changes, the prompt shown in figure 2-5 will appear.)
AUTOEXEC.BAT	Select this option to insert new commands into the <code>autoexec.bat</code> file that will be transferred to the Falcon.
CONFIG.SYS	Select this option to insert new commands into the <code>config.sys</code> file that will be transferred to the Falcon.
Text File	Select this option to view or modify any text file that will be transferred to the Falcon.
Select Product	Select Falcon 31x if you are configuring Falcon model 310 or 315. Select Falcon 32x if you are configuring model 320 or 325.
Prev	Select Prev to return to the previous file-configuration window.
Done	Select Done if you are finished setting file-configuration options for the custom installation.

>> The Program Settings Windows

When you select **Program Settings** from the Custom Configuration Menu, the first of five Program Settings windows appears. Use these windows to view or change settings for bar code symbologies and other programmable options. The first window is shown below. The next three windows are similar.

Figure 2-13: The First Program Settings Window

The screenshot shows a dialog box titled "Program Settings" with a dark blue header. Below the header, it displays "Program Settings File: default.PRS" and two buttons: "Save" and "Browse...". The main area is divided into four sections for different barcode symbologies:

- Code 39:** Enable, Minimum Length: 0, Maximum Length: 20, Enable Checksum, Send Checksum, Full ASCII Mode.
- Codabar/Ames:** Enable, Minimum Length: 4, Maximum Length: 20, Send Start/Stop, Codabar to CLSI Conversion, Allow Wide Inter-Character Gaps.
- Code 128:** Enable, Minimum Length: 2, Maximum Length: 20, UCC/EAN 128 Enable.
- Code 11:** Enable, Minimum Length: 4, Maximum Length: 10, Require 2 Check Digits, Send Check Digit(s).

At the bottom, there is a "Select Product" section with radio buttons for "Falcon 31x" and "Falcon 32x" (selected). Three buttons are at the very bottom: "Help", "Next >>", and "Done".

Program Settings File

This field shows the program-settings file that will be loaded into the Falcon.

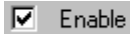
Save

After selecting options in this window, select **Save** to save the revisions in the current program-settings file or in a new one.

Browse

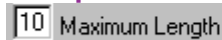
Select **Browse** to use a different program-settings file. An Open dialog box will appear. Use the dialog box to choose a program-settings file from the Progsets folder. (If you select Browse after making changes in this or any other Program Settings window without saving the changes, the prompt shown in figure [2-6](#) will appear.)

On/Off Switches



The smaller white boxes are on/off switches. Click in the box to toggle a switch.

Input Fields



Enter specific settings for parameters in the larger white boxes. (See appendix [A](#) for a table of parameters and settings.)

Select Product

Select **Falcon 31x** if you are configuring Falcon model 310 or 315. Select **Falcon 32x** if you are configuring model 320 or 325.

Prev

Select **Prev** to return to the previous program-settings window.

Next

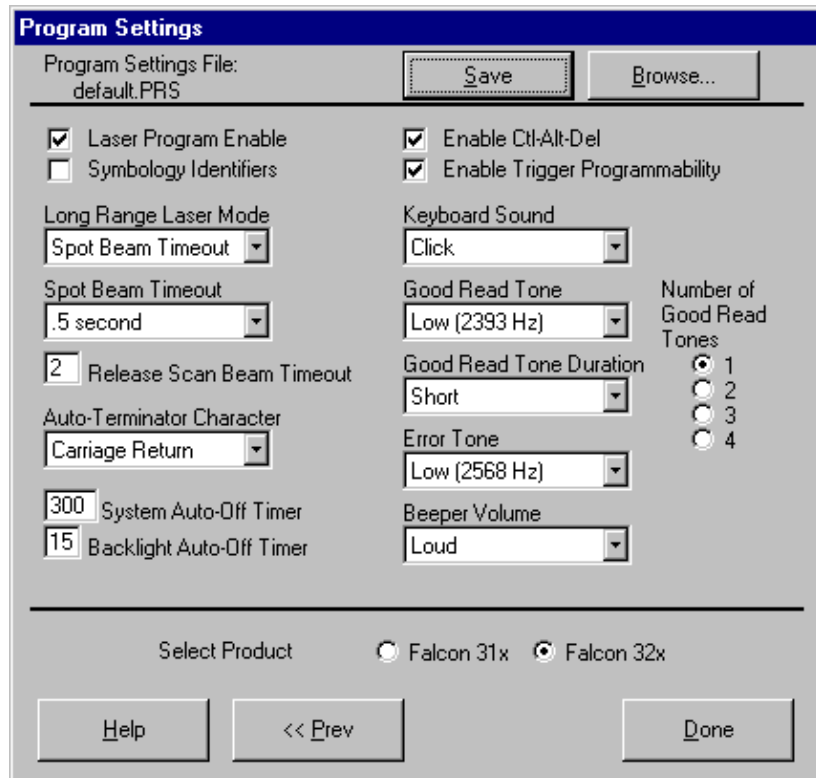
Select **Next** to move on to the next program-settings window.

Done

Select **Done** if you are finished making program settings for the custom installation.

The last Program Settings window appears if you select **Next** in the fourth window.

Figure 2-14: The Last Program Settings Window



Program Settings File

This field shows the program-settings file that will be loaded into the Falcon.

Save

After selecting options in this window, select **Save** to save the revisions in the current program-settings file or in a new one.

Browse

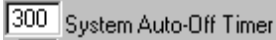
Select **Browse** to use a different program-settings file. An Open dialog box will appear. Use the dialog box to choose a program-settings file from the Progsets folder. (If you select Browse after making changes in this or any other Program Settings window without saving the changes, the prompt shown in figure 2-6 will appear.)

On/Off Switches



The smallest white boxes are on/off switches. Click in the box to toggle a switch.

Input Fields



Enter specific settings for parameters in the larger white boxes. (See appendix A for a table of parameters and settings.)

Drop-Down Lists



Click on the list to view the options, and click on the option you want to select.

Radio Buttons



Select the setting you want by clicking on it.

Prev

Select **Prev** to return to the previous program-settings window.

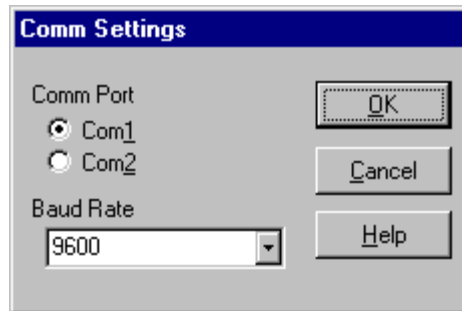
Done

Select **Done** if you are finished making program settings for the custom installation.

>> The Comm Settings Dialog Box

When you select **Comm Settings** from the main menu or the Custom Configuration menu, the Comm Settings dialog box appears. Use this dialog box to view or change settings for serial communications with the Falcon.

Figure 2-15: The Comm Settings Dialog Box



Comm Port

Select the serial port that your PC will use to communicate with the Falcon. The default port is COM1.

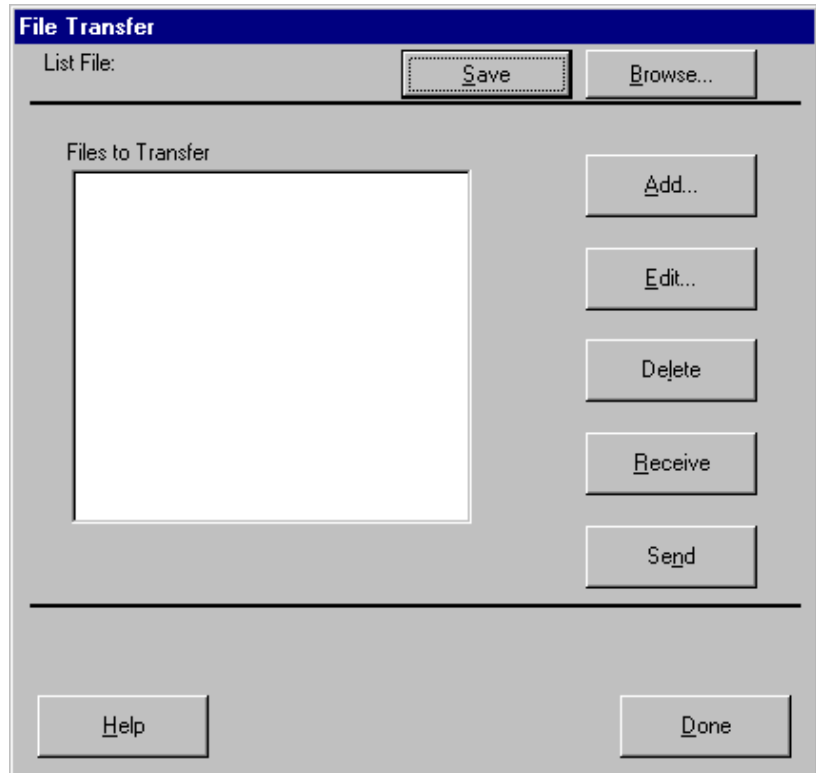
Baud Rate

Select the baud rate for serial communications between your PC and the Falcon. The default baud rate is 9600.

>> The File Transfer Window

When you select **Transfer Files** from the main menu, the File Transfer window appears. Use this window to select data files for transfer between the Falcon and your computer.

Figure 2-16: The File Transfer Window



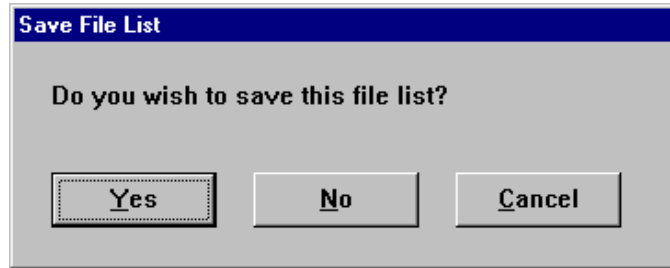
List File This field shows the name of the file-list file, if you have one selected.

NOTE: You can select files to transfer without using or creating a list file.

Save After making changes in the Files to Transfer list, you can select **Save** to save the revisions in the current file-list file (if any), in another existing file, or in a new file.

Browse Select **Browse** to use a different file-list file. An Open dialog box will appear. Use the dialog box to choose a file-list file from the Filelist folder. If you select Browse after making changes in the File Transfer window without saving the changes, the following prompt will appear:

Figure 2-17: The Prompt for Saving Changes to the Current File List



Yes Select this option to save the changes. A Save As dialog box will open. Use the dialog box to specify the location and name of the new file-list file.

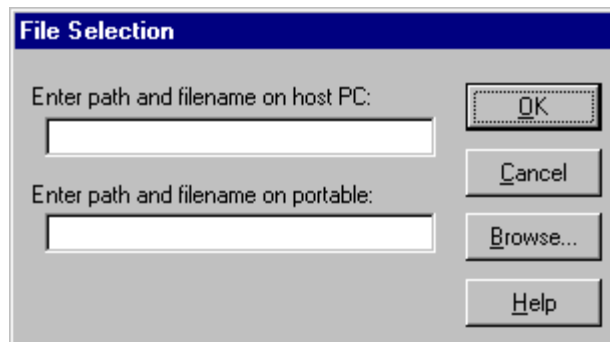
No Select this option to discard the changes.

Cancel Select this option to return to the File Transfer window without saving or discarding the changes.

Files to Transfer This field shows the files that will be included in the transfer between the Falcon and the PC.

Add Select **Add** to include additional data files in the transfer. The File Selection dialog box opens.

Figure 2-18: The File Selection Dialog Box for Adding a Data File



Enter path and filename on host PC Use this field to specify the location of the file on the PC.

Enter path and filename on portable Use this field to specify the location and name you want the transferred file to have on the Falcon.

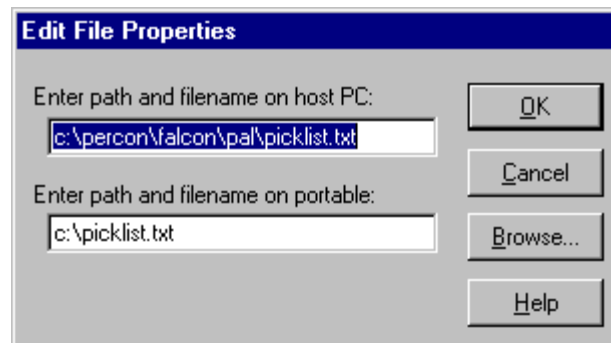
OK Select **OK** to return to the File Transfer window. The specified data file will appear in the Files to Transfer list.

Cancel Select **Cancel** to return to the File Transfer window without adding a file to the Files to Transfer list.

Browse Select **Browse** to view the files on your computer. An Open dialog box will appear. Use the dialog box to choose a data file to be included in the transfer.

Edit Select a file in the file list and then select **Edit** to change the source path or destination path for the file. The Edit File Properties dialog box opens. The fields and buttons in this dialog box are the same as in the File Selection dialog box above.

Figure 2-19: The Edit File Properties Dialog Box for Editing a Data File



- Delete** To delete a file from the list, select the file and then select **Delete**.
- Receive** Select **Receive** to begin a file transfer from the Falcon to the PC. When you select it, the configuration utility prepares files to be transferred to the PC and opens the Important dialog box (see “[Default](#)” on page [13](#)). Make sure your Falcon unit is properly connected to the serial port specified in the Comm Settings dialog box (see page [29](#)). Then run the `1d.bat` file on the Falcon, and select **OK** in the Important dialog box on the computer.
- Send** Select **Send** to begin a file transfer from the PC to the Falcon. The configuration utility prepares files to be transferred to the Falcon and opens the Important dialog box (see “[Default](#)” on page [13](#)). Make sure your Falcon unit is properly connected to the serial port specified in the Comm Settings dialog box (see page [29](#)). Then run the `1d.bat` file on the Falcon, and select **OK** in the Important dialog box on the computer.
- Done** Select **Done** when you are finished selecting and transferring files.

>> Chapter Three

Using XFER

XFER (**xfer.exe**) is a DOS utility that directs the serial transfer of ASCII and binary data between two computers. This chapter provides information for advanced users and system administrators who want to use XFER to transfer files between a Falcon and a PC.

Finding Out More: For basic information about XFER, see page [8](#).

CHAPTER CONTENTS

About XFER	<36>
Syntax and Parameters	<36>
The XFER_ARGS Environment	
Variable	<50>
Multiple-Option Blocks	<51>
The Modem-Initialization File	<53>
Format	
Keywords	
Sample Modem-Initialization File	
Performance	<56>
Error Codes	<57>

>> About XFER

XFER supports Xmodem and Zmodem transfers at speeds up to 115200 baud. Modem support is provided via an initialization file that specifies option settings for the modem. Option settings can be specified on the command line or with an environment variable called `XFER_ARGS`.

XFER supports RTS/CTS handshaking for Xmodem protocol and both XON/ XOFF and RTS/CTS handshaking for Zmodem protocol.

>> Syntax and Parameters

The command line syntax for XFER is as follows:

```
XFER [/option1 [/option2] . . .] filename(s)
```

You can use a slash (/) or a hyphen (-) to denote options, and you can use uppercase or lowercase letters for them. Options can be placed before or after filenames on the command line.

Basic options and their defaults are listed and described in table 3-1. The "Default" column indicates whether the option is used ("On") or ignored ("Off") if you do not include it in the command line. For options that have two or more possible values, the default value is given.

A page number for each option indicates where you can find a full explanation of it.

Table 3-1: XFER Parameters

Option	Description	Default	
		Xmodem	Zmodem
<i>filename(s)</i> (p. 38)	Identifies the file to be transferred or received.	None	None
<i>@file</i> (p. 39)	(Zmodem only) Specifies a response file consisting of two or more names of files to be transferred. Replace <i>file</i> with the name of the response file to use.	N/A	None
<i>#</i> (p. 40)	Specifies the communication port to use. Replace the # symbol with the desired setting: 1 = COM1 2 = COM2	1	1
<i>B#</i> (p. 41)	Specifies the baud rate. Replace the # symbol with the desired setting: 2400 4800 9600 19200 38400 57600 115200	19200	19200

table continues

Option	Description	Default	
		Xmodem	Zmodem
C# (p. 41)	(Zmodem only) Specifies how the retransmission of a file should be handled if a previous transfer was interrupted. Replace the # symbol with the desired setting: 1 = Never recover (start transmission from the beginning of the file) 2 = Follow sender (use the sender's crash-recovery options; ignore the receiver's) 3 = Always recover (send data from the point where transmission was interrupted)	N/A	1
D# (p. 42)	Specifies the number of seconds for XFER to wait for activity before canceling the transfer. Replace the # symbol with the desired number of seconds for the timeout delay. Acceptable values are 0 (no timeout) through 65,535.	60	60
E# (p. 42)	Specifies the maximum number of times XFER should attempt retransmission of a packet after an error occurs. If the final attempt fails, XFER aborts the transfer. Replace the # symbol with the desired maximum number of attempts. A setting of 0 allows for unlimited attempts.	0	30
F# (p. 42)	Specifies the type of flow control to use for data transfer. Replace the # symbol with the desired setting: 0 = No flow control 1 = XON/ XOFF (Zmodem only) 2 = RTS/ CTS	0	2
H or ? (p. 43)	Displays help for the XFER command.	None	None
Ifile (p. 43)	Specifies the modem-initialization file. Replace <i>file</i> with the name of the file to use.	modem.ini	modem.ini
Mcommand (p. 43)	Sends a command to the modem before beginning the data transfer. Replace <i>command</i> with one of the following: A = Auto answer ATstr = Send ATstring commands D#x = Dial memory #x (0-9) D#str = Dial phone number str	None	None
O# (p. 44)	Overwrites an existing file with a new file having the same name. Replace the # symbol with the desired setting: 1 = Overwrite if source is longer or newer 2 = Overwrite if CRCs don't match 3 = Append to existing file 4 = Always overwrite 5 = Overwrite if source is newer 6 = Overwrite if dates or lengths don't match 7 = Never overwrite	7	7
Q (p. 46)	Toggles quiet mode on and off. In quiet mode, only the filename and a "Transmitting..." or "Receiving..." message is displayed. When quiet mode is off, additional information is displayed. Each use of this option reverses the quiet-mode state.	Off	Off
R (p. 46)	Receives the specified file or files.	Off	On if no file is specified

table continues

Option	Description	Default	
		Xmodem	Zmodem
s# (p. 47)	(Zmodem only) Specifies how paths and names should be handled when sending files. Replace the # symbol with the desired setting: 0 = Strip paths 1 = Send paths 2 = Send new paths/filenames	N/A	0
T (p. 48)	Transmits the specified file or files.	On	On if file is specified
w# (p. 49)	Window size (Zmodem only): x = Multiple of 1024K 0 = Streaming	N/A	0
x (p. 50)	Uses Xmodem protocol for the transfer.		
z (p. 50)	Uses Zmodem protocol for the transfer.		

filename(s) Name(s) of the File(s) to Send or Receive

Xmodem Only one file can be transferred at a time. You can specify the entire path, including the drive letter. If you do not specify a path, the file is sent from or received into the current directory. If the directory or file doesn't exist on a send, or the directory doesn't exist on a receive, the transfer will fail, with a file-open error.

You must include the filename on the command line when receiving a file. If multiple files are specified for Xmodem, only the first file will be transferred. All other file names will be ignored.

Zmodem Zmodem allows up to 256 files to be transferred in one session. You can specify the entire path for each file, including the drive letter. If you do not specify a path, the file is sent from or received into the current directory. If the directory or file doesn't exist on a send, or the directory doesn't exist on a receive, the transfer will fail, with a file-open error. DOS accepts only 128 characters on the command line, but you can use a response file to get around this limitation. (See the `@file` option on page 39.)

When receiving, the filename does not need to be specified in Zmodem. Zmodem transfers the name of the file to be sent prior to sending the file. (See the `s` option on page 47 for information on controlling how Zmodem sends the filename.) Any filename specified on the receiving end will override the default name that is sent by the sender.

In case of multiple files being sent, a one-to-one correspondence is established. For example, if ten files are specified on the sender's side and five filenames are specified on the receiver's side, the first five files transferred will be received with the names specified on the receiver's command line, while the last five files will be received with the names specified on the sender's command line (assuming all the files exist). Each filename on the command line can include its own path.

XFER also recognizes the wildcard characters `?` and `*`. Normal DOS pattern-matching rules for these characters apply. Be careful about using wildcards when receiving: XFER expands the wildcards of the specified filenames and then searches for any matching patterns in the specified directory. If no matching files are found (in an empty directory, for instance), then no file names will be sent to the Zmodem receive procedures. This may or may not be the intended result. Also, DOS does not store files in alphabetical order, and so using `*.*` to both send and receive files may result in files being renamed because they are loaded in different orders. This can occur even if the file lists on both sides of the transfer are the same.

@file Response File (Zmodem only)

The `@file` option allows use of a response file for specifying filenames when conducting Zmodem transfers. This allows you to circumvent the DOS limitation of 128 characters on the command line when sending many files. (Response files do not work with Xmodem protocol.)

When using a response file, simply fill the file with the filenames you wish to send or receive. The filenames may include drives and paths. One filename should appear on each line of the response file. You can specify up to 256 files this way.

The following is a sample response file, named `response.txt`:

```
autoexec.bat
c:\programs\theprog.exe
c:\data\file1.dat
c:\data\file2.dat
c:\data\file3.dat
```

Not all the filenames need to be specified in the response file. For example, the following command would send seven files, including the five in `response.txt` above.

```
xfer /z c:\default\config.sys /@response.txt c:\data\file4.dat
```

In this case, each filename is read in as it appears on the command line. `c:\default\config.sys` would be the first file sent, the `autoexec.bat` file specified in the `response.txt` file would be the second file sent, and the `c:\data\file4.dat` file specified on the command line would be the seventh file sent.

You can specify multiple response files. Regardless of the number of response files you use, the 256-file limitation for a single transfer session is always in effect.

There is no option to undo the effects of a previously assigned response file. Therefore, if you include a response file in the `XFER_ARGS` environment variable (see page 50), there is no way to prevent XFER from trying to use those filenames during the transfer.

Response files can be used with the `s2` option (see page 47). In this case, the format for the response file would be:

```
autoexec.bat c:\autoexec.bat
c:\programs\theprog.exe c:\prog.exe
c:\data\file1.dat d:\data1
c:\data\file2.dat d:\data2
c:\data\file3.dat d:\data3
```

In this case, the first filename on a line specifies the file on the sender's side. The second filename on the line specifies the filename the sender will send to the receiver.

NOTE: *The receiver's filename is separated from the sender's filename by a single space. If more than one space is used, the additional spaces will be included as part of the receiver's filename. Because a space character is used as a delimiter, Windows 95 and Windows NT filenames with spaces as part of the filename will not be parsed correctly and should not be used. Placing quotes around the filename will not help in this situation.*

The `@file` option can be used only in Zmodem transfers. The `z` option must be specified before the `@file` option. If a response file is used with the `s2` option, the `s2` option must be specified prior to the `@file` option.

All other rules governing filenames are in affect when using response files.

COM Port

This option specifies the communication port to use for the file transfer. Replace the `#` symbol with the desired setting:

1 = COM1 2 = COM2

B# This option specifies the baud rate to be used for the file transfer. Replace the # symbol with the desired setting:

2400 4800 9600 19200 38400 57600 115200

C# *Crash Recovery (Zmodem only)*

Crash recovery refers to the ability of Zmodem protocol to detect that a file transfer was not completed and to attempt to recover at the point of failure (instead of recopying the entire file). For instance, if the communications link is severed during the transfer of a 10K file, with only 5K of data successfully transferred, crash recovery can be used to send the remaining 5K without resending the entire 10K file.

***NOTE:** This option is supported by Zmodem protocol only. Xmodem will never attempt to recover a file after a crash.*

Crash recovery uses the CRC of the sender's and receiver's files to ensure that the portion of the file that was successfully received in the first transfer is identical to that portion of the sender's copy of the file. If this is true, then the receiver instructs the sender to begin sending data beginning at the point in the file that corresponds to the end of the file on the receiver's side.

By default, crash recovery is turned off. To use the **D** option, replace the # symbol in the option with the desired setting:

- 1 - No crash recovery
- 2 - Follow sender
- 3 - Always use crash recovery

If you use the **C** option without including a number, then the 1 setting is assumed. An invalid setting will result in an error on the command line.

The **C** option can be used in conjunction with the **O** option. If you specify **C2** (follow sender) on the command line, the receiver will use the sender's crash and overwrite options. Any overwrite options local to the receiver will be ignored. This is useful if a default crash recovery command is specified in the **XFER_ARGS** environment variable (see page 50), and you wish to temporarily disable crash recovery for the current transfer.

D# Delay Before Terminating Transmission

This option allows a timeout to be specified. After the specified number of seconds with no characters received by either the sender or the receiver, the program will abort the transfer session.

NOTE: *For multiple file transfers in Zmodem, the entire session is aborted, not just the transfer of the current file.*

To use the D option, replace the # symbol with the desired number of seconds for the timeout delay. Acceptable values are 0 (no timeout) through 65,535; the default is 60. A value of 0 disables the timeout feature, meaning that the transfer session will never be aborted because of inactivity. If you use the D option without specifying a number, an error will occur.

E# Maximum Number of Errors Before Aborting Transfer

The E option allows you to specify the maximum number of times the receiver will request a retransmission of a packet before aborting the transfer session.

NOTE: *For multiple file transfers in Zmodem, the entire session is aborted, not just the transfer of the current file.*

For Xmodem, the default is 0, meaning that Xmodem will never be aborted because of errors. For Zmodem, the default is 30. A value of 0 disables the error-count tracking. If you use the E option without specifying a number, an error will occur.

F# Flow Control

This option allows you to specify what type of flow control to use for the file transfer.

Flow control is the ability of the receiver to detect that data is coming in faster than it can receive it and to issue a command to the sender instructing it to stop sending data. When the receiver is again able to handle new data, it issues another command instructing the sender to resume the transfer of data.

XFER supports two types of flow control: software (XON/XOFF) and hardware (RTS/CTS). Only one type can be used at a time. The sender and receiver should have this option enabled similarly.

NOTE: *XON/XOFF cannot be used with Xmodem protocol, because Xmodem protocol sends raw binary data and cannot distinguish between the XON/XOFF characters and file data. Attempting to specify XON/XOFF with Xmodem protocol will result in an error.*

To use the **F** option, replace the **#** symbol with the desired setting:

- 0 - No flow control
- 1 - XON/ XOFF (*Zmodem only*)
- 2 - RTS/ CTS

If you do not include the **F** option in the command line, no flow control is used. If you use the **F** option without specifying a number, it is assumed to mean **F0**, or no flow control. This is useful in the case where flow control is specified in the **XFER_ARGS** environment variable (see page [50](#)) but you wish to turn flow control off for the current transfer session.

An invalid setting will result in an error on the command line.

On a Falcon portable, RTS/CTS flow control should be used when transferring with Zmodem, especially if Zmodem is in streaming mode. In streaming mode, the Zmodem sender does not wait for acknowledgments to packets but continuously sends a stream of data. Often this can overwhelm the receiver, especially if it is receiving at high data rates or receiving to the **C** drive or to a PCMCIA ATA card, which writes data to disk much slower than the **D** drive.

H or ? Help

This option displays a brief description of the XFER syntax, options, and settings.

Ifile Modem-Initialization File

This option specifies the file that contains modem setup and control commands. See "[The Modem-Initialization File](#)" (beginning on page [53](#)) for complete information about the file.

Mcommand Modem Command

This option allows commands to be sent to the modem prior to the start of the file transfer. To use the **M** option, replace *command* with one of the following commands:

- A** = Auto answer
- ATstr** = Send ATstr commands
- D#x** = Dial memory #x (0-9)
- D#str** = Dial phone number str

Specifying `XFER /MA` instructs the modem to go into auto-answer mode. In this mode the modem will wait for the timeout period specified in the modem-initialization file for another modem to dial into it. On detection of a ring-in signal, XFER will answer the line and attempt to start a file transfer.

Specifying `XFER /ATstr` allows a string of AT commands to be sent to the modem. These commands will be sent to the modem after the initialization string from the modem-initialization file and can be used to specify a different configuration. Since only one modem command can be specified on the command line, any commands to auto-answer or dial must be included in the AT string.

There are two methods for dialing numbers. The first method uses the format `XFER /D#x`, where `x` is a number from 0 through 9. This number describes the directory number from the modem-initialization file that should be dialed for this transfer (see the `NUMBER#` keyword on page 55).

The second dialing method is to use the `XFER /D#str` format, where `str` is a dialing command to be sent directly to the modem. This string can be any recognizable modem phone number text, such as (555) 444-3333 or 5554443333.

The last `M` option on the command line or in the environment variable is the one used for controlling the modem. All earlier modem-command settings are ignored.

If you use the `M` option without specifying a command, the modem will be disabled for that transfer session. This is useful in the case where modem commands are specified in the `XFER_ARGS` environment variable (see page 50) but you wish to transfer files without a modem for the current transfer session.

O# Overwrite Existing File(s)

The `O` option specifies what action the receiver should take if the file being transferred already exists on the receiver's side. When using this option, replace the `#` symbol with the desired setting:

- 1 - Overwrite if source is longer or newer
- 2 - Overwrite if CRCs don't match
- 3 - Append to existing file
- 4 - Always overwrite
- 5 - Overwrite if source is newer
- 6 - Overwrite if dates or lengths don't match
- 7 - Never overwrite

By default, XFER operates in O7 mode; existing files are never overwritten. For Xmodem, the only valid options are O4 and O7. All other settings resolve to O7 (never overwrite). All settings for the O option are available with Zmodem protocol.

To remain compatible with previous versions of XFER, the O option used without a setting resolves to the O4 setting (always overwrite). This differs from most other XFER options, where specifying the option without a setting usually resolves to the default condition for that option. If the default condition is desired with the O option, then O7 must be used.

An invalid setting will result in an error on the command line.

NOTE: XFER no longer prompts you to specify whether a file should be overwritten in the default mode. If no overwrite option is specified, the file will never be overwritten. Under Xmodem, the session will abort. Under Zmodem, the file will be skipped and the next file will be transferred.

For the O6 setting under Zmodem, XFER does not set the date or time of a received file to match the original date or time of the sender's version of the file. Instead, the date and time are taken from the receiver's operating system when the file is received and created. Because of this, it is highly unlikely that a file transferred with XFER will have the exact date of the original file. Since the resolution of times for files under DOS is in two-second increments, it is unlikely the timestamps will be the same, even if the times on the receiver and sender are calibrated frequently. Also, DOS doesn't report the seconds field when displaying the modification time of a file during a file listing. So, two files may appear to have the same timestamp when in fact they don't. The tendency with the O6 option is for files to be overwritten, because times often do not match exactly.

The O3 setting is unique in that it causes the transferred file (binary or ASCII) to be appended to the end of the old file rather than overwritten.

Under Zmodem the O option can be used in conjunction with the C option. If you specify C2 (follow sender) on the command line, then the receiver will use the sender's crash and overwrite options. Any options local to the receiver will be ignored.

Q Quiet Mode

This option toggles quiet mode. When quiet mode is enabled, minimal information is displayed about the file transfer in progress. Generally, only the name of the file being transmitted is displayed. When not in quiet mode, XFER will display a copyright message, the filename, and file statistics, including a running count of bytes and errors.

Running XFER in quiet mode on the Falcon can have serious consequences for serial-transfer performance, especially at high speeds. At 115200 baud, transfers are four times faster if quiet mode is enabled on the Falcon than if it is disabled, owing to the delay inherent to writing a running byte count to the LCD screen.

Since the `Q` option is a toggle switch, each occurrence of it in the command line inverts the quiet-mode state. By default, XFER has quiet mode off. The first occurrence of the `Q` option turns quiet mode on, the next turns it back off, and so on. This is most useful if the option is specified in the `XFER_ARGS` environment variable (see page 50) to activate quiet mode by default but you wish to deactivate quiet mode for the current transfer session.

R Receive File(s)

This option specifies that files are to be received. With Xmodem protocol, the option must be specified whenever you wish to receive a file. The option can be included in the `XFER_ARGS` environment variable (see page 50), forcing Xmodem protocol to default to receiving instead of sending files.

With Zmodem, receive is the default if *no* file is specified on the command line (for example, `XFER /z`). The name of the file is sent by the transmitting side prior to the file being transferred, and the file is received with that path and name. If you wish to receive a file under Zmodem with a different name from the sender's original filename, you can use the `R` option to force Zmodem into receive mode and then specify a filename. (See the `filename(s)` option on page 38 for information on specifying filenames.) Using the `R` without a filename is the same as not using the option at all—the sender's filename (and path) is used.

NOTE: *The last transfer option, `R` or `r`, on the command line controls whether to transmit or receive for the session.*

S Specify Sender's Path-and-File Handling (*Zmodem only*)

The `s` option controls how the sender handles the paths and filenames of the files that it sends. This option is available only with Zmodem protocol. You must specify the `z` option prior to using the `s` option. You can use one of the following settings. The settings are described individually below.

- 0 - Strip paths
- 1 - Send paths
- 2 - Send new paths/filenames

If you do not include the `s` option in the command line, the default setting (0) is used. If you use the `s` option without specifying a setting (for example, `XFER /s`), it is assumed to mean `s0`. This is useful in the case where another option is specified in the `XFER_ARGS` environment variable (see page [50](#)) but you wish to strip the paths for the current transfer session.

An invalid setting will result in an error on the command line.

Strip Paths In this mode (the default), the sender will remove any paths that are specified with a filename before sending the filename to the receiver. The new filename will be transmitted to the receiver without any path information. The file will be stored by the receiver in the current directory.

Send Paths This option performs no processing on the filenames specified on the sender's side. Any drive and path information that is specified is transmitted to the receiver. If a path is specified and the path does not exist on the receiver's side, the transfer will abort with a failure.

Send New Paths/Filenames When the `s2` setting is used, all filenames are treated as filename pairs. This gives you the ability to specify dual filenames during a transfer. The first filename (optionally including a drive and path component) is the one that the sender uses to open the file. The second filename is used by the sender when it transmits the filename to the receiver. This gives you the ability to transfer a file and have it renamed on the receiver's side. It also gives you the ability to transfer a file from one directory on the sender's side to another directory on the receiver's side.

You may specify multiple filename pairs, which may be separated by other XFER options.

You must observe the following conventions when using the `s2` option. Failure to conform may result in errors.

- You must specify the `s2` command ahead of any filenames.
- When specifying dual filenames, you must separate the filenames by a single space. The first filename in the pair is what the sender uses, and the second filename is what the sender will transmit to the receiver. If you use more than one space, all spaces after the first will be included as part of the receiver's filename.
- You may specify drive and path components for either filename in a pair. However, you cannot specify *just* a drive and path; that is, the actual name of the file must be included in both filenames.

Correct examples:

```
C:\> xfer /zs2 file1 file2
C:\> xfer /zs2 c:\data\file1 d:\file2
C:\> xfer /zs2 c:\data\file1 d:\file1 c:\yourfile myfile
C:\> xfer /zs2 file1 data1 /b115200 file2 data2
```

Incorrect examples:

```
C:\> xfer /zs2 file1 /b115200 file2
C:\> xfer /zs2 file1   file2
C:\> xfer /zs2 file1[TAB]file2
C:\> xfer /z file1file2 /s2
C:\> xfer /zs2 file1d:\
```

In the second incorrect example, the additional spaces would be included in the receiver's filename. This is generally wrong, but it may be correct if that is the intended result. The third incorrect example demonstrates that the filenames must be separated by a single space. No other white-space character is valid as a delimiter. This mistake most often occurs when using response files (see page [39](#)).

NOTE: *If a filename is specified on the receiver's side, that filename will override whatever filename the sender transmits to the receiver.*

T Transmit File(s)

This option specifies that files are to be sent. For Xmodem protocol, this is the default; that is, typing `XFER foo` on the command line would send the file `foo`.

For Zmodem protocol, `transmit` is the default if you specify a file on the command line. So, for example, the command `XFER /Z foo` is equivalent to the command `XFER /Z /T foo`.

The **T** option is most useful if the **XFER_ARGS** environment variable (see page [50](#)) includes the **R** command but you want to send a file.

NOTE: *The last transfer option, **R** or **T**, on the command line controls whether to transmit or receive for the session.*

W Window Size (Zmodem only)

This option controls the use of windowing with Zmodem protocol. By default, Zmodem is a streaming protocol, and windowing is disabled. Streaming refers to the method by which Zmodem transfers data. In streaming mode, the sender forms 1K data packets and sends one packet after another in a continuous stream. The sender does not wait, and does not expect, to get acknowledgments from the receiver for any of the packets it sends. Instead, the sender assumes the file transfer is going normally, unless it gets an error indication from the receiver.

This works fine in robust, error-free environments. But the receiver can get lost or disconnected without the sender realizing it. Windowing provides a way in noisy environments to give some level of packet acknowledgment to Zmodem protocol. With windowing, the sender will send up to the number of bytes specified by the size of the window and then stop transmitting data until the receiver acknowledges that it has received all of the packets that have been sent. The sender then sends more packets, up to the size of the window, and so on. Since waiting for acknowledgments slows down transfer speeds, this method allows a dynamic compromise between speed and robustness by specifying the size of the window to use.

XFER allows you to specify the window size with any value from 0 to 65535. Use **w0** for streaming mode. For a 1K window, use **w1024**. Values below 1024 are used exactly as specified. Values above 1024 should be multiples of 1024 (that is, 1024, 2048, 4096, etc.). Values that are not multiples of 1024 are rounded up to the nearest multiple of 1024.

If you do not include the **w** option on the command line, streaming is used. If you use the option without specifying a setting, **w0** (streaming) is used by default. This is useful in the case where a window size is specified in the **XFER_ARGS** environment variable (see page [50](#)) but you wish to use streaming for the current transfer session. An invalid setting will result in an error during parsing.

NOTE: *Always place the **w** option after the **z** option, or a parsing error will occur. See the information about the **z** option on page [50](#) for more details.*

X Xmodem Transfer

The **x** option instructs XFER to transfer all files in that session using Xmodem protocol. Because this is the default file-transfer protocol, it does not normally need to be specified.

Z Zmodem Transfer

The **z** option instructs XFER to transfer all files in that session using Zmodem protocol. The **z** option should come before all other options that are valid for Zmodem only (such as **C#**). This is required because Xmodem is the default protocol for XFER, and option parsing will fail if an illegal setting is specified under Xmodem.

In addition, specifying the **z** option resets the Zmodem window size for that session, making it necessary for the **z** option to come before any use of the **w#** option.

***NOTE:** The last protocol option in the command line specifies the protocol that will be used for the file transfer; that is, Xmodem would be used for the transfer ordered by **XFER /Z /X foo**, and Zmodem would be used for the transfer ordered by **XFER /X /Z foo**.*

>> The XFER_ARGS Environment Variable

XFER recognizes the environment variable **XFER_ARGS**. This variable can be used to send options to XFER that are desired in almost every transfer session. You can even specify files in the **XFER_ARGS** environment. Parsing of the XFER command line starts with the **XFER_ARGS** variable and then continues through any other specified options or filenames.

You can use a single slash or hyphen to merge multiple options in the **XFER_ARGS** variable, or you can use a slash or hyphen for each option. The following example illustrates both cases:

```
C:\> set XFER_ARGS = /ZROB57600
C:\> set XFER_ARGS = /Z /R /O /B57600
```

For each command line in the example, files will be received (option **R**) using Zmodem protocol (**Z**) and a baud rate (**B**) of 57600 baud, and existing files will always be overwritten (**O**).

***NOTE:** You can use up to ten slashes or hyphens for options in the environment variable. See the following section, "[Multiple-Option Blocks](#)," for more details.*

Any of the parameters used in the example could be overwritten on the command line. For example, if you issued the following command:

```
C:\> xfer /x foo
```

XFER would use Xmodem for the file transfer instead of Zmodem.

There are two precautions for mixing options in the environment variable with options on the command line. The first is about options that are supported by Zmodem protocol only. In the following example:

```
C:\> set XFER_ARGS = /z /w4096 /c3  
C:\> xfer /x foo
```

the environment variable is set up to use Zmodem with a 4K window and crash recovery turned on. In the XFER command line, Xmodem protocol is specified. Since Xmodem doesn't support windows or crash recovery, both those options are ignored. Another, more troublesome scenario is the following:

```
C:\> set XFER_ARGS = /z /o5  
C:\> xfer /x foo
```

Here, XFER defaults to Zmodem from the environment variable, with an overwrite setting of 5 (overwrite if source is newer). However, in the XFER command line, Xmodem protocol is specified. The only overwrite settings that Xmodem supports are 04 (always overwrite) and 07 (never overwrite); all other 0 options resolve to 07 for Xmodem. If the file `foo` exists, the transfer will abort. In this case, you should either specify 04 on the command line or not use the `x` option.

NOTE: *If a filename is included in the `XFER_ARGS` environment variable, that file will always be the first file sent or received, and there is no way to negate that operation. For this reason, you should generally not include filenames in the environment variable.*

>> Multiple-Option Blocks

XFER supports multiple options after a single slash (/) or hyphen (-) on the command line or in the `XFER_ARGS` environment variable. Each of the following examples is a valid command line:

```
C:\> xfer /zrob115200 foo  
C:\> xfer /zro /d30 /b115200 foo  
C:\> xfer /zrw2 /o5c3f2 foo
```

There are limitations to the use of multiple parameters, however:

Any option can immediately follow another without a separate delimiter (/ or -) unless the preceding option has a multicharacter setting. The following options have multicharacter settings:

- B** (baud rate)
- D** (timeout)
- E** (error count)
- I** (modem-initialization file)
- M** (modem command)

The setting for each of these options *must* be followed by a space. Except for the **I** and **M** options, the option is ignored during parsing if there is no space following its setting. For the **I** option, any extra characters directly following the initialization-file filename will be treated as part of the filename. Any extra characters following the **M** option will be treated as part of the modem command; this may produce modem-command syntax errors.

Correct examples:

```
C:\> xfer /zrob115200 foo
C:\> xfer /zo5c3f2b115200 foo
C:\> xfer /z /mA foo
```

The first two command lines above are valid because the **/b115200** setting is always at the end of the multiple-option block. Also in the second example, the **O**, **C**, and **F** options all have single-digit settings.

Incorrect examples:

```
C:\> xfer /zb115200ro foo
C:\> xfer /z /mAro6 foo
```

In the first command line above, XFER would load and run. However, the **R** and **O** options would not be detected, because the **b115200** setting is not followed by a space. Therefore, XFER would try to send the file **foo**, not receive it. The second incorrect example is wrong because the **R** and **O6** parameters would be included as part of the modem command **A**.

Be careful when specifying the COM port in a multiple-option block. For example, the following command:

```
C:\> xfer /zo2 foo
```

sets the overwrite option to 2 (overwrite if CRCs don't match) but doesn't cause XFER to use COM port 2. The following commands could be used to do this:

```
C:\> xfer /z2o foo
C:\> xfer /z2o2 foo
C:\> xfer /zo22 foo
```

In the last example, the first 2 is treated as the setting for the O, and the second 2 is parsed as the COM port to use for the transfer.

>> The Modem-Initialization File

Format The modem-initialization file (*modem.ini*) contains keyword-and-value pairs, with each pair on a separate line. Each keyword is separated from its value by one or more spaces; tabs and other white-space characters are not valid. Keywords can be in any order within the file.

Lines beginning with a # character or with a space character are treated as comment lines and are ignored. The # character can also be used to embed comments within a line. Any characters after a # character in a line are ignored for that line.

***NOTE:** Individual lines cannot exceed 128 characters in length, including comments, or errors in parsing may occur.*

Keywords **ACCESS_DELAY**

This keyword specifies the number of milliseconds for XFER to wait for a response from the local modem when sending a command to the local modem. The default is 2000 ms (2 seconds).

CARRIER_TIMEOUT

This keyword specifies the number of seconds for XFER to wait for a carrier signal from the remote end before timing out. The default is 60 seconds.

COMPRESSION_STRINGS

This keyword specifies a list of response strings (expected responses from the modem) that indicate that a connection was made using data compression. Because modems often support various types of data compression, they may have more than one string indicating that compression is taking place. You can use the **COMPRESSION_STRING** keyword as many times as necessary to create the desired list. However, the combined length of all the values assigned to **COMPRESSION_STRING** cannot exceed 115 characters.

DIAL_METHOD

This keyword specifies how the modem attempts to dial phone numbers. This can be **T** for tone dial or **P** for pulse dial.

ECHO

This keyword indicates whether responses received from the modem should be displayed to the screen. Enable the keyword by setting the value string to **T** (for true); clear it by setting the value string to **F** (for false).

FAIL_STRINGS

This keyword is a list of response strings (expected responses from the modem) that indicate that the connection to the remote end failed. These strings contain text (such as **NO CARRIER** and **BUSY**) that the modem might return on an error. Using a configurable list of errors allows the serial-in/out modem run-time library to compensate for differences in messages between individual modems. You can use the **FAIL_STRING** keyword as many times as necessary to create the desired list. However, the combined length of all the values assigned to **FAIL_STRING** cannot exceed 115 characters.

HANDSHAKING

This keyword controls the type of handshaking the Falcon will perform with the modem. Generally, RTS/CTS handshaking is necessary to support the advanced functions of modems, such as compression and error-correcting protocols.

You can set this keyword to **N** for no handshaking, **X** for Xon/Xoff software flow control, or **R** for RTS/CTS hardware handshaking.

NOTE: The **HANDSHAKING** keyword in the modem-initialization file takes precedence over any flow-control options selected on the command line in XFER.

LOCKED_BAUDRATE

When enabled, this keyword indicates that the local modem is configured to communicate with the PC at a fixed baud rate, regardless of the speed of the connection between the modem and the remote end. This is the normal state of affairs for any modem that supports either data compression or error-correcting protocols. Having this keyword enabled implies that the modem is using hardware handshaking.

The baud rate used for a file transfer will be the baud rate set for XFER by the **B** option (the default is 19200 baud). If the **B** option is not set, the baud rate used is the baud rate the modem indicates it is using in the **CONNECT** message after connecting to the remote end.

Enable this keyword by setting the value string to **T** (for true). Clear it by setting the value string to **F** (for false).

NUMBER#

The **NUMBER#** keyword specifies a user-defined phone number. The value string can be any sequence of characters that defines a valid number for the modem to dial. Up to ten phone numbers may be specified, with each one assigned a keyword **NUMBER0** through **NUMBER9**. Any number may be assigned to a keyword (unless the number has already been assigned). You do not need to define all ten **NUMBER#** keywords, and they do not need to be defined in numerical order.

See the *Mcommand* section on page [43](#) for information on how to access the **NUMBER#** strings to dial when using XFER.

PROTOCOL_STRINGS

This keyword is a list of response strings (expected responses from the modem) that indicate that a connection was made using an error-correcting protocol. Since modems often support various error-correcting protocols, they may have more than one string indicating that such a protocol is being used. You can use the **PROTOCOL_STRING** keyword as many times as necessary to create the desired list. However, the combined length of all the values assigned to **PROTOCOL_STRING** cannot exceed 115 characters.

ZINIT_STRING

This keyword specifies the string to use to initialize the modem. The initialization-string value should include the preceding AT sequence for Hayes compatible modems, as the serial in/out modem routines do not attach these characters to the beginning of the initialization string prior to sending it to the modem.

Sample Modem-Initialization File

```
INIT_STRING      ATE1Q0V1X4&C1&D2S7=10s9=6s10=20S11=55S0=2
FAIL_STRING      NO CARRIER
FAIL_STRING      ERROR
FAIL_STRING      NO DIALTONE
FAIL_STRING      BUSY
FAIL_STRING      NO ANSWER

# This is a comment.
# So is this.
COMPRESSION_STRING CLASS 5
COMPRESSION_STRING V.42BIS
PROTOCOL_STRIN   LAPM
PROTOCOL_STRIN   ALT
DIAL_METHOD      T # (T)ONE or (P)ULSE dialing
CARRIER_TIMEOUT 60
ACCESS_DELAY     2000
LOCKED_BAUDRATE  T # (T)RUE or (F)ALSE
ECHO             T # (T)RUE or (F)ALSE
HANDSHAKING      R # (R)TSCTS, (X)ONXOFF, or (N)ONE
NUMBER0          9 555-1234
NUMBER9          (111) 555-9999
NUMBER5          (111) 555-7777
NUMBER1          9 555-4321
```

NOTE: The keywords are all case-sensitive and must be capitalized. For the **FAIL_STRINGS**, **COMPRESSION_STRINGS**, and **PROTOCOL_STRINGS** settings, the value string should also be capitalized. Any responses from the modem are converted to capital letters prior to being compared with the various substrings.

>> Performance

This section describes issues relating to serial performance on the Falcon. Although these issues are described in the context of XFER, they apply to serial transfers in general.

File-transfer speed is severely reduced when byte and error counts are displayed on the Falcon screen. Use the **Q** option in XFER to enable quiet mode and speed up transfers (see page [46](#)).

Performance also is affected when data is transferred to the **C** drive or to a PCMCIA ATA card. Both of these devices write data to the disk very slowly. Problems may arise as the disks become full and loading algorithms are executed to clean up space for new data. This can become particularly severe with Zmodem in streaming protocol, because the sender never waits for the receiver to write data to the disk. To avoid this problem, use Zmodem with either windowing or flow control enabled. Flow control is generally faster and permits the receiver to operate at its peak capacity.

Another factor in performance is the presence of other software running in the background. Card and socket services, for instance, use the timer interrupt for certain functions. Because card & socket services chain into this interrupt and execute some code with interrupts disabled, the system may become overburdened when running serial interrupts at high speed (greater than 19200 baud). In this case, overrun, parity, and framing errors become more common, sometimes resulting in an inability to send a file efficiently. If this occurs, a lower baud rate should be selected and flow control should be enabled.

>> Error Codes

XFER returns the following errors.

Value	Mnemonic	Description
0	OK	All files transferred OK.
1	ABORT_FILESSKIPPED	One or more files were skipped during a multifile download.
2	ABORT_TIMEOUT	The transfer was aborted because of a timeout.
3	ABORT_KEYPRESS	The abort key (ESC) was pressed.
4	ABORT_LINEERROR	A communications line error (e.g., OVER-RUN) occurred.
5	ABORT_FILEERROR	A read, write, open, close, or access error occurred on a file.
6	ABORT_FTPERROR	An invalid file-transfer protocol option was selected.
7	ABORT_CHECKERROR	A checksum or CRC error occurred.
8	ABORT_MEMORYERROR	An error occurred while trying to allocate memory from the heap.
9	ABORT_RECEIVERERROR	The receiver encountered a problem and requested to skip the file.
10	ABORT_COMERROR	An error occurred while opening the COM port.
11	ABORT_ARGERROR	An error in an argument (option setting) occurred on the command line.
20	ABORT_UNKNOWN	An unknown error occurred.
31	MODEM_ABORT_CONNECTION	XFER could not establish a connection with the remote end.
32	MODEM_ABORT_NORESPONSE	The modem did not respond.
33	MODEM_ABORT_INIFILE	The specified modem-initialization file could not be found.

>> Chapter Four

The Falcon 4-Slot Dock

This chapter provides advanced information about using the Falcon 4-Slot Dock. It is intended for use by system administrators and advanced users who need to modify the Falcon 4-Slot Dock's factory-configuration settings.

Finding Out More: For basic information about the Falcon 4-Slot Dock, see the Falcon DOS Portable Terminals **User's Guide**. For information about programming the Falcon 4-Slot Dock, see the Falcon DOS Portable Terminals **Programmer's Reference**.

CHAPTER CONTENTS

<i>Configuring the Falcon 4-Slot Dock</i>	<60>
<i>Baud Rate</i>	
<i>Operating Modes</i>	
<i>Basic Communications with the Falcon 4-Slot Dock</i>	<61>
<i>Loading 4SLOT.SYS</i>	
<i>Transferring Files with XFER</i>	

>> Configuring the Falcon 4-Slot Dock

Baud Rate The Falcon 4-Slot Dock automatically detects the baud rate you are operating at and reconfigures itself accordingly. It is capable of operating at the following baud rates:

- 9600
- 19200
- 38400
- 115200

The Falcon 4-Slot Dock is configured for 19200 baud at the factory. To change the default baud rate, complete the following steps:

1. Set your host communications package or program to 2400 baud. Send a single space to the dock network. You will not get any feedback from any of the Falcon 4-Slot Docks in the dock network.

***NOTE:** This step isn't always necessary, but it ensures that the Falcon 4-Slot Docks will detect a baud rate error as quickly as possible. It also eliminates concern about what baud rates the various docks are already in.*

2. Now set your host communications package or program to the desired baud rate and send a plus character (+) to the dock network. This is a synchronization character that the docks use to verify that they are correctly getting data from the host on a baud rate change.
3. To verify that the docks are correctly configured, send the **Request Unit IDs** command **DCRI** to the dock network.

***NOTE:** You can type in the command at a terminal communications program, send it out the COM port from an application, or even copy it to the COM port device driver under DOS. For complete information about sending commands to a Falcon 4-Slot Dock, see the Falcon DOS Portable Terminals **Programmer's Reference**.*

You should receive the unit IDs in brackets (i.e., **[Unit ID]**) from any portables in dock slots. For each empty slot in the dock network, the Falcon 4-Slot Dock will receive a blank Unit ID (**[]**). If a slot has a portable in it that does not have the Falcon 4-Slot Dock device driver loaded, that slot will return the reserved unit ID **[UNKNOWN]**.

If any problems occur with a dock, or if you are unable to verify that a dock is correctly configured, attach that dock directly to the host computer with an RS-232 cable. Then repeat the above steps with only the one dock attached to the host. The dock should send back four unit IDs, with unit IDs of [] for empty slots and [UNKNOWN] for slots that have portables without the Falcon 4-Slot Dock device driver loaded.

Operating Modes

To get the most benefit from the Falcon 4-Slot Dock, it may be necessary to embed functionality into your portable or host application that can detect and interact with the Falcon 4-Slot Dock to assign access to the proper portable.

The Falcon 4-Slot Dock network can operate in either of two modes:

- In *terminal-demand mode* (the default), the Falcon 4-Slot Dock network grants access to the communications line to individual portables on a priority basis.
- In *host-controlled mode*, the host controls which portable gets access to the communications line at any time.

For information about setting and using these modes, see the *Falcon DOS Portable Terminals Programmer's Reference*.

>> Basic Communications with the Falcon 4-Slot Dock

Loading 4SLOT.SYS

The device driver `4slot.sys` is required on the portable to access the additional functionality of the Falcon 4-Slot Dock. The exception to this is that the `xfer.exe` program has limited ability to access the Falcon 4-Slot Dock network when the `4slot.sys` driver is not loaded. This allows you to use the Falcon 4-Slot Dock network to download the Falcon 4-Slot Dock device driver to any portable that doesn't already have it.

The Falcon Configuration Utility can be used to transfer the `4slot.sys` driver to the portable and modify the `config.sys` file for you. First you must install the Falcon 4-Slot Dock software to your computer. (You should also install the Falcon Configuration Utility at this point, if you haven't already done so.)

Once the Falcon 4-Slot Dock software is installed, you are ready to download the Falcon 4-Slot Dock configuration to the Falcon. In the Falcon Configuration Utility on the host, use the Custom button and select `4slot.cfg` as the configuration file to load.

NOTE: For complete information about the Falcon Custom Configuration Utility, see [chapter 2](#).

Transferring Files with XFER

You can transfer files with the XFER utility on the portable. XFER has the ability to automatically request access to the communications line from the Falcon 4-Slot Dock, whether the Falcon 4-Slot Dock device driver is loaded or not.

To use XFER to transfer a file to the portable, place the portable in a slot on the Falcon 4-Slot Dock. On the portable, type in the command to receive a file, which is typically the following:

```
C:\> xfer /r filename.txt
```

On the host, type the following at a DOS prompt:

```
C:\> xfer filename.txt
```

Once you execute the XFER command on the portable, the portable requests access to the communications line from the Falcon 4-Slot Dock. When access is given to the portable, the COM LED for that slot will come on with a green light. When you execute the XFER command on the host, XFER will transfer the file from the host to the portable. When XFER exits on the portable, the COM LED goes out, indicating that the portable has relinquished access to the communications line.

NOTE: XFER's default protocol is Xmodem, but it also supports Zmodem. When running XFER with the Falcon 4-Slot Dock, you cannot use hardware flow control. For Zmodem, the default flow control is XON/ XOFF; for Xmodem, the default is no flow control. This restriction holds for any communications done with the Falcon 4-Slot Dock.

For complete information about the XFER utility, see [chapter 3](#).

>> Chapter Five

Default Disk Drive Organization and Unit Configurations

This chapter details the default contents of the disk drives and the organization of system software on the Falcon.

CHAPTER CONTENTS

<i>File System Format</i>	<64>
<i>Overview</i>	
<i>Drive A</i>	
<i>Drive B</i>	
<i>Drive C</i>	
<i>Drive D</i>	
<i>Drive E</i>	
<i>System Software</i>	<69>
<i>BIOS and DOS</i>	
<i>Device Drivers and Utilities</i>	
<i>PC Card and RF Networking Software</i>	
<i>Unit Configurations</i>	<70>
<i>Default Configuration</i>	
<i>I/O PC Card Support</i>	

>> *File System Format*

Overview The Falcon's default file system format is as follows:

Drive **A** is a read-only flash drive. The contents of this drive cannot be changed.

Drive **B** is a read-only flash drive that you can update by using the core update program, `corexfer.exe`, from a host machine. (Normal operation will not require updating of this drive.)

Drive **C** is a resident flash disk drive that is accessed with the `flashdisk.sys` device driver. You can use this drive to store applications and data.

Drive **D** is a RAM disk accessed with the `vdisk.sys` device driver. This drive can be used to store applications as well as data. However, you should take great care in selecting files for this drive. As with any RAM drive, its contents will be lost if power is removed.

Additional drives are mapped to ATA flash PC cards or to network drives via radio frequency (RF).

This organization provides an easy-to-use, extensible system that allows a high degree of performance, usability, and customization. The following sections more fully describe the contents and intended uses of each of the Falcon's drives.

Drive A Drive **A** is a disk image that is located in flash along with the BIOS and ROM-DOS. A read-only drive, it is intended to remain secure throughout the life of the unit. Drive **A** contains the following files:

- `autoexec.bat`, the first file in the startup sequence
- `command.com`, the DOS shell
- `config.saf`, the first file in the safe-boot sequence
- `config.sys`, the first file in the boot sequence
- `rev.com`, the utility that reports the firmware revision

The `config.sys` and `autoexec.bat` files contain only basic commands that the unit needs to run correctly (see figures [5-1](#) and [5-2](#)). Both files chain to their respective counterparts on the B drive, as described in the next section.

The `config.saf` file (see figure [5-3](#)) is used in the safe-boot sequence. (For more information about the `config.saf` file, see page [66](#). For information about the safe boot, see page [75](#).)

Figure 5-1: CONFIG.SYS File on Drive A

```
REM *****  
REM Initial CONFIG.SYS file for starting the system.  
REM *****  
  
NEWFILE = B:\CONFIG.SYS
```

Figure 5-2: AUTOEXEC.BAT File on Drive A

```
@echo off  
REM *****  
REM Initial AUTOEXEC.BAT file for starting the system.  
REM *****  
VER  
PATH=A:\  
B:\AUTOEXEC.BAT
```

Figure 5-3: CONFIG.SAF File on Drive A

```
REM *****  
REM Initial CONFIG.SYS for safe boot.  
REM *****  
NEWFILE = B:\CONFIG.SAF
```

Drive B Drive **B** is a disk image that is located in flash. A read-only drive, it is intended to be used to store mandatory utilities and initialize the boot process. During normal use this drive will not be updated. Drive **B** includes the following files:

- `autoexec.bat`, the second file in the startup sequence
- `cfgdev.sys`, the system-parameter-configuration device driver
- `config.saf`, the second file in the safe-boot sequence
- `config.sys`, the second file in the boot sequence
- `coreupd.com`, the utility that updates the firmware image
- `decode.sys`, the bar-code-decoding device driver
- `flashdisk.sys`, the resident flash disk device driver
- `format.com`, the disk-reformatting utility
- `functest.com`, the functional test for system components
- `ld.bat`, the batch file for `response.bat` processing
- `lock.com`, the utility for locking or unlocking the resident flash disk and RAM disk
- `organize.com`, the utility for reorganizing flash disk data
- `pm.com`, the power-management utility
- `vdisk.sys`, the RAM-disk device driver
- `xfer.exe`, the serial-transfer utility

The `config.sys` file (see figure 5-4) will load the `flashdisk.sys` driver to create the **C** drive and the `vdisk.sys` driver to create the **D** drive. It will then load `decode.sys` and `cfgdev.sys` before chaining to its counterpart on drive **C**. The `autoexec.bat` file (see figure 5-5) will chain to its counterpart on drive **C**, except during a safe boot.

The purpose of the `config.saf` file (see figure 5-6) is to provide a minimal boot configuration that preserves the flash and RAM drives. When the safe-boot sequence is initiated (see page 75), the `config.saf` file is processed instead of the `config.sys` file. This prevents chaining into the drive-**C** initialization files by omitting the `NEWFILE = C:\CONFIG.SYS` command. It also eliminates the loading of the `decode.sys` and `cfgdev.sys` drivers.

Figure 5-4: CONFIG.SYS File on Drive B

```
REM *****  
REM Standard CONFIG.SYS for building system.  
REM *****  
  
REM *****  
REM Create flash and RAM drives  
REM *****  
DEVICE = B:\FLASHDSK.SYS 4096  
DEVICE = B:=VDISK.SYS 4096 /e  
  
REM *****  
REM Load decode/configuration drivers  
REM *****  
DEVICE = B:\DECODE.SYS  
DEVICE = B:\CFGDEV.SYS  
NEWFILE = C:\CONFIG.SYS
```

Figure 5-5: AUTOEXEC.BAT File on Drive B

```
@echo off  
REM *****  
REM Standard AUTOEXEC.BAT for starting system.  
REM *****  
PATH=%PATH%;B:\  
C:  
IF EXIST A:\CONFIG.SAF IF EXIST C:\AUTOEXEC.BAT C:\AUTOEXEC.BAT
```

Figure 5-6: CONFIG.SAF File on Drive B

```
REM *****  
REM Standard CONFIG.SYS for safe boot.  
REM *****  
  
REM *****  
REM Recover flash and RAM drives  
REM *****  
DEVICE = B:\FLASHDSK.SYS 4096  
DEVICE = B:\VDISK.SYS 4096 /e
```

Drive C Drive C is a resident flash disk drive that is accessed via the `flashdsk.sys` device driver. By default, this drive is configured to have full read and write access. The default `config.sys` and `autoexec.bat` files reside here. You can modify these files to customize the system for an application. They will be called after the `config.sys` and `autoexec.bat` files on drives A and B are processed.

The Falcon Configuration Utility gives you the ability to customize what software tools and utilities are placed on the unit. By default, the utility installs this software to the C drive. If the unit is intended to be used with PC cards, then the Phoenix PC card drivers are installed in the PCM subdirectory on the C drive, and the proper entries will be transferred into the `config.sys` and `autoexec.bat` files. The exact configuration of files and the contents of the `config.sys` and `autoexec.bat` files will change according to the configuration of the unit.

For RF capability, additional files for network connectivity would be required. These might include a Telnet program or peer-to-peer networking tools, depending on what you choose to install.

If something happens to a unit and data integrity becomes questionable, use the CHKDSK utility to detect and correct errors on drive C. (`chkdsk.com` does not come installed on the Falcon. You can download it using XFER or the Falcon Configuration Utility.) You can also use `organize.com` to recover unused sectors (see page [86](#)).

Drive D Drive **D** is a RAM disk drive. By default, this drive resides in extended memory and is sized 1MB less than the total RAM in the Falcon unit (either 1MB or 3MB).

Drive **D** can be used for short-term data storage. Programs that need to be loaded into memory and then removed from memory quickly can also be located there. The drive can also be used for scratch disk space or temporary files.

The Falcon preserves the data on drive **D** between warm or cold boots by checking for an existing RAM disk. However, only minimal checking is performed on any disk that is found; if something happens to a unit and data integrity becomes questionable, you should use the CHKDSK utility to detect and correct errors on drive **D**.

NOTE: `chkdsk.com` *does not come installed on the Falcon. You can download it using XFER or the Falcon Configuration Utility.*

Drive E Drive **E** is the ATA flash card. The PC card looks like a hard disk drive to the operating system and the user. More flexible than flash disk drive **C**, it can be used for safer and more permanent bulk storage of batch data than the RAM disk (drive **D**). This drive exists only on systems configured to use ATA flash cards.

Drive **E** could also be a peer-to-peer or client-server network drive that is accessed through an RF or Ethernet network card link. This option allows the developer to make many network drives (drives **E**, **F**, etc.) available to applications.

>> System Software

BIOS and DOS The Falcon uses a modified version of General Software's BIOS with Datalight's ROM-DOS. Both products are burned into system flash in a single 256K image. The **A** drive is included in the image, physically addressed just below the BIOS image.

ROM-DOS uses the Datalight `command.com` processor. This processor is fully Microsoft 6.2 compatible, except that it occupies about half the space in memory. In addition, drive **C** contains Datalight DOS files in a DOS subdirectory.

Device Drivers and Utilities

The following device drivers are always present on the Falcon's B drive:

- `cfgdev.sys`
- `decode.sys`
- `flashdsk.sys`
- `vdisk.sys`

The following utilities also are always present on drive B:

- `coreupd.com`
- `format.com`
- `functest.com`
- `lock.com`
- `organize.com`
- `pm.com`
- `xfer.exe`

See page [66](#) for descriptions of these drivers and utilities.

PC Card and RF Networking Software

PC card device drivers and utilities as well as RF networking solutions can be installed to drive C (the flash drive) or drive D (the RAM drive). You can use the Falcon Configuration Utility to download the necessary files to the Falcon.

>> Unit Configurations

Several configurations are possible for the system software on the unit, depending upon the type of hardware that is to be supported. Two standard configurations are listed below, along with directory structures and default `config.sys` and `autoexec.bat` file descriptions.

Default Configuration

The configuration created by `default.cfg` in the Falcon Configuration Utility will create a C drive that contains `palpro.exe`, part of the Percon Portable Applications Library (PAL). The data files will be stored on drive D. PC card drivers able to support ATA cards will be loaded.

Figure 5-7: CONFIG.SYS File on Drive C for Default Configuration

```

FILES = 30

REM -----
REM --- Phoenix Card and Socket services for accessing PC Cards
REM -----
device = c:\pcm\cnfignam.exe /NORMAL
device = c:\pcm\pcmssit.exe
device = c:\pcm\pcmcs.exe
device = c:\pcm\pcmata.sys

```

Figure 5-8: AUTOEXEC.BAT File on Drive C for Default Configuration

```

REM -----
REM --- The following lines set up default parameters for some
REM --- environment variables. These lines may be modified or
REM --- overridden in the USER section below.
REM -----
set prompt=$p$g
set dircmd=/ogn /p

REM -----
REM --- The following section is for customized user entries.
REM -----
REM --- Insert user-specific options and commands here.
REM -----

REM -----
REM --- The following lines add system components to the PATH
REM --- and run the main application executable, if one was
REM --- specified.
REM -----
IF EXIST c:\bparams.ini copy c:\bparams.ini PARAMS
set path=c:\;c:\dos;%path%
c:
cd\
upgstart

```

I/O PC Card Support

The Falcon Configuration Utility has an option for support of I/O cards. This option will download the Phoenix Super Client Driver (`pcmscd.exe`) to the C:\PCM subdirectory on the Falcon. It will also add the command for loading of this driver to the `config.sys` file. The `config.sys` and `autoexec.bat` files below were created by eliminating ATA card support and adding I/O card support to `default.cfg`.

Default Disk Drive Organization and Unit Configurations

Figure 5-9: CONFIG.SYS File on Drive C for I/O Card Support

```
FILES = 30

REM -----
REM --- Phoenix Card and Socket services for accessing PC Cards
REM -----
device = c:\pcm\cnfignam.exe /NORMAL
device = c:\pcm\pcmsst.exe
device = c:\pcm\pcmcs.exe
device = c:\pcm\pcmscd.exe
```

Figure 5-10: AUTOEXEC.BAT File on Drive C for I/O Card Support

```
REM -----
REM --- The following lines set up default parameters for some
REM --- environment variables. These lines may be modified or
REM --- overridden in the USER section below.
REM -----
set prompt=$p$g
set dircmd=/ogn /p

REM -----
REM --- The following section is for customized user entries.
REM -----
REM --- Insert user-specific options and commands here.
REM -----

REM -----
REM --- The following lines adds system components to the PATH
REM --- and runs the main application executable, if one was
REM --- specified.
REM -----
IF EXIST c:\bparams.ini copy c:\bparams.ini PARAMS
set path=c:\;c:\dos;%path%
c:
cd\
upgstart
```

>> Chapter Six

Resetting the Falcon

One of the critical features of any portable data-collection device is its ability to protect against accidental data loss. In the event a Falcon unit becomes “locked up” or otherwise in a state in which data can no longer be processed, you can use any of four methods to reset the unit. This chapter describes these methods and provides guidelines for avoiding data loss.

CHAPTER CONTENTS

<i>Warm Boot</i>	<74>
<i>Cold Boot</i>	<74>
<i>Safe Boot</i>	<75>
<i>Hardware Reset</i>	<76>
<i>Recommended Programming Practices</i>	<77>

>> Warm Boot

The warm boot is one of two software methods for resetting a Falcon unit. It is analogous to the **CTRL-ALT-DEL** key sequence available with IBM-compatible PCs, and the same key sequence is used to force a warm boot on a Falcon unit. This method of resetting a unit should be used first to attempt to bring it back to a usable state.

Since a warm boot can be attempted only from a unit that has been turned on, certain assumptions are made. For example, the rigorous hardware tests that are performed as part of the cold-reset boot sequence are not all necessary. This means the unit restarts faster with a warm boot than with a cold boot.

During a warm boot, data written to the RAM drive (D), the flash drive (C), or an ATA flash PC card will remain intact. However, if an application is running on a unit before a warm boot, its state cannot be restored.

NOTE: *You can disable the **CTRL-ALT-DEL** key sequence, making it difficult for an end user to reset the unit.*

>> Cold Boot

The second software method for resetting a Falcon unit is the cold boot. A cold boot should be used only if a warm boot is unsuccessful. A cold boot is analogous to the results of pressing the Reset button on the front of many desktop PCs or to turning the power off and then on.

To perform a cold boot on a Falcon unit, first turn it off. Then press and hold the ALT and FN 2 buttons while pressing the power button. Release all three buttons simultaneously, and the unit will reset.

Simply removing and inserting the battery pack will not force a cold boot, because the backup battery can provide minimal power to sustain the unit's operating state. The exception to this is when the backup battery is drained of all power, a condition that you should not let happen.

When a cold boot is performed, all transient data is lost, along with the state of any application that was running on the unit. Data written to the RAM drive (D), the flash drive (C), or an ATA flash PC card will remain intact. If a cold boot happens because of a dead backup battery, data written to a RAM drive may not be maintained.

A cold boot begins with a retesting of the hardware and then follows with the DOS boot sequence. The unit reboots DOS under the same conditions you would expect from a desktop PC: Device drivers included in the `config.sys` file and applications included in the `autoexec.bat` file are automatically loaded as part of the boot process. Any other programs that were loaded before the cold boot will not be reloaded automatically.

>> Safe Boot

During the DOS boot sequence, control is passed sequentially to the `config.sys` files on drives **A**, **B**, and **C**. A similar process is followed for the `autoexec.bat` files on each drive. (See chapter one for more information.) It is possible for a driver or application loading from the `config.sys` or `autoexec.bat` file on drive **C** to hang the system. A cold or warm boot of the unit will not directly correct the problem, because the troublesome program will be started again as part of the boot sequence.

What is required is a method by which the `config.sys` or `autoexec.bat` file on drive **C** and the loading of the `cfgdev.sys` and `decode.sys` drivers can be taken out of the normal boot sequence. DOS can help some (with the **CTRL-C**, **F5**, and **F8** escape sequences).

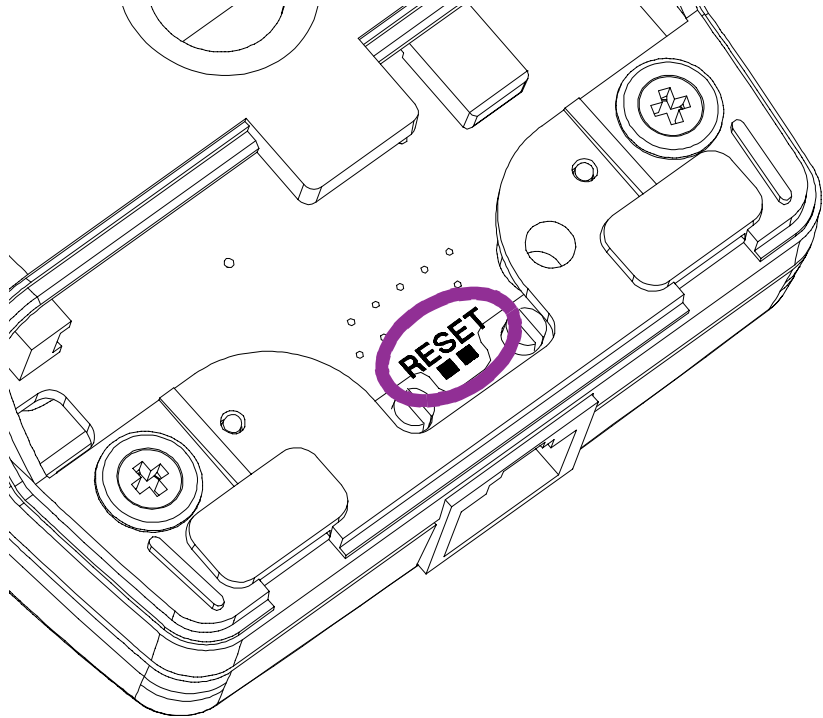
The Falcon provides an alternative method that is less accessible to the end user. To bypass the `config.sys` and `autoexec.bat` files on drive **C**, reboot the unit (with either a cold or warm boot), and when the message "wait..." appears on the display, immediately press the **ESC** and **DEL** keys at the same time. This causes alternative `config.sys` and `autoexec.bat` processing to take place on drives **A** and **B** without chaining into the files on drive **C**. The unit indicates safe-boot mode by emitting a low tone followed by a higher-pitched tone. The flash drive (**C**) and the RAM drive (**D**) will be preserved, and the boot sequence will place the unit at the `C:\>` prompt.

NOTE: *If you do not hear the two tones indicating the safe-boot sequence, you probably did not press the **ESC** and **DEL** keys quickly enough. Reboot the Falcon and press the two keys simultaneously as soon as you see the "wait..." message on the display.*

>> *Hardware Reset*

In the extremely rare situation where none of the rebooting methods is successful, you can use a reset mechanism that is located under the PC card slot cover (see figure [6-1](#)). Touch a metal device (such as a paper clip) to the two contacts on the reset mechanism. This will cause the Falcon to begin a cold boot.

Figure 6-1: Location of the Hardware Reset Mechanism



>> Recommended Programming Practices

The programming methodologies used to develop well-behaved DOS applications for a desktop PC also apply to creating applications for the Falcon. Some of these are more critical to success on a Falcon than on a PC, because of the hardware environment of the Falcon. The following are some important guidelines for programming Falcon applications:

- ❑ **Always check for reported error conditions.** Many of the BIOS and DOS service routines are useful only if the calling application checks for any flagged error conditions that signal a failure to comply with the original request. For example, not enough available memory or lack of disk storage space must be dealt with immediately; otherwise data may be lost.
- ❑ **Carefully choose how data will be stored.** A Falcon unit has a number of data-storage options: RAM drive, flash drive, ATA flash PC card, remote storage via RF, and other PC card possibilities. You should be aware of the strengths and limitations of each storage method and plan to store critical and temporary data in appropriate locations to maximize both system performance and data security.
- ❑ **Disable the CTRL-ALT-DEL key combination.** The Falcon allows for easy rebooting during the development process. To prevent the end user from resetting the unit in the field, disable the CTRL-ALT-DEL key combination. You can accomplish this by using bar code programming (see appendix C) or by calling an API function (see the *Falcon DOS Portable Terminals Programmer's Reference*).

>> Chapter Seven

Using PC Cards

PC card support expands the feature set of the Falcon to include:

- Supplemental storage capacity through ATA flash cards
- Modem support
- LAN connectivity
- Radio frequency (RF) communications

Falcon supports Type 1 and Type 2 PC cards. Only one card can be installed in the unit at any one time.

This chapter describes the use of PC cards with the unit. It does not provide details about specific cards, such as communications protocols for fax/modem cards. This type of information should be available from the PC card vendor.

Finding Out More: For information about opening the PC card slot cover, inserting and removing PC cards, and card recognition, see the Falcon DOS Portable Terminals User's Guide.

CHAPTER CONTENTS

Overview	<80>
Configuring Your PC Card	<80>
<i>PhoenixCARD Manager Plus (PCM+)</i>	
<i>ATA Cards</i>	
<i>Fax/Modem Cards</i>	
<i>LAN Cards</i>	
<i>RF Cards</i>	

>> Overview

The primary intended uses for PC cards with the Falcon are expanded storage space and radio frequency (RF) communications. Using ATA flash memory cards enables you to configure a Falcon for a theoretical maximum of 64MB of memory. The card appears as a hard drive to the user. RF communications cards allow you to connect the portable Falcon unit to a traditional network and to access shared network resources in a transparent fashion. The Falcon can also use network and fax/modem cards.

Use the Falcon Configuration Utility to download the appropriate drivers for ATA and generic I/O cards.

>> Configuring Your PC Card

PhoenixCARD Manager Plus (PCM+)

To attain compatibility with as many PC cards as possible, the Falcon utilizes card and socket services supplied by Phoenix Technologies. These card and socket services are implemented as installable device drivers and are loaded using the `DEVICE=` command in the `config.sys` file.

Table 7-1 identifies the PhoenixCARD Manager Plus (PCM+) drivers needed for each type of card. Information about these drivers follows the table. Additional drivers or application programs may be required for some cards, particularly RF, LAN, and fax/modem cards.

Table 7-1: PCM+ Drivers Required for PC Cards

PCM+ Driver	ATA	Fax/Modem	LAN	RF
<code>cnfignam.exe</code>	X	X	X	X
<code>pcm.ini</code>	X	X	X	X
<code>pcmata.sys</code>	X			
<code>pcmcs.exe</code>	X	X	X	X
<code>pcmscd.exe</code>		X	<i>See below.</i>	
<code>pcms.exe</code>	X	X	X	X

- `cnfignam.exe` identifies the section in the PCM+ configuration file (`pcm.ini`) that should be loaded as the default.
- `pcm.ini` is the configuration file for the PC cards.

- `pcmata.sys` is an ATA-IDE device driver. It communicates with the `pcmcs.exe` driver to allow the system to access ATA-configured PC cards.
- `pcmcs.exe` is the card-services driver. Required for all PC cards, it provides an interface between applications and the socket services. It also manages resource allocations for the PC cards.
- `pcmscd.exe` is referred to as the superclient driver. It uses the `pcm.ini` file to configure and request system resources for generic PC cards as well as specific cards that have been included in the `pcm.ini` file. Some PC cards will have specific client drivers to handle configuration; `pcmscd.exe` should not be used with those cards.
- `pcmss.exe` is the socket-services driver. Required for all PC cards, it provides an interface to the PC card hardware.

ATA Cards

ATA cards are flash memory cards that utilize the IDE interface. This allows the flash to appear as a hard drive. Any ATA PC card should work on the Falcon unit with the appropriate drivers installed. The following ATA cards are approved for use on the Falcon:

- 2.6MB SunDisk Seagate FlashCard, Model ST72P5
- 4MB SanDisk PCMCIA PC CARD ATA
- 8MB SanDisk PCMCIA PC CARD ATA

Additional cards may be approved. Contact your Percon representative for information.

Fax/Modem Cards

The Falcon should recognize any generic fax/modem PC card. Fax/modem PC cards are configured to use COM2, I/O address 2F8h, and IRQ 3. The following fax/modem cards are approved for use on the Falcon:

- US Robotics Sportster 28.8 Faxmodem PC card with DataView Connector
- Xircom CreditCard Modem 28.8

NOTE: *The Falcon does not support fax/modem cards with XJack connectors.*

LAN Cards

LAN cards require assignment of an I/O window, a memory window, and an IRQ. If the LAN card will be configured by the super client driver (`pcmscd.exe`), additional entries to the `pcm.ini` file will be necessary. Many LAN cards come with their own resource-management system and do not require the `pcmscd.exe` driver.

NOTE: *Because of the Falcon's memory-handling properties, some LAN cards may not work. Please contact Percon before purchasing a LAN card.*

RF Cards

Because of the special requirements of RF environments, Percon is distributing Falcon RF units to qualified customers only. Please contact your Percon representative to see if you are qualified.

>> Chapter Eight

System Utilities

This chapter describes the utilities that are designed to be run on a Falcon unit. For each device driver or executable file, a brief description is given, followed by any command line parameters and their meanings. Parameters may be supplied using either uppercase or lowercase letters, and the hyphen (-) may be substituted for the forward slash (/).

CHAPTER CONTENTS

<i>CFGDEV.SYS</i>	<84>
<i>DECODE.SYS</i>	<85>
<i>FLASHDSK.SYS</i>	<85>
<i>FORMAT.COM</i>	<85>
<i>LOCK.COM</i>	<86>
<i>ORGANIZE.COM</i>	<86>
<i>PM.COM</i>	<87>
<i>VDISK.SYS</i>	<88>
<i>XFER.EXE</i>	<88>

>> CFGDEV.SYS

The `cfgdev.sys` file is located on drive B of the Falcon unit. When `cfgdev.sys` is loaded by `b:\config.sys`, a logical character-based device called PARAMS is created. This device receives programming strings that modify the configuration parameters that control the behavior of several system components. There are three ways you can send programming strings to the PARAMS device:

- ❑ **Bar Code Scanning** Special bar code labels beginning with `+$-$-` and ending with `EE` can be scanned using any bar code scanning device. Appendix C contains many useful labels.
- ❑ **File Copy** Files containing the bar code programming strings can be created and copied to the PARAMS device. The Falcon Configuration Utility creates a file called `bparams.ini` that contains all of the configuration settings. The `autoexec.bat` file on the Falcon's drive C contains the following command:

```
EXIST c:\bparams.ini copy c:\bparams.ini PARAMS
```

which copies the `bparams.ini` file to PARAMS. You may also create your own file containing configuration settings.

Example: Use the D2 defaults with the `CTRL-ALT-DEL` key sequence disabled:

1. Create a `myparams.ini` file with the following text:

```
+$-$-D2E00EE
```

D2 sets the D2 defaults, E00 disables the `CTRL-ALT-DEL` key combination.

2. Copy `myparams.ini` to the PARAMS device using the following DOS command:

```
COPY MYPARAMS.INI PARAMS
```

- ❑ **Run-Time Library API** The API in the Falcon run-time library contains two function calls that allow you to send configuration strings to the PARAMS driver from within a C application. See `CFG_Write()` and `CFG_Read()` in the *Falcon DOS Portable Terminals Programmer's Reference*.

Attempts to create, delete, or modify a file called `params` will conflict with the PARAMS device. You cannot eliminate the loading of `cfgdev.sys` except through the safe-boot process.

>> DECODE.SYS

This driver controls bar code scanning devices and decodes bar code labels into text. If special programming bar codes are read, the decoded information will be sent to the PARAMS device (see [CFGDEV.SYS](#), on page 84). Otherwise, the resulting text is placed in the keyboard buffer for reading by the application. `decode.sys` is located on drive **B** of the Falcon unit and is loaded by the `config.sys` file located on that drive. You cannot eliminate the loading of `decode.sys` except through the safe-boot process.

>> FLASHDSK.SYS

This driver configures all of the unit's available flash memory as a read-write disk, called a flash disk. Reading from a flash disk is fast, but writing can be somewhat slower. This driver is located on the Falcon's **B** drive and is loaded by the `config.sys` file located on that drive. You cannot eliminate the loading of `flashdisk.sys`.

>> FORMAT.COM

Formats a read-write disk drive. Only the flash and RAM drives may be formatted using this command. FORMAT uses the existing drive parameters to recreate the drive, but without any files stored on it. This provides a quick method for cleaning out an existing drive.

Syntax `FORMAT [drive:] [/y]`

Options `drive`

Specifies the letter of the drive to be formatted. If you do not specify a drive letter, the current drive is used.

`/y`

Suppresses output from being sent to the display, and bypasses the usual prompt to proceed with formatting.

>> LOCK.COM

Use this utility to enable or disable write protection on flash or RAM drives. (This is the software equivalent of the write-protect tab on a floppy disk.) You can use LOCK to protect against accidental erasure of all files stored on the flash disk.

Whenever the Falcon is rebooted, write protection on a drive will be disabled, which is the default state. To lock the drive upon bootup, include a LOCK command in the `autoexec.bat` file on drive C.

Syntax `LOCK [drive:] [/l] [/u] [/y]`

Options `drive`

Specifies the drive letter of the drive which is to be locked or unlocked. If you do not specify a drive letter, the current drive is used.

`/l` (*the letter L*)

Lock the flash drive to prevent further disk writes.

`/u`

Unlocks the flash drive, allowing further disk writes.

`/y`

Suppresses output from being sent to the display.

NOTE: *If both the /l and /u options are given, the last option specified will take effect.*

>> ORGANIZE.COM

This utility reorganizes sectors on a flash drive to improve flash disk access. Data is relocated so that all free sectors are grouped together, reducing the overhead of freeing space for new data. `organize.com` has a greater effect on a flash disk as the disk gets closer to being full.

NOTE: *The only Falcon drive you can organize with this utility is the flash disk created with the `flashdisk.sys` driver. ATA flash drives will not respond properly to this utility.*

Syntax `ORGANIZE [drive:] [/y]`

Options `drive`

Specifies the drive letter of the flash drive which is to be reorganized. If a drive letter is not given, the current drive will be used.

`/y`

Suppresses output from being sent to the display.

>> PM.COM

Use this utility to control various power-management features of the Falcon. You can use it at the command line or in a batch file (including `autoexec.bat`) to set or read power-management settings.

Syntax `PM [/b#] [/f#] [/h] [/i] [/r] [/t#]`

Options `/b#`

Sets the automatic timeout for the backlight. Replace the `#` with the number of seconds to wait after a keypress before automatically turning off the backlight. Each keypress restarts the timeout countdown. The range of acceptable values for `#` is 0 (off) to 255; the default is 15 seconds.

`/f#`

Sets the time interval for audio indication when the batteries are low. Replace the `#` with the number of minutes between sounding of the tone. The tone will not sound until the battery has been in the low state for the same number of minutes. The range of acceptable values for `#` is 0 (off) to 255; this option is off by default.

NOTE: *The audio warning uses extra current, causing the batteries to drain faster. Therefore, you might want to set the interval for as long as possible or not use it at all.*

`/h`

Displays a help file for the utility.

`/i`

Displays timer settings for auto-off, backlight timeout, and low-battery audio indication.

`/r`

Resets power management to the cold-boot defaults.

`/t#`

Sets the auto-off timeout. Replace the `#` with the number of seconds of nonuse before the unit should transition to its lowest power state. The range of acceptable values for `#` is 16 to 1032, or 0 for off; the default is 300 seconds (5 minutes).

>> *VDISK.SYS*

This driver configures all of the unit's available extended memory as a read/write disk, called a RAM disk. This driver is located on drive `B` of the Falcon unit and is loaded by the `config.sys` file located on that drive. You cannot configure the parameters for this driver.

>> *XFER.EXE*

The XFER utility gives you the ability to transfer files to and from a PC through the Falcon's serial port. For basic information about XFER, see pages [8-9](#). For complete information, see chapter [3](#).

>> Chapter Nine

ROM-DOS Commands

This chapter lists the Datalight ROM-DOS commands that are available in the Falcon. ROM-DOS commands that differ from their MS-DOS equivalents are identified and described.

The information in this chapter is taken from Datalight's *ROM-DOS 6.22 User's Guide* and is used by permission.

CHAPTER CONTENTS

<i>Command Overview</i>	<90>
<i>ROM-DOS vs. MS-DOS</i>	<92>
<i>Command Descriptions</i>	<93>

>> Command Overview

Table 9-1 lists the Datalight ROM-DOS commands that are provided with the Falcon and gives a brief description of each command.

Table 9-1: DOS Commands Used with the Falcon

Command	Description
?	A <code>config.sys</code> command. Directs ROM-DOS to pause for confirmation before processing a command.
@	Suppresses the display of a single batch file command line.
;	Same as <code>REM</code> . Identifies nonexecuting lines.
<code>ATTRIB</code>	Displays or modifies the attributes associated with a file.
<code>BREAK</code>	A <code>config.sys</code> command. Enables or disables the ability to stop program execution at a non-I/O point.
<code>BUFFERS</code>	A <code>config.sys</code> command. Sets the number of internal data buffers.
<code>CALL</code>	Batch subcommand. Invokes execution of a secondary batch file.
<code>CHDIR</code>	<code>CH</code> ange <code>DIR</code> ectory (also <code>CD</code>). Changes the default directory.
<code>CHKDSK</code>	<code>CH</code> eck <code>DISK</code> . Checks integrity of data on a disk. Displays information.
<code>CLS</code>	<code>CL</code> ear <code>S</code> creen. Clears all information from the viewport.
<code>COMMAND</code>	Spawns a second DOS command processor.
<code>COPY</code>	Copies files from one storage location to another.
<code>CTTY</code>	<code>CH</code> ange <code>Te</code> le <code>T</code> ype. Changes the default terminal interacting with ROM-DOS.
<code>DATE</code>	Displays the date from the system's internal calendar and allows revision.
<code>DEL</code>	<code>DE</code> lete. Deletes specified files.
<code>DEVICE</code>	A <code>config.sys</code> command. Installs a device driver into ROM-DOS.
<code>DOS</code>	A <code>config.sys</code> command. Installs ROM-DOS into high-memory area (HMA).
<code>ECHO</code>	Batch subcommand. Turns on the display of batch execution on the viewport.
<code>ERASE</code>	Same as <code>DEL</code> . Erases specified files.
<code>EXIT</code>	Exits "nested" running of ROM-DOS within another program.
<code>FCBS</code>	A <code>config.sys</code> command. Specifies the number of file control blocks (FCBS) open at one time.
<code>FILES</code>	A <code>config.sys</code> command. Sets the maximum number of files that can be open at one time on the system.
<code>FIND</code>	Works as a filter to display only lines that contain a specified string.
<code>FOR</code>	Batch subcommand. Performs one DOS command on a set of files.
<code>GOTO</code>	Batch subcommand. Moves control to a specified line in the batch file.
<code>HELP</code>	Lists all available ROM-DOS commands along with brief descriptions.
<code>IF</code>	Batch subcommand. Performs a command based on a specified condition.

Command	Description
INCLUDE	A <code>config.sys</code> command. Allows instructions in one configuration block to be included with instructions in another configuration block.
INSTALL	Loads terminate-and-stay resident (TSR) programs during <code>config.sys</code> processing.
LASTDRIVE	Sets the maximum number of drives.
MENUDEFAULT	Sets the default menu item choice and time-out value for making a selection.
MENUITEM	Specifies an item to be placed in the startup menu displayed during system boot.
MKDIR	Ma ke DI rectory (also MD). Creates a new subdirectory.
NEWFILE	Allows continuation of <code>config.sys</code> processing from a new file.
PATH	Displays current command search path(s). A new path line can be specified.
PAUSE	Batch subcommand. Causes execution to halt until a key is pressed.
PRINT	Prints a list of files, up to ten files.
PROMPT	Resets the appearance of the system prompt line.
REM	RE mark. A batch subcommand for identifying nonexecuting lines.
REN	RE name. Renames files.
RMDIR	RE move DI rectory (also RM). Deletes a specified subdirectory.
SET	Sets environment variables and command processor strings.
SHARE	Installs the capabilities for file sharing and file locking on your hard disk.
SHELL	Allows selections of an alternate boot program other than the default <code>command.com</code> command processor.
SHIFT	Batch subcommand. Shifts replaceable parameters one position "to the left."
SORT	Sorts a text file and displays the output to the standard device.
STACKS	Allows for the use of dynamic data stacks to handle interrupts. Note: ROM-DOS does not utilize this command.
SUBMENU	Defines a menu item that represents a secondary menu.
SWITCHES	Allows special <code>config.sys</code> file options.
TIME	Displays the current time from the system's internal clock. Allows revision.
TREE	Displays the path of each directory on a specified drive.
TYPE	Displays the contents of a text file on the viewport.
VER	Displays current version of ROM-DOS on the viewport.
VERIFY	Displays the current VERIFY state or sets the VERIFY state to on or off.
VOL	Displays the volume label on a disk.
XCOPY	Copies multiple files and, optionally, subdirectories.
XDEL	Deletes files and subdirectories including empty subdirectories.

>> ROM-DOS vs. MS-DOS

Table 9-2 identifies differences between ROM-DOS commands and their MS-DOS equivalents. For more information, see the individual command descriptions in the next section.

Table 9-2: Difference Between ROM-DOS and MS-DOS Commands

Command	Features Exclusive to ROM-DOS	Features Exclusive to MS-DOS
ATTRIB	<code>-C</code> : Clears all file attributes	<code>/S</code> : Processes files in all subdirectories
BUFFERS		Allows designation of secondary buffer cache <code>BUFFERS n,m</code> where <code>m</code> is secondary buffer cache
CHKDSK	<code>/C</code> : Corrects errors without user confirmation	
COMMAND	Internal command help only	<code>/Y</code> : Steps through batch file specified by <code>/C</code> or <code>/K</code> switch <code>help.hlp</code> in MS-DOS includes all commands (external and internal), installable drivers, etc.
DIR	<code>/A X</code> : Shows attributes	<code>/O:C</code> and <code>/O:-C</code> : Sorts by compression ratio <code>/C[H]</code> : Displays compression ratio of files compressed using Drivespace or Doublespace
FCBS	<code>, [minimum number]</code>	
FIND		<code>/I</code> : Not case-sensitive
HELP	Requires <code>command.hlp</code>	MS-DOS requires external file, <code>help.com</code> , but provides more help text
NEWFILE	ALL	Not available with MS-DOS
PRINT	<code>/F</code> : Sets maximum number of files	<code>/D:LPTx</code> (Falcon has no parallel port.) <code>/U</code> <code>/M</code> <code>/S</code> <code>/Q</code>
SHARE	<code>/U</code> : Unloads <code>share.exe</code>	<code>/F:space</code> : Allocates space to record
SWITCHES		<code>/W</code>
TREE		<code>/A</code> : Indicates no graphics characters for tree symbols
VER	<code>/R</code> : Shows full version and release number Allows revision of the version number	
XCOPY		<code>/Y</code> <code>/-Y</code> : Turns confirmation prompts on or off
XDEL	All	Not available with MS-DOS Similar to <code>DELTREE</code>

>> Command Descriptions

This section provides descriptions of the ROM-DOS commands listed in table 9-2. Each entry includes an explanation of the command's purpose, the command entry syntax, remarks, and examples.

Each command also has a label to designate whether it is an internal or external command. Internal commands are part of the command processor program, `command.com`. These functions are available only while `command.com` is running. External commands are actually stand-alone utility programs. They are independent from `command.com`. Internal commands that are unique to `config.sys` processing are also identified. These commands can be used only in a `config.sys` file.

ATTRIB

Type External

Purpose The `ATTRIB` command either displays or modifies the attribute of a file.

Syntax `ATTRIB [+ | -][option][filespec]`

Remarks The file attributes define the characteristics of a file. They determine if a file may be deleted or modified, or if it is archived. The `ATTRIB` command is used to manage these file attributes.

Wildcard characters may be used in the `ATTRIB filespec`.

The `ATTRIB` command will modify file attributes if modify commands are given to `ATTRIB`. The modify commands are:

Command	Description
+/-	Add(+) or remove(-) attribute
A	Archive attribute
C	Clear all attributes
H	Hidden file attribute
R	Read only attribute
S	System file attribute

If no modify commands are found by `ATTRIB`, then the files are displayed along with the file names and their current attributes.

Examples `ATTRIB` will add the read-only attribute to the file `myfile.dat`.

```
ATTRIB +r myfile.dat
```

`ATTRIB` will remove the read-only attribute and the archive attribute of all files with the DAT extension.

```
ATTRIB -a -r *.dat
```

`ATTRIB` will display the attributes of all files with the DAT extension.

```
ATTRIB *.dat
```

BUFFERS

Type `config.sys`

Purpose ROM-DOS has internal buffers to temporarily hold data read from the disk. Increasing the number of internal buffers will speed system performance.

Syntax `BUFFERS = number`

Remarks Each buffer used by ROM-DOS requires 512 bytes of RAM. The `BUFFERS` command will increase or decrease the amount of RAM used by the operating system.

The minimum number of buffers is 2 and the maximum number is 40. If a number less than 2 is given then the number of `BUFFERS` is set to 2. If a number larger than 40 is given then `BUFFERS` is set to 40.

Example The following example causes ROM-DOS to have 10 buffers. These 10 buffers will use 5120 bytes of RAM.

```
BUFFERS = 10
```

CHKDSK

Type External

Purpose The CHKDSK command checks the disk directories and File Allocation Table (FAT) and displays a disk and memory report.

Syntax `CHKDSK [d:][path][filespec][/C] [/F] [/V]`

Remarks `CHKDSK` examines a disk and determines if the disk has any errors in the File Allocation Table (FAT) and will optionally fix errors.

Options The `/F` option causes `CHKDSK` to fix errors on the disk if any were found. The errors that can be found are directory or FAT errors. If the `/F` is not specified then `CHKDSK` acts as if it will fix the disk, but the corrections will not be written out to the disk.

If errors are detected, you will be prompted with a message similar to the following:

```
15 lost allocation units found in 5 chains.
Convert lost chains to files?
```

If you answer `Y` for Yes, each lost chain will be written to a file in the root directory of the current default drive. Each file will have the name `filennnn.chk`. `nnnn` will be a sequential number. The first chain will be in `file000.chk`. These files can be verified to see if they contain valuable information, and then deleted if desired. Answering `N` for No to the above prompt, `CHKDSK` will still make the corrections however the lost chains will not be saved to the disk.

The `/C` option allows `CHKDSK` to correct errors without user confirmation. This option must be used along with the `/F` option for corrections to be made.

The `/V` option causes `CHKDSK` to display each path and file as it is processed.

If a file specification is specified, then `CHKDSK` displays all files matching the specification that have noncontiguous data areas on the disk. Files that are stored in noncontiguous areas, especially `.exe` files, have slower disk access times. If `CHKDSK` reports a large number of files with this problem, a utility program that optimizes the files and free space on your disk should be used.

After checking the disk, `CHKDSK` displays any error messages followed by a report on the state of the disk that was checked. An example of the report is shown below.

```
Volume ROM-DOS created June 1,1990 1:00a
Volume Serial Number is 190E-4AA2

362496 bytes total disk space
    0 bytes in 1 hidden files
    6144 bytes in 2 user files
356352 bytes available on disk
655360 bytes total memory
595360 bytes free
```

CHKDSK does not wait for a disk to be inserted before the checking is initiated nor does it repair any errors.

Examples

CHKDSK will check the integrity of drive **A**. The report will be printed to the console.

```
CHKDSK a:
```

CHKDSK will check the integrity of RAM disk **D**. The report will be saved in a file called `drive_d.rpt`.

```
CHKDSK d: >drive_d.rpt
```

COMMAND

Type External

Purpose Start a new command processor.

Syntax `COMMAND[device][/E:number][/K:filename][/P][/C string][/MSG]`

Remarks This command starts a new copy of the ROM-DOS command processor. The command processor is the program that has all the internal DOS commands in it.

Starting a new command processor will also produce a new environment. The size of the environment is 128 bytes by default, but it can be changed using the **/E** switch.

Command and its arguments can also be used in a **SHELL=** statement in your `config.sys` file. See the full description of **SHELL** for more details.

Options The *device* option specifies that `command.com` should use a different device, such as AUX, for input and output.

The `/E:number` switch sets the environment size. `Number` represents the size of the environment in bytes. `Number` must be in the range from 160 to 32768. All other values will be ignored and the default value of 256 will be used. ROM-DOS will round the value entered up to the nearest multiple of 16.

The `/K:filename` option tells the command processor to run the specified filename and then display the ROM-DOS command prompt. It is not recommended that this option be used in a `CONFIG.SYS SHELL=` statement.

The `/P` switch causes `COMMAND` not to exit, or in other words, to remain permanent. The `/P` switch should be used only when `command` is used in a `CONFIG.SYS SHELL` statement.

The `/c string` switch causes `COMMAND` to execute the command in `string` and then terminate. The string command can be any internal or external command.

The `/MSG` option indicates that all error messages should be stored in memory. This option is recommended only for diskette based systems. ROM-DOS keeps many of its error messages in the resident part of `COMMAND.COM` rather than using valuable memory to store them. If an error message is needed and you have loaded ROM-DOS from a diskette, the message will only be available if the boot disk is still in the drive. By using the `/MSG` option, the messages will be available in memory at all times. The `/P` option must be used along with the `/MSG` option.

Examples The following command will cause a new copy of `COMMAND` to be executed. It will perform a `DIR` command on the C drive and then exit back to the previous Command Processor.

```
COMMAND /C DIR C:
```

The following example shows loading of a permanent copy of `command` with an environment size of 256 bytes.

```
SHELL=C:\COMMAND.COM /P /E:256
```

DIR

Type Internal

Purpose Displays a list of the files that are in a directory.

Syntax `DIR [d:][path][filename][/option]`

Remarks The `DIR` command can be used to list all the files in a directory, or to show the directory entries of specific files. The standard directory display format includes columns for filenames, filename extensions, file sizes, and the dates and times the files were created.

Options `/A attributes`

The `/A` option causes the `DIR` command to display only the files that match the specified filespec and have the given attribute. The table below shows the legal attribute descriptions.

Letter	Description
<code>A</code>	Archive
<code>D</code>	Directories
<code>H</code>	Hidden files
<code>R</code>	Read-only files
<code>S</code>	System files
<code>X</code>	Show attributes

`-`

The dash (-) symbol can be used to negate listed attributes. For example, to select all files that do not have the archive bit set, use the `/A -A` option.

`/B`

The `/B`, or bare, option causes the display to be displayed without volume label, date, time, or size information.

`/L`

The `/L` option causes the filenames to be displayed in lowercase.

`/P`

The `/P` option selects page mode, which makes ROM-DOS pause the display each time the screen is full. Press any key to go on to the next page of entries.

`/O`

The `/O` option causes the filenames to be displayed in sorted order. The sort order can contain one or more of the following letters:

Letter	Description
<code>D</code>	By date and time newest first
<code>E</code>	Alphabetic order by extension

Letter	Description
G	Directories grouped before files
N	Alphabetic order by name
S	Size, smallest first

-

The dash (-) symbol can precede the sort option to reverse the sort order. For example, to sort all files in the directory in reverse alphabetic order, use the **/O-N** option.

/S

The **/S** option causes the display to include files in subdirectories also.

/W

Display list in a wide format without date, time, or size.

The **DIRCMD** environment variable can be used to set the default preferences for the **DIR** command. The **SET** command will assign the values to an environment variable. Refer to the **SET** command section for proper usage. For example, if you wanted to always have the **/P** option set for **DIR**, the statement **SET DIRCMD=/P** could be used. The default settings in **DIRCMD** can be overridden by using the minus sign (-) preceding the option. If you wanted to cancel the paging for a single use of the **DIR** command, you would enter **DIR /-P**.

Examples

To see the directory entries of all files in the current drive and directory, type:

```
DIR
```

To see all files in the subdirectory MEMOS on drive B, type:

```
DIR B:\MEMOS
```

Display all files sorted by file name order.

```
DIR /ON
```

Display all hidden files.

```
DIR /AH
```

Display all files with a .DOC extension without file sizes, or volume labels.

```
DIR *.DOC /B
```

FCBS

Type	CONFIG.SYS
Purpose	The FCBS command allows you to specify the number of file control blocks (FCBs) open at one time.
Syntax	<code>FCBS = number [,minimum number]</code>
Remarks	<i>Number</i> specifies the maximum number of FCBs open at any given time. The default for this value is 4. The value for <i>number</i> must be in the range from 1 to 255. The minimum number specifies the minimum number of FCBs to be open at all times. The <i>minimum number</i> argument has the same default and range value as the <i>number</i> argument.
Example	Set the maximum number of FCBs to 8 and leave at least 4 open at all times. <pre>FCBS = 8,4</pre>

FIND

Type	External
Purpose	FIND is a filter to display only lines that contain a specified string. The input to FIND may come from a file, or it may be piped in from another filter or a DOS command.
Syntax	<code>FIND [/option] match-string [filename]</code>
Options	The /C option causes FIND to only display the count of lines found with the specified string. The /N option causes FIND to display the line number of the line found containing the string. The /V option causes FIND to display the lines that do not contain the string. The <i>match-string</i> argument specifies the word or group of characters to search for.

The `filename` argument specifies the file or group of files to search in. The complete drive and path can be specified. Wildcard characters can be used in the filename.

Examples

The following example shows each line in the file JUNK.C that contains the string "printf".

```
    FIND printf junk.c
```

The following example shows each line in a directory listing that contains a `DIR`. The command first executes a `DOS DIR` command with the output piped into the `FIND` command. The `FIND` command then displays each line that contains the string "DIR".

```
    dir | FIND DIR
```

The following example give a count of the lines in the file MANUAL.TXT that contain the string "ROM-DOS".

```
    FIND /C ROM-DOS MANUAL.TXT
```

```
    .....MANUAL.TXT: 105
```

HELP

Type Internal

Purpose Provides on-line help of each ROM-DOS command.

Syntax `HELP <command>`

Remarks `HELP` serves as a memory aid. For complete information about ROM-DOS commands, always consult this manual.

`HELP` for each command can also be displayed by entering `/?` following the command name.

The file `COMMAND.HLP` must be available to use this command.

Examples To list the help of the DIR command you can type:

```
HELP DIR
```

or

```
DIR /?
```

All available batch file commands are also listed by **HELP**.

NEWFILE

Type CONFIG.SYS

Purpose The **NEWFILE** command allows you to continue CONFIG.SYS file processing from a new file. The file can be located on in another directory or even on a different drive.

Syntax **NEWFILE=filename**

Remarks The **NEWFILE** command is especially useful when the CONFIG.SYS file is located on an inaccessible drive or in ROM. Additional device drivers or instructions can be added easily to the new file and will be processed along with the main CONFIG.SYS file upon starting the system.

When the **NEWFILE=** instruction is processed, control will be passed from the present file (the one containing the **NEWFILE** instruction) to the file specified in the command. Any commands placed after the **NEWFILE** instruction in the original file will not be processed. If for some reason the specified file name can not be located, CONFIG.SYS processing will be terminated (even if instructions were to be placed after the **NEWFILE** command) and the remainder of the startup process will be completed.

NEWFILE commands can be nested. That is, your original CONFIG.SYS can call a second set of instructions via the **NEWFILE** command. The second file can in turn call a third file by using the **NEWFILE** command, and so on. Be sure that each filename in the successive steps has a unique name, otherwise, an infinite loop will be created as control is passed back to the same file repeatedly.

Each filename given in a **NEWFILE** command line will have an environment variable of the same name.

Example The following example will cause instructions in the file NEWCFG.SYS, located in the C:\BIN directory, to be executed as part of the CONFIG.SYS file. The contents of NEWCFG.SYS may include any of the commands listed in this section.

```
NEWFILE=C:\BIN\NEWCFG.SYS
```

PRINT

Type External

Purpose The PRINT utility prints a single file or a list of files.

Syntax `PRINT [/d:] [filename] [/options]`

Remarks `PRINT` allows you to enter between 1 and 32 files for spooling to the printer. The files are output to the device in a spooled manner (while the user performs other operations).

If `PRINT` is entered without any parameters, then it displays all the files that are in the queue.

The first time `PRINT` is used the operator is prompted for the device to perform the operation. The following message is used to prompt the operator for the device.

```
Name of list device [PRN]:
```

The legal devices for printing are LPT1, LPT2, LPT3, LPT4, COM1, COM2, COM3, COM4, AUX, or PRN.

Options The `/B` option allows the user to set the buffer size. The default buffer size is 512 bytes. A larger buffer size causes print to operate faster. The maximum buffer size is 32k bytes and the minimum size 256 bytes. This option is only allowed the first time `PRINT` is run.

The `/C` option cancels only the file names listed after the `/C` command.

The `/F` option allows the user to set the maximum number of files to be queued up at one time. The default number of files is 10. The minimum is 2 and the maximum is 32. Support for more files is often useful when using wild cards in file names. This option is only allowed the first time `PRINT` is run (or until the next system reboot).

The **/P** option causes all files listed after this option to be submitted for printing. This is the default for filenames encountered on the **PRINT** command line.

The **/T** option cancels all the files from the print queue (list). Think of this as a terminator.

The **/H** option will display the help screen.

Examples

PRINT puts three files into the print queue. The first file will start being printed after the command ends.

```
PRINT FILE1.TXT FILE2.TXT FILE3.TXT
```

The file FILE2.TXT will be removed from the print queue. All other files in the queue will print normally.

```
PRINT /C FILE2.TXT
```

All files in the print queue are canceled. Printing may continue for a short time because of the buffer in your printer.

```
PRINT /T
```

SHARE

Type External

Purpose **SHARE** installs the capabilities for file-sharing and file-locking on your hard disk.

Syntax **SHARE** [/options]
Or from CONFIG.SYS:
INSTALL=[d:][path]SHARE.EXE [/options]

Remarks The **SHARE** utility is most commonly used in a network or multitasking environment where file sharing is necessary. When **SHARE** is loaded, DOS will utilize the **SHARE** utility to validate read and write requests from application programs and users.

The **/L:#** option specifies the maximum number of files that can be locked at one time. The default number of files is 20.

The `/U` option unloads the share utility and frees the memory. `SHARE` will not unload if other TSRs have been loaded on top of it. The other TSRs must be unloaded first before trying to unload `SHARE`.

Examples

The following example loads the `SHARE` program from the command line:

```
SHARE
```

The next example installs `SHARE` from the CONFIG.SYS file and changes the maximum number of locked files to 30:

```
INSTALL=C:\UTILS\SHARE.EXE /1:30
```

The final example unloads `SHARE` and frees the used memory.

```
SHARE /U
```

SWITCHES

Type

CONFIG.SYS

Purpose

The `SWITCHES` command allows special CONFIG.SYS file options.

Syntax

```
SWITCHES=[/k][/n][/f]
```

Remarks

The `/k` argument makes an enhanced keyboard behave like a conventional-style keyboard.

The `/n` argument prevents the use of the `F5` and `F8` function keys to bypass the startup commands.

The `/f` argument instructs ROM-DOS to skip the delay after displaying the `Starting ROM-DOS...` message at boot time. The delay allows the user time to use the `F5` and `F8` options to alter the processing of the startup files.

Examples

The following example prevents the user from using the `F5` and `F8` keys at boot time.

```
switches = /n
```

TREE

Type	External
Purpose	The TREE command displays each subdirectory and optionally the files within them for a specified drive.
Syntax	TREE [<i>d:</i>] [/F]
Remarks	<p>The TREE command displays the full path of each subdirectory on a specified disk.</p> <p>The <i>d:</i> specifies the drive that TREE will display the subdirectories from. This argument must be specified.</p>
Options	The /F switch causes TREE to display the files in each subdirectory.
Examples	<p>This command will display all subdirectories on drive C.</p> <pre>C:\DATA> TREE C:</pre> <p>This command will display all subdirectories on drive A along with the files within each sub directory.</p> <pre>C:\DATA> TREE A: /F</pre>

VER

Type	Internal
Purpose	Displays the version number of ROM-DOS in use. Allows revision of this version number.
Syntax	VER [<i>n.nn</i>] [/R]
Remarks	<p>If a new version number is specified, two digits after the decimal are required. Note that this command revises only the record of the DOS version number; it does not change the actual operating system loaded in the computer.</p> <p>The version command shows both the version of the VER command itself and the version of DOS in operation.</p>

Options The `/R` option shows the full version and release number of ROM-DOS.

Example The following example changes the record of current DOS version in use to DOS 5.0. Any programs that are executed, following this command, will recognize that DOS 5.0 is running.

```
VER 5.0
```

XCOPY

Type External

Purpose The `XCOPY` command copies multiple files and optionally subdirectories from one disk to another.

Syntax `XCOPY [source] [target] [/options]`

Remarks The `XCOPY` command is used for copying multiple files and subdirectories, if they exist.

The `source` and the `target` parameter are complete drive path and file specification descriptions. If you do not specify a path, `XCOPY` assumes the default path. If a file name is not specified then `*.*` is assumed.

The `ATTRIB` command may be used to modify the archive bit for the various `XCOPY` options that check the archive status of files. Refer to the `ATTRIB` command description for instructions.

Options The `/A` copies only source files that have the archive bit set in them. The archive is not reset.

The `/D<mm-dd-yy>` option causes `XCOPY` to copy only those files with a date later than the date specified in the `/D` option.

The `/E` option causes `XCOPY` to create subdirectories on the target even if they are empty.

The `/M` option causes `XCOPY` to copy only those source files that have the archive bit set. Once the source file is copied the archive bit is reset.

The **/P** option causes **XCOPY** to prompt before each file is copied. The prompt appears as follows:

```
C:\COMMAND.COM (Y/N)?
```

If a Y is entered then the file is copied, otherwise, the file is not copied.

The **/S** option causes **XCOPY** to copy files in subdirectories of the source directory.

The **/V** option causes **XCOPY** to verify each write to the disk.

The **/W** option causes **XCOPY** to wait before starting to copy files. The following prompt is displayed.

```
Press any key to begin copying file(s)
```

Example **XCOPY** to the **A** drive all files in the **BIN** subdirectory that have an **EXE** extension and that have the archive bit set.

```
XCOPY \bin\*.exe a: /a
```

XDEL

Type External

Purpose The **XDEL** command deletes files and subdirectories including empty subdirectories.

Syntax **XDEL** *filespec* [/options]

Remarks The **XDEL** command allows the deletion of files and subdirectories in the same step.

The *filespec* argument is the starting point for the deletion. The *filespec* argument can contain the drive and path for reaching the starting point and can also contain wild card characters to designate a group of file or directory names.

Options The **/D** option deletes empty subdirectories. Deletion will not occur if there are any files in the subdirectory, unless the **/S** option is specified along with the **/D** option.

The **/P** option gives you a confirmation prompt before deleting each file.

The `/R` option allows deletion of read-only files without having to change the file attributes prior to the delete.

The `/S` option deletes files in subdirectories located below the specified starting directory. When used along with the `/D` option, it will delete the files within a subdirectory along with the subdirectory entry itself.

Examples

The following examples will all use the diagram below as a reference:

The following `XDEL` command would delete all of the files in the directories DIR1, SUB1, and SUB2 but not the directory headings themselves.

```
XDEL DIR1 /S
```

To delete the empty subdirectory heading for SUB3, use `XDEL` as follows:

```
XDEL DIR1 /D
```

To delete all of the files in the three directories and the directory headings at the same time, use the following command:

```
XDEL DIR1 /S /D
```


>> Appendix A

Programming Parameters

This chapter contains information about programmable settings for the Falcon. You can use the Falcon Configuration Utility (see chapter [2](#)) or the bar codes provided in appendix [C](#) to program your Falcon.

CHAPTER CONTENTS

<i>Code 39</i>	<113>
<i>Interleaved 2 of 5</i>	<113>
<i>Matrix 2 of 5</i>	<113>
<i>Standard 2 of 5</i>	<113>
<i>Code 11</i>	<113>
<i>Codabar/Ames</i>	<114>
<i>MSI</i>	<114>
<i>Code 93</i>	<114>
<i>Universal Product Code-A (UPC-A)</i>	<114>
<i>Universal Product Code-E (UPC-E)</i>	<114>
<i>European Article Numbering (EAN)</i>	
<i>Japan Article Numbering (JAN)</i>	<114>
<i>UPC, EAN, JAN Extensions</i>	<115>
<i>Code 128</i>	<115>
<i>Labelcode 4/5</i>	<115>
<i>Other Controls</i>	<115>

Table [A-1](#) provides the following information:

- ❑ **Code Parameter** is the “human” name for the programming option.
- ❑ **I.D. #** is the “decoder” name for the programming option. For example, if you wanted to set a Code 39 minimum label length, you would use I.D. #01. Programming I.D. numbers given in this chapter can be used with all programming methods.
- ❑ **Type** tells what kind of setting to use for each code parameter.
 - **On/Off** is a toggle. 1 turns the parameter on, and 0 turns it off.
 - **Value** requires a two-character entry (e.g., 02 for two beeps after each good read).
- ❑ **Acceptable Input** gives the settings or range of settings that you can use for each code parameter.
- ❑ **Defaults** tells how the parameter is set when you select pre-defined default **D0**, **D1**, or **D2**.
 - **Predefined Default 0 (D0)** turns every on/off parameter off and sets all minimum and maximum lengths to the lowest values.
 - **Predefined Default 1 (D1)** turns every on/off parameter on, sets all minimum lengths to the lowest values, and sets all maximum lengths to the highest values. This default set is normally used only for troubleshooting. It gives you the best chance of reading an unknown bar code symbology and also identifies the symbology of each bar code you scan.
 - **Predefined Default 2 (D2)** is the default set that was installed in the Falcon at the factory. This default set will work for most applications. If you have changed any settings and want to reset the Falcon to the original defaults, scan the **D2** bar code on page [124](#).

Table A-1: Parameters, Settings, and Defaults

Code Parameter	I.D. #	Type	Acceptable Input	Defaults		
				D0	D1	D2
Code 39				<i>Enter 1 for On and 0 for Off.</i>		
Enable	00	On/Off	On or Off	Off	On	On
Minimum length	01	Value	00–50	00	00	00
Maximum length	02	Value	01–50	01	50	20
Enable checksum	03	On/Off	On or Off	Off	Off	Off
Send checksum	04	On/Off	On or Off	Off	Off	Off
Full ASCII mode	05	On/Off	On or Off	Off	On	On
Interleaved 2 of 5				<i>Enter 1 for On and 0 for Off.</i>		
Enable	08	On/Off	On or Off	Off	On	On
Minimum length	09	Value	02–50	02	02	06
Maximum length	0A	Value	02–50	02	50	10
Enable checksum	0B	On/Off	On or Off	Off	Off	Off
Send checksum	0C	On/Off	On or Off	Off	Off	Off
Use lengths 6 and 14 only (case code)	0D	On/Off	On or Off	Off	Off	Off
Matrix 2 of 5				<i>Enter 1 for On and 0 for Off.</i>		
Enable	10	On/Off	On or Off	Off	On	Off
Minimum length	11	Value	01–50	01	01	06
Maximum length	12	Value	01–50	1	50	10
Enable checksum	13	On/Off	On or Off	Off	Off	Off
Send checksum	14	On/Off	On or Off	Off	Off	Off
Standard 2 of 5				<i>Enter 1 for On and 0 for Off.</i>		
Enable	15	On/Off	On or Off	Off	On	Off
Minimum length	16	Value	01–50	01	01	06
Maximum length	17	Value	01–50	01	50	10
Enable checksum	18	On/Off	On or Off	Off	Off	Off
Send checksum	19	On/Off	On or Off	Off	Off	Off
Use 2-bar start/stop	1A	On/Off	On or Off	Off	Off	Off
Code 11				<i>Enter 1 for On and 0 for Off.</i>		
Enable	1B	On/Off	On or Off	Off	On	Off
Minimum length	1C	Value	01–50	01	01	04
Maximum length	1D	Value	01–50	01	50	10
Require 2 check digits	1E	On/Off	On or Off	Off	Off	Off
Send check digit(s)	1F	On/Off	On or Off	Off	On	Off

table continues

Programming Parameters

Code Parameter	I.D. #	Type	Acceptable Input	Defaults		
				D0	D1	D2
Codabar/Ames				<i>Enter 1 for On and 0 for Off.</i>		
Enable	20	On/Off	On or Off	Off	On	On
Minimum length	21	Value	01–50	01	01	04
Maximum length	22	Value	01–50	01	50	20
Send start/stop	23	On/Off	On or Off	Off	On	Off
Codabar-to-CLSI conversion	24	On/Off	On or Off	Off	Off	Off
Wide intercharacter gaps allowed	25	On/Off	On or Off	Off	Off	On
MSI				<i>Enter 1 for On and 0 for Off.</i>		
Enable	26	On/Off	On or Off	Off	On	Off
Minimum length	27	Value	01–14	01	01	04
Maximum length	28	Value	01–14	01	14	10
Require 2 check digits	29	On/Off	On or Off	Off	Off	Off
2nd check digit Mod 11	2A	On/Off	On or Off	Off	Off	Off
Send check digit(s)	2B	On/Off	On or Off	Off	On	Off
Code 93				<i>Enter 1 for On and 0 for Off.</i>		
Enable	2C	On/Off	On or Off	Off	On	Off
Minimum length	2D	Value	00–50	00	01	02
Maximum length	2E	Value	01–50	00	50	20
Universal Product Code-A (UPC-A)				<i>Enter 1 for On and 0 for Off.</i>		
Enable	30	On/Off	On or Off	Off	On	On
Send system digit	31	On/Off	On or Off	Off	On	On
Send check digit	32	On/Off	On or Off	Off	On	Off
Convert UPC-A to EAN-13	33	On/Off	On or Off	Off	On	Off
Universal Product Code-E (UPC-E)				<i>Enter 1 for On and 0 for Off.</i>		
Use system digit 0	34	On/Off	On or Off	Off	On	On
Use system digit 1	35	On/Off	On or Off	Off	On	On
Convert UPC-E to UPC-A	36	On/Off	On or Off	Off	On	Off
Send system digit	37	On/Off	On or Off	Off	On	Off
Send check digit	38	On/Off	On or Off	Off	On	Off
European Article Numbering (EAN) Japan Article Numbering (JAN)				<i>Enter 1 for On and 0 for Off.</i>		
Enable EAN-8/JAN-8	39	On/Off	On or Off	Off	On	On
Enable EAN-13/JAN-13	3A	On/Off	On or Off	Off	On	On
Convert EAN-13 to ISBN	3B	On/Off	On or Off	Off	Off	Off
Send EAN/JAN checksum	3F	On/Off	On or Off	Off	Off	Off

table continues

Code Parameter	I.D. #	Type	Acceptable Input	Defaults		
				D0	D1	D2
UPC, EAN, JAN Extensions				<i>Enter 1 for On and 0 for Off.</i>		
Allow 2-digit extensions	3C	On/Off	On or Off	Off	On	On
Allow 5-digit extensions	3D	On/Off	On or Off	Off	On	On
Require extensions	3E	On/Off	On or Off	Off	Off	Off
Code 128				<i>Enter 1 for On and 0 for Off.</i>		
Enable	40	On/Off	On or Off	Off	On	On
Minimum length	41	Value	01-50	01	01	02
Maximum length	42	Value	01-50	01	50	20
Enable UCC/EAN 128	43	On/Off	On or Off	Off	Off	Off
Labelcode 4/5				<i>Enter 1 for On and 0 for Off.</i>		
Enable	52	On/Off	On or Off	Off	On	Off
Convert	53	On/Off	On or Off	Off	Off	Off
Other Controls				<i>Enter 1 for On and 0 for Off.</i>		
Laser programming enable	B0	On/Off	On or Off	On	On	On
Autoterminator	B1	Value	<i>Any single ASCII character (00 = Off)</i>	(CR)	(CR)	(CR)
Auto-off timer	B2	Value	<i>01-99 (in 4-sec. increments) (00 = Off)</i>	75	75	75
Send assigned symbology identifiers*	B5	On/Off	On or Off	Off	On	Off
*A = UPC-A D = M 2 of 5 G = EAN-8 J = Code 11 M = EAN-13 B = I 2 of 5 E = UPC-E H = MSI K = Code 128 N = Labelcode 4/5 C = Code 39 F = S 2 of 5 I = Codabar L = Code 93 O = Ames						
Good-read beep tone	B8	Value	00 = 2400 Hz 01 = 2600 Hz 02 = 2800 Hz 03 = 3000 Hz 04 = 3200 Hz 05 = 3400 Hz 06 = 3600 Hz 07 = 3800 Hz	00	00	00
Number of good-read beeps	B9	Value	01-04	01	01	01
Good-read beep duration	BA	Value	00 = 0.07 sec. 01 = 0.13 sec. 02 = 0.18 sec. 03 = 0.36 sec.	00	00	00

table continues

Programming Parameters

Code Parameter	I.D. #	Type	Acceptable Input	Defaults		
				D0	D1	D2
Beeper volume	BC	Value	00 = Off 01–09 (01 = Lowest; 07 = Highest; 08 = Higher; 09 = Lower)	07	07	07
Error beep tone	BD	Value	00 = 2400 Hz 01 = 2580 Hz 02 = 2770 Hz 03 = 3000 Hz 04 = 3270 Hz 05 = 3600 Hz 06 = 4000 Hz 07 = 4520 Hz	01	01	01
Long-range trigger mode (<i>long-range scanner models only</i>)	D4	On/Off	On = Release Scan Off = Spot Timeout	Off	Off	Off
Spot beam timeout (<i>long-range scanner models only</i>)	D5	Value	00 = 0.25 sec. 01 = 0.5 sec. 02 = 1 sec. 03 = 1.5 sec. 04 = 2 sec.	01	01	01
Release scan timeout (<i>long-range scanner models only</i>)	D6	Value	01–30 (<i>seconds</i>)	02	02	02
Keypress sound	DD	Value	00 = Off 01 = Click 02 = Beep	01	01	01
Enable CTRL-ALT-DEL reboot	E0	On/Off	On or Off	Off	Off	On
Enable trigger programmability	E1	On/Off	On or Off	On	On	On
Backlight auto-off timeout	E2	Value	01–99 (<i>seconds</i>) (00 = Off)	15	15	15

>> Appendix B

Connector Configurations

Cables for all Falcon products are available from Percon and its resellers. However, you may want to make your own cables for custom uses. This chapter provides information about Falcon cables, along with diagrams and tables describing the configurations of the connectors for Falcon DOS portable terminals, the Falcon Dock, and the Falcon 4-Slot Dock.

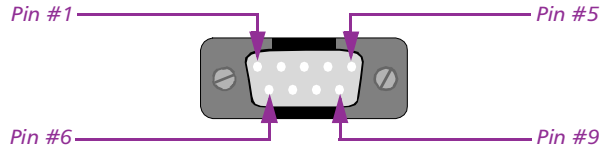
CHAPTER CONTENTS

<i>Falcon Portable Terminals</i>	<118>
<i>Laser-Scanner Connector</i>	
<i>Serial Port Jack</i>	
<i>Falcon Dock</i>	<119>
<i>25-Pin Connector</i>	
<i>Cradle Jack</i>	
<i>Falcon 4-Slot Dock</i>	<120>
<i>Host-Interface Cable</i>	
<i>Dock-Network Cable Jacks</i>	

>> Falcon Portable Terminals

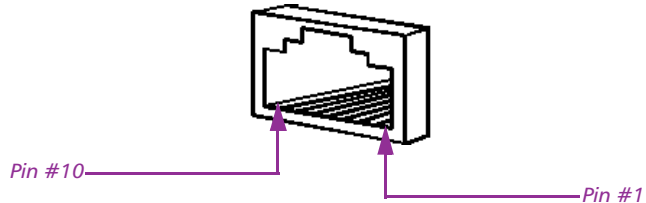
The Falcon portable terminals have a standard 9-pin connector for scanner input at the top and a 10-pin telephone-style jack in the serial port at the bottom.

Laser-Scanner Connector



Pin #	Signal
1	Laser sync in
2	Bar code data in
3	Good-read out
4	Scan-enable out
5	Laser trigger in
6	LSE out
7	GND
8	GND
9	5V out

Serial Port Jack



Pin #	Signal
1	DCD in
2	5V out
3	RXD
4	HS out
5	TXD

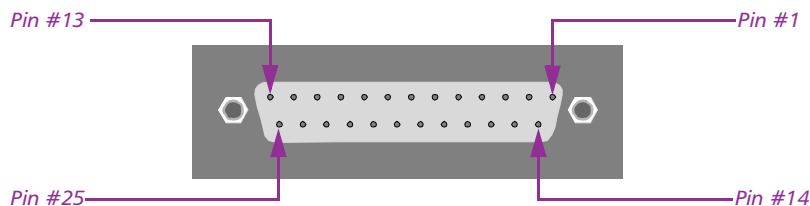
table continues

Pin #	Signal
6	Battery
7	Ring in
8	GND
9	HS in
10	GND

>> Falcon Dock

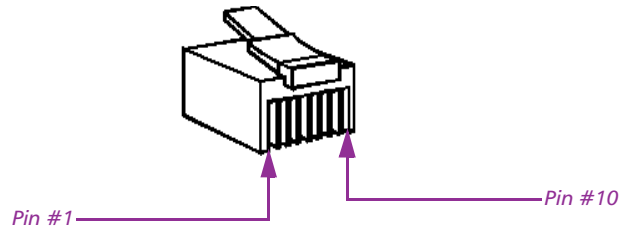
The Falcon Dock has a 25-pin dock-to-host cable connector on the back panel and a 10-pin telephone-style plug inside the cradle.

25-Pin Connector



Pin #	Signal
1	GND
2	TXD
3	RXD
4	RTS
5	CTS
6	DSR
7	GND
8	DCD
9-19	Unconnected
20	DTR
21	Unconnected
22	Ring in
23-25	Unconnected

Cradle Jack



Pin #	Signal
1	DCD in
2	5V out
3	RXD
4	HS out
5	TXD
6	Battery
7	Ring in
8	GND
9	HS in
10	GND

>> Falcon 4-Slot Dock

The Falcon 4-Slot Dock has a 10-pin telephone-style plug inside each cradle that is identical to the one in the Falcon Dock cradle (see above).

On the back panel is a 25-pin dock-to-host cable connector identical to the one on the back of the Falcon Dock (see page [119](#)). The wiring for the cable is described below.

Also on the back panel of the Falcon 4-Slot Dock are two dock-network cable jacks:

- A 10-pin telephone-style connector for upstream communications (from the dock to the host, or from one dock to another dock closer to the host)
- An 8-pin telephone-style connector for downstream communications (from the host to the dock, or from one dock to another dock farther from the host)

Wiring for these jacks is described at the end of this section.

Host-Interface Cable

The wiring for the Falcon 4-Slot Dock RS-232 host-interface cable is shown in the following table. All signals in the same row are tied together. The standard cable is 6 feet (2 meters) long.

Host Pin #		Host Signal	4SD Pin # (25-pin)	Notes
(9-pin)	(25-pin)			
shell	1/shell	Chassis GND	1/shell	
3	2	HostTxD#	3	
2	3	HostRxD#	2	
7	4	HostRTS	5	Tied together (also to pin 8, DCD)
8	5	HostCTS	4	
6	6	HostDSR	20	Tied together; must be active to select RS-232 interface
4	20	Host DTR	6	
5	7	GND	7	

Dock-Network Cable Jacks

The wiring for the dock-network cable jacks are shown in the following table. The network signals are differential type and are routed on twisted-pair wiring. The pinouts are assigned so as to accommodate certain off-the-shelf cables (e.g., L-COM #TRD855-* EIA568 Flex patch cables).

In the signal names, "in" and "out" refer to the logical direction of the signal with respect to the 4-Slot Dock. Note that Vcc and GND are present only on the upstream connector; they are reserved for future networking options (e.g., in-line RS-232/422 converter, RS-422 line repeater).

Pin #	Signal	
	Upstream Connector	Downstream Connector
1	Vcc	HostXmtOut+
2	HostXmtIn+	HostXmtOut-
3	HostXmtIn-	ReqIn+
4	ReqOut+	HostRcvIn+
5	HostRcvOut+	HostRcvIn-
6	HostRcvOut-	ReqIn-
7	ReqOut-	InhOut+
8	InhIn+	InhOut-
9	InhIn-	—
10	GND	—

>> Appendix C

Bar Codes for Configuring Falcon

This chapter provides bar codes for common setup parameters for programming the Falcon. Factory default settings are underlined.

To make settings that aren't provided here, you can use your own bar codes or use the Falcon Configuration Utility to program the Falcon (see chapter [2](#)).

CHAPTER CONTENTS

<i>Predefined Defaults</i>	<124>
<i>Code 39</i>	<124>
<i>Interleaved 2 of 5</i>	<125>
<i>Matrix 2 of 5</i>	<126>
<i>Standard 2 of 5</i>	<127>
<i>Code 11</i>	<128>
<i>Codabar/Ames</i>	<128>
<i>MSI</i>	<129>
<i>Code 93</i>	<130>
<i>Code 128</i>	<131>
<i>Labelcode 4/5</i>	<132>
<i>UPC-A</i>	<132>
<i>UPC-E</i>	<132>
<i>EAN/JAN</i>	<133>
<i>UPC/EAN/JAN Extensions</i>	<133>
<i>Other Controls</i>	<134>

>> Predefined Defaults

D2



>> Code 39

ENABLE

On



Off



MINIMUM LENGTH

0



10



20



30



40



50



MAXIMUM LENGTH

1



10



20



30



40



50



ENABLE CHECKSUM

On



Off



SEND CHECKSUM

On



Off



FULL ASCII MODE

On



Off



>> Interleaved 2 of 5

ENABLE

On



Off



MINIMUM LENGTH

2



6



10



20



30



40



50



MAXIMUM LENGTH

2



10



20



30



40



50



ENABLE CHECKSUM

On



Off



SEND CHECKSUM

On



Off



USE LENGTHS 6 AND 14 ONLY (case code)

On



Off



>> Matrix 2 of 5

ENABLE

On



Off



MINIMUM LENGTH

1



6



10



20



30



40



50



MAXIMUM LENGTH

1



10



20



30



40



50



ENABLE CHECKSUM

On



Off



SEND CHECKSUM

On



Off



>> Standard 2 of 5

ENABLE

On



Off



MINIMUM LENGTH

1



6



10



20



30



40



50



MAXIMUM LENGTH

1



10



20



30



40



50



ENABLE CHECKSUM

On



Off



SEND CHECKSUM

On



Off



USE 2-BAR START/STOP

On



Off



>> Code 11

ENABLE

On



Off



MINIMUM LENGTH

4



10



20



30



40



50



MAXIMUM LENGTH

1



10



20



30



40



50



REQUIRE 2 CHECK DIGITS

On



Off



SEND CHECK DIGIT(S)

On



Off



>> Codabar/Ames

ENABLE

On



Off



MINIMUM LENGTH

1



4



10



20



30



40



50

**MAXIMUM LENGTH**

1



10



20



30



40









50







**SEND STOP/START****On****Off****CODABAR-TO-CLSI CONVERSION****On****Off****WIDE INTERCHARACTER GAPS ALLOWED****On****Off****>> MSI****ENABLE****On****Off**

Bar Codes for Configuring Falcon

MINIMUM LENGTH

1	 * \$ + \$ - 2 7 0 1 E E *
4	 * \$ + \$ - 2 7 0 4 E E *
7	 * \$ + \$ - 2 7 0 7 E E *
10	 * \$ + \$ - 2 7 1 0 E E *
12	 * \$ + \$ - 2 7 1 2 E E *
14	 * \$ + \$ - 2 7 1 4 E E *

MAXIMUM LENGTH

1	 * \$ + \$ - 2 8 0 1 E E *
4	 * \$ + \$ - 2 8 0 4 E E *
7	 * \$ + \$ - 2 8 0 7 E E *
10	 * \$ + \$ - 2 8 1 0 E E *
12	 * \$ + \$ - 2 8 1 2 E E *
14	 * \$ + \$ - 2 8 1 4 E E *



REQUIRE 2 CHECK DIGITS

On	 * \$ + \$ - 2 9 1 E E *
Off	 * \$ + \$ - 2 9 0 E E *

2ND CHECK DIGIT MOD 11



On	 * \$ + \$ - 2 A 1 E E *
Off	 * \$ + \$ - 2 A 0 E E *

SEND CHECK DIGIT(S)



On	 * \$ + \$ - 2 B 1 E E *
Off	 * \$ + \$ - 2 B 0 E E *





>> Code 93

ENABLE







On	 * \$ + \$ - 2 C 1 E E *
Off	 * \$ + \$ - 2 C 0 E E *

MINIMUM LENGTH



2	 * \$ + \$ - 2 D 0 2 E E *
10	 * \$ + \$ - 2 D 1 0 E E *

20	
	* \$ + \$ - 2 D 2 0 E E *
30	
	* \$ + \$ - 2 D 3 0 E E *
40	
	* \$ + \$ - 2 D 4 0 E E *
50	
	* \$ + \$ - 2 D 5 0 E E *








MAXIMUM LENGTH

2	
	* \$ + \$ - 2 E 0 2 E E *
10	
	* \$ + \$ - 2 E 1 0 E E *
<u>20</u>	
	* \$ + \$ - 2 E 2 0 E E *
30	
	* \$ + \$ - 2 E 3 0 E E *
40	
	* \$ + \$ - 2 E 4 0 E E *
50	
	* \$ + \$ - 2 E 5 0 E E *







>> Code 128**ENABLE**

<u>On</u>	
	* \$ + \$ - 4 0 1 E E *
Off	
	* \$ + \$ - 4 0 0 E E *

MINIMUM LENGTH

1	
	* \$ + \$ - 4 1 0 1 E E *
<u>2</u>	
	* \$ + \$ - 4 1 0 2 E E *
10	
	* \$ + \$ - 4 1 1 0 E E *
20	
	* \$ + \$ - 4 1 2 0 E E *
30	
	* \$ + \$ - 4 1 3 0 E E *
40	
	* \$ + \$ - 4 1 4 0 E E *
50	
	* \$ + \$ - 4 1 5 0 E E *

MAXIMUM LENGTH

1	
	* \$ + \$ - 4 2 0 1 E E *
10	
	* \$ + \$ - 4 2 1 0 E E *
<u>20</u>	
	* \$ + \$ - 4 2 2 0 E E *
30	
	* \$ + \$ - 4 2 3 0 E E *
40	
	* \$ + \$ - 4 2 4 0 E E *
50	
	* \$ + \$ - 4 2 5 0 E E *

Bar Codes for Configuring Falcon

ENABLE UCC/EAN 128

On



Off



>> Labelcode 4/5

ENABLE

On



Off



CONVERT

On



Off



>> UPC-A

ENABLE UPC-A

On



Off



SEND SYSTEM DIGIT

On



Off



SEND CHECK DIGIT

On



Off



CONVERT UPC-A TO EAN-13

On



Off



>> UPC-E

USE SYSTEM DIGIT 0

On



Off



USE SYSTEM DIGIT 1

On



Off



CONVERT UPC-E TO UPC-A

On 
* \$ + \$ - 3 6 1 E E *

Off 
* \$ + \$ - 3 6 0 E E *

SEND SYSTEM DIGIT

On 
* \$ + \$ - 3 7 1 E E *

Off 
* \$ + \$ - 3 7 0 E E *

SEND CHECK DIGIT

On 
* \$ + \$ - 3 8 1 E E *

Off 
* \$ + \$ - 3 8 0 E E *

>> EAN/JAN

ENABLE EAN-8/JAN-8

On 
* \$ + \$ - 3 9 1 E E *

Off 
* \$ + \$ - 3 9 0 E E *

ENABLE EAN-13/JAN-13

On 
* \$ + \$ - 3 A 1 E E *

Off 
* \$ + \$ - 3 A 0 E E *

CONVERT EAN-13 TO ISBN

On 
* \$ + \$ - 3 B 1 E E *

Off 
* \$ + \$ - 3 B 0 E E *

SEND EAN/JAN CHECKSUM

On 
* \$ + \$ - 3 F 1 E E *

Off 
* \$ + \$ - 3 F 0 E E *

>> UPC/EAN/JAN Extensions

ALLOW 2-DIGIT EXTENSIONS

On 
* \$ + \$ - 3 C 1 E E *

Off 
* \$ + \$ - 3 C 0 E E *

ALLOW 5-DIGIT EXTENSIONS

On 
* \$ + \$ - 3 D 1 E E *

Off 
* \$ + \$ - 3 D 0 E E *

REQUIRE EXTENSIONS

On 
* \$ + \$ - 3 E 1 E E *

Off 
* \$ + \$ - 3 E 0 E E *

>> Other Controls

AUTOTERMINATOR

(CR)



(TAB)



None



AUTO-OFF TIMER

Off



20 sec.



40 sec.



1 min.



2 min.



3 min.



4 min.



5 min.



6 min.



SEND SYMBOLOGY IDENTIFIER

On



Off



GOOD-READ BEEP TONE (in Hertz)

2400



2800



3200



3600



NUMBER OF GOOD-READ BEEPS

1



2



3



4



GOOD-READ BEEP DURATION (in seconds)

0.07



0.13



0.18 
* \$ + \$ - B A 0 2 E E *

0.36 
* \$ + \$ - B A 0 3 E E *

BEEPER VOLUME

Off 
* \$ + \$ - B C 0 0 E E *

Lowest 
* \$ + \$ - B C 0 1 E E *

Medium 
* \$ + \$ - B C 0 4 E E *

Highest 
* \$ + \$ - B C 0 7 E E *

Higher 
* \$ + \$ - B C 0 8 E E *

Lower 
* \$ + \$ - B C 0 9 E E *

ERROR BEEP TONE (in Hertz)

2580 
* \$ + \$ - B D 0 1 E E *

3000 
* \$ + \$ - B D 0 3 E E *

3600 
* \$ + \$ - B D 0 5 E E *

4520 
* \$ + \$ - B D 0 7 E E *

LONG-RANGE TRIGGER MODE

(On = Release Scan; Off = Spot Timeout)

On 
* \$ + \$ - D 4 1 E E *

Off 
* \$ + \$ - D 4 0 E E *

SPOT BEAM TIMEOUT (in seconds)

0.25 
* \$ + \$ - D 5 0 0 E E *

0.5 
* \$ + \$ - D 5 0 1 E E *

1 
* \$ + \$ - D 5 0 2 E E *

1.5 
* \$ + \$ - D 5 0 3 E E *

2 
* \$ + \$ - D 5 0 4 E E *

RELEASE SCAN TIMEOUT (in seconds)

1 
* \$ + \$ - D 6 0 1 E E *

2 
* \$ + \$ - D 6 0 2 E E *

3 
* \$ + \$ - D 6 0 3 E E *

5 
* \$ + \$ - D 6 0 5 E E *

10 
* \$ + \$ - D 6 1 0 E E *

30 
* \$ + \$ - D 6 3 0 E E *

Bar Codes for Configuring Falcon

KEYPRESS SOUND

Off



Click



Beep



BACKLIGHT AUTO-OFF TIMEOUT *(in seconds)*

Off



2



5



7



9



15



20



30



Index

; (DOS command) [90](#)
@ (DOS command) [90](#)

A

Ames
 bar codes for [128–129](#)
 programming parameters for [114](#)
archive [98](#)
ATA memory cards [2–6](#)
 installing drivers with Falcon Configuration
 Utility [22](#)
ATTRIB [90, 93](#)
attribute [98](#)
auto-off timer
 bar codes for [134](#)
 settings for [115](#)
autoterminator
 bar codes for [134](#)
 settings for [115](#)

B

backlight
 timeout [116](#)
 bar codes for [136](#)
bar codes for programming [123–136](#)
baud rate, selecting with the configuration
 utility [29](#)

beeper options [115–116](#)
 bar codes for [134–135](#)
BREAK [90](#)
BUFFERS [94](#)

C

CALL [90](#)
cards, PC [2–6](#)
 inserting [4–5](#)
 installing drivers with Falcon Configuration
 Utility [22](#)
 recognition and configuration [5](#)
 removing [6](#)
CHDIR [90](#)
CHKDSK [90, 94](#)
Chkdsk [95](#)
CLS [90](#)
Codabar
 bar codes for [128–129](#)
 programming parameters for [114](#)
 wide intercharacter gaps option [114](#)
Code 11
 bar codes for [128](#)
 programming parameters for [113](#)
Code 128
 bar codes for [131–132](#)
 programming parameters for [115](#)

Code 39

- bar codes for [124](#)
- programming parameters for [113](#)

Code 93

- bar codes for [130–131](#)
- settings for [114](#)

Comm Settings dialog box (in Falcon Configuration Utility) [29](#)

COMMAND [90](#), [96](#)

Commands

- ATTRIB [93](#)
- BUFFERS [94](#)
- CHKDSK [94](#)
- COMMAND [96](#)
- DIR [97](#)
- FCBS [100](#)
- FIND [100](#)
- HELP [101](#)
- NEWFILE [102](#)
- PRINT [103](#)
- SHARE [104](#)
- SWITCHES [105](#)
- TREE [106](#)
- VER [106](#)
- XCOPY [107](#)

communications, serial

- port [6](#)
- selecting with configuration utility [29](#)
- transfer utility (XFER) [8](#), [36](#)

computer, transferring files to and from [8](#), [36](#)

CONFIG.SYS

- BUFFERS [94](#)
- FCBS [100](#)
- NEWFILE [102](#)
- SWITCHES [105](#)

configuration of PC cards [5](#)

configuration utility see [Falcon Configuration Utility](#)

configuring the Falcon [8](#), [11–29](#)

conversion options

- EAN-13 to ISBN [114](#)
- Labelcode 4/5 [115](#)
- UPC-A to EAN-13 [114](#)
- UPC-E to UPC-A [114](#)

COPY [90](#)

CTTY [90](#)

D

D2 (predefined default set), bar code for [124](#)

DATE [90](#)

defaults (predefined), bar code for [124](#)

DEL [90](#)

DIR [97](#)

DIRCMD [99](#)

Directory [97](#)

Disk checking [95](#)

disk drives [Z](#)

DOS files, installing with Falcon Configuration Utility [22](#)

DOS, transferring files in [8](#), [36](#)

drives [Z](#)

drop-down lists (in Falcon Configuration Utility) [29](#)

EEAN-13, converting from UPC-A [114](#)ECHO [90](#)ERASE [90](#)

European Article Numbering (EAN)

bar codes for [133](#)extensions [115](#), [133](#)programming parameters for [114](#)EXIT [90](#)extensions (UPC, EAN, and JAN) [133](#)programming parameters for [115](#)**F**

Falcon

automatic shutoff

bar codes for [134](#)settings for [115](#)communications port [6](#)configuring [8](#), [11–29](#)drives [7](#)

laser

long-range

bar codes for [135](#)settings for [116](#)predefined defaults, bar code for [124](#)serial port [6](#)transferring files to and from [8](#), [36](#)Falcon Configuration Utility [8](#), [11–29](#)Add button [19](#), [31](#)AUTOEXEC.BAT button [25](#)baud rate setting [29](#)Browse button [19](#), [20](#), [22](#), [25](#), [27](#), [28](#), [31](#),
[32](#)Comm Port setting [29](#)Comm Settings button [14](#), [17](#)Comm Settings dialog box [29](#)CONFIG.SYS button [25](#)Custom button [14](#)Custom Configuration Menu [15](#)Default button [13](#)Delete button [33](#)Done button [24](#), [25](#), [27](#), [29](#), [33](#)Download button [17](#)drop-down lists [29](#)Edit button [32](#)File Configuration button [16](#)File Configuration windows [18–25](#)File Transfer window [30–33](#)input fields [27](#), [29](#)installing [12](#)Main Menu [13](#)More button [22](#)Next button [23](#), [27](#)on/off switches [27](#), [28](#)Prev button [23](#), [25](#), [27](#), [29](#)Program Settings button [16](#)Program Settings windows [26–29](#)radio buttons [29](#)Receive button [33](#)Send button [33](#)starting [12](#)Text File button [25](#)Transfer Files button [14](#)FAT [94](#)FAT Checking [95](#)fax/modem cards [2–6](#)installing drivers with Falcon Configuration
Utility [22](#)

Index

- FCB [100](#)
 - FCBS [100](#)
 - File
 - Locking [104](#)
 - Sharing [104](#)
 - File Configuration windows (in Falcon Configuration Utility) [18–25](#)
 - File Control Block [100](#)
 - File Transfer window (in Falcon Configuration Utility) [30–33](#)
 - FIND [90](#), [100](#)
 - flash memory cards [2–6](#)
 - installing drivers with Falcon Configuration Utility [22](#)
 - FOR [90](#)
- G**
- GOTO [90](#)
- H**
- HELP [90](#), [101](#)
 - hidden file [98](#)
- I**
- I/O cards [2–6](#)
 - installing drivers with Falcon Configuration Utility [22](#)
 - identifying symbologies [115](#)
 - bar codes for [134](#)
 - IF [90](#)
 - input fields (in Falcon Configuration Utility) [27](#), [29](#)
 - installing the Falcon Configuration Utility [12](#)
 - integrated laser
 - long-range
 - bar codes for [135](#)
 - settings for [116](#)
 - Interleaved 2 of 5
 - bar codes for [125](#)
 - programming parameters for [113](#)
- J**
- Japan Article Numbering (JAN)
 - bar codes for [133](#)
 - extensions [115](#), [133](#)
 - programming parameters for [114](#)
- K**
- keypress sound
 - bar codes for [136](#)
 - programming [116](#)
- L**
- Labelcode 4/5
 - bar codes for [132](#)
 - programming parameters for [115](#)
 - laser
 - long-range
 - bar codes for [135](#)
 - settings for [116](#)
 - long-range laser
 - bar codes for [135](#)
 - settings for [116](#)

M

Matrix 2 of 5
bar codes for [126](#)
programming parameters for [113](#)
memory cards [2-6](#)
installing drivers with Falcon Configuration Utility [22](#)
MKDIR [91](#)
MSI
bar codes for [129-130](#)
programming parameters for [114](#)

N

NEWFILE [102](#)

O

on/off switches (in Falcon Configuration Utility) [27](#), [28](#)

P

PATH [91](#)
PAUSE [91](#)
PC card slot [3-6](#)
removing the cover [3](#)
PC cards [2-6](#)
inserting [4-5](#)
installing drivers with Falcon Configuration Utility [22](#)
recognition and configuration [5](#)
removing [6](#)
PC, transferring files to and from [8](#), [36](#)
PCMCIA cards see [PC cards](#)

port, serial [6](#)
selecting with the configuration utility [29](#)
predefined defaults, bar code for [124](#)
PRINT [91](#), [103](#)
Program Settings windows (in Falcon Configuration Utility) [26-29](#)
programming laser triggers [116](#)
programming, bar codes for [123-136](#)
PROMPT [91](#)

R

radio buttons (in Falcon Configuration Utility) [29](#)
read only file [98](#)
recognition of PC cards [5](#)
REM [91](#)
REN (rename) [91](#)
RMDIR (Remove Directory) [91](#)

S

serial communications
transfer utility (XFER) [8](#), [36](#)
serial port [6](#)
selecting with the configuration utility [29](#)
SET [91](#)
SHARE [91](#), [104](#)
Shell [96](#)
SHIFT [91](#)
shutoff, automatic
bar codes for [134](#)
settings for [115](#)
SORT [91](#)

Standard 2 of 5

bar codes for [127](#)

programming parameters for [113](#)

starting the Falcon Configuration Utility [12](#)

sub-directory [98](#)

SWITCHES [105](#)

symbology identifier option [115](#)

bar codes for [134](#)

system file [98](#)

T

TIME [91](#)

timeout

automatic shutoff

bar codes for [134](#)

settings for [115](#)

backlight [116](#)

bar codes for [136](#)

timer, auto-off

bar codes for [134](#)

settings for [115](#)

transfer utility (XFER) [8](#), [36](#)

transferring files between Falcon and PC [8](#),
[36](#)

TREE [91](#), [106](#)

triggers, laser

enabling programmability [116](#)

TYPE [91](#)

U

Universal Product Code (UPC)

bar codes for [132–133](#)

conversion options [114](#)

extensions [133](#)

programming parameters for [115](#)

UPC-A

converting from UPC-E [114](#)

converting to EAN-13 [114](#)

programming parameters for [114](#)

UPC-E

converting to UPC-A [114](#)

programming parameters for [114](#)

V

VER [91](#), [106](#)

VERIFY [91](#)

W

wide intercharacter gaps option for
Codabar [114](#)

X

XCOPY [91](#), [107](#)

XDEL [91](#), [108](#)

XFER [8](#), [36](#)