

Technical Reference

**6220 Hand-Held
Computer**

► NOTICE

The information contained herein is proprietary and is provided solely for the purpose of allowing customers to operate and service Intermec manufactured equipment and is not to be released, reproduced, or used for any other purpose without written permission of Intermec.

Disclaimer of Warranties. The sample source code included in this document is presented for reference only. The code does not necessarily represent complete, tested programs. The code is provided “**AS IS WITH ALL FAULTS.**” **ALL WARRANTIES ARE EXPRESSLY DISCLAIMED, INCLUDING THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.**

We welcome your comments concerning this publication. Although every effort has been made to keep it free of errors, some may occur. When reporting a specific problem, please describe it briefly and include the book title and part number, as well as the paragraph or figure number and the page number.

Send your comments to:
Intermec Technologies Corporation
Publications Department
550 Second Street SE
Cedar Rapids, IA 52401

ANTARES, INTERMEC, NORAND, NOR*WARE, PEN*KEY, ROUTEPOWER, TRAKKER, and TRAKKER ANTARES are registered trademarks and ENTERPRISE WIRELESS LAN, INCA, Mobile Framework, TE 2000, UAP, and UNIVERSAL ACCESS POINT are trademarks of Intermec Technologies Corporation.

© 1999 Intermec Technologies Corporation. All rights reserved.

Acknowledgments

ActiveX, Microsoft, MS, and MS-DOS, Windows, and Windows NT are registered trademarks and *MSDN, Visual Basic, Visual C++, and Windows for Pen* are trademarks of Microsoft Corporation.

AS/400, IBM, IBM PC, Micro Channel, PS/2, and OS/2 are registered trademarks and *AIX and Presentation Manager* are trademarks of International Business Machines Corporation.

Intel is a registered trademark of Intel Corporation.

SanDisk is a trademark of SanDisk Corporation.

XEROX is a registered trademark of Xerox Corporation.

XMS is a registered trademark of Lotus, Intel, Microsoft, and AST Research.

CONTENTS



SECTION 1

Introduction

About this Manual	1-1
Introduction to the 6220 Computer	1-1
Batteries	1-2
Main Battery	1-2
Backup Battery	1-2
Battery Check	1-3
Connectors	1-3
Display	1-3
Backlight	1-3
Touch Screen	1-3
Docks and Chargers	1-4
Keyboard	1-4
Keyboard Remapping	1-4
Suspend/Resume	1-4
Memory	1-4
CMOS	1-4
System Memory (DRAM)	1-4
Flash ROM	1-4
PC Cards	1-4
Open System Environment	1-5
Ports	1-5
Power Management	1-6
Printers	1-6
Processor	1-6
Reset Switch	1-6
Scanners	1-6
System BIOS	1-7
Related Publications	1-7
Customer Support	1-8
Factory Service	1-8
Customer Support Center	1-8
Web Site	1-8
Bulletin Board Service	1-8

SECTION 2

DOS Device Drivers

APM Clock Driver	2-1
BGI Support	2-1
Using N6220.BGI Driver	2-1
Bitmap Text Output	2-2
BGI Demonstration Files	2-2

DOS Utilities	2-2
Configuration Utility	2-2
Remapping Keyboard	2-2
Configuring Power Management	2-2
Using SC400CFG	2-2
Create Download Include File	2-3
Drive Mapping Utility	2-4
File Integrity Verification Utility	2-4
Get or Set ROM DOS Boot Flags	2-5
Multipurpose Delay Utility	2-5
Reset the System	2-6
RAM Drive	2-7
IrDA Printer	2-8
LAN Communications	2-8
NPCP Printer	2-9
Installation and Configuration	2-9
Required CONFIG.SYS Entry	2-9
Required AUTOEXEC.BAT or CONFIG.SYS Entry	2-9
Functionality and Usage	2-10
PC Card Services	2-11
Pen Calibration	2-12
Configuration	2-12
Required Calibration Files	2-12
Operation	2-12
Pen Driver	2-13
PenPrint	2-13
Installation and Configuration	2-13
Return Codes	2-14
Usage Notes	2-14
Power Management	2-15
Screen Emulation	2-16
Command Line Switches	2-16
Font File Format	2-16
Interrupts Supported by FONTMAP.EXE	2-17
Unsupported 4000 Series Video Interrupts	2-17
Tethered Scanner	2-18
Installation and Configuration	2-18
Usage	2-18
Command Line Switches	2-19
Upper Memory Provider	2-20

SECTION 3

Converting Applications

Converting to the 6000 Series Computers	3-1
Files No Longer Supported	3-1
Changed Files	3-2
New 6000 Series Files	3-3
Unchanged Files	3-3
4000 Series Programming Interfaces	3-4
Installation and Configuration	3-4
Command Line Switches	3-4

SECTION 4**System Information**

Binary Image Files Posted to Bulletin Board	4-1
Flash Release Contents	4-2
Non-Ethernet 6220FL2A	4-2
Ethernet 6220FL2B	4-2
Updating Flash Using an Application Soft-Download	4-2
Prerequisites	4-3
Procedure	4-3
Updating Flash using INTERLNK	4-3
Prerequisites	4-3
Procedure	4-3
Updating Flash Using an ATA Card	4-4
Prerequisites	4-4
Procedure to Build an ATA Card	4-4
Procedure to Reflash with an ATA Card	4-4
Updating Flash Using a 4 MB SRAM Card	4-5
Prerequisites	4-5
Procedure to Build IFL SRAM Card	4-5
Procedure to Reflash with an IFL Card	4-5
Recommended Methods	4-6
Partial Flash Download	4-6
ATA Boot Card	4-6
SRAM Boot Card	4-6
Full Flash Download	4-6

SECTION 5**Communications**

Using INTERLNK and INTERSVR	5-1
Norand Utilities	5-1
Initial Program Load	5-1
NPCP Network	5-2
NRInet	5-2
TFTP	5-2
TTY	5-3
Calling Norand Utilities from Application	5-3
System Setup Requirements	5-3
NPCP	5-3
TTY	5-3
NRInet Using PSROMOC Version 3.xx	5-4
NRInet Using PSROMOC Version 2.xx	5-4
Valid Control File Parameters	5-5
Upload Control File Parameters	5-7
Communications Log File	5-8
Protocol Errors	5-9
TCOM Session	5-12
Session Control File	5-12
Download Request File	5-12
Upload and Download Files	5-13
PL/N File Descriptor for Binary Files	5-14

FIGURES

Figure 1-1 Hand-Held Computers (front view)	1-2
Figure 1-2 Main Battery Pack (bottom view) Showing Contacts	1-3
Figure 1-3 Location of Components (rear, top)	1-5
Figure 1-4 Location of Components (top end)	1-6
Figure 2-1 Power Management Software	2-15

TABLES

Table 2-1 MMBFLAG.COM Error Levels	2-5
Table 2-2 FONTMAP Interrupts Supported	2-17
Table 4-1 6220FL2A BBS Equivalents	4-1
Table 5-1 NPCP Protocol Errors	5-9
Table 5-2 MININET Protocol Errors	5-9
Table 5-3 NRInet Protocol Errors	5-10
Table 5-4 TTY Protocol Errors	5-11

GLOSSARY

INDEX

Section 1

Introduction



This publication contains information for developing applications for the PEN*KEY[®] 6220 Hand-Held Computer from Intermec Technologies Corporation.

The majority of this book is for experienced application programmers. However, developing applications for this computer should be similar to developing DOS applications for any standard PC.

About this Manual

This manual contains the following sections.

- ▶ **Section 1 — Introduction**
Provides technical information about the 6220 Computer, related publications, and customer service information.
- ▶ **Section 2 — DOS Device Drivers**
Contains installation and configuration information for DOS device drivers, including communications and DOS Utilities.
- ▶ **Section 3 — Converting Applications**
Details information applicable for converting 4000 Series applications for use on a 6000 Series Computer.
- ▶ **Section 4 — System Information**
Defines system information included with the 6220 Computer.
- ▶ **Section 5 — Communications**
Describes communications supported for this computer.

Introduction to the 6220 Computer

The 6220 Computer has a number of features that are useful in the mobile systems marketplace, such as:

- ▶ Lithium ion smart battery for full-day battery life.
- ▶ Power managed for longer battery life.
- ▶ Compatible with existing Intermec docks, chargers, printers, and other peripherals.
- ▶ Optional touch screen for signature capture.
- ▶ Backlit display.

Figure 1-1 shows a front view of each of the two models.

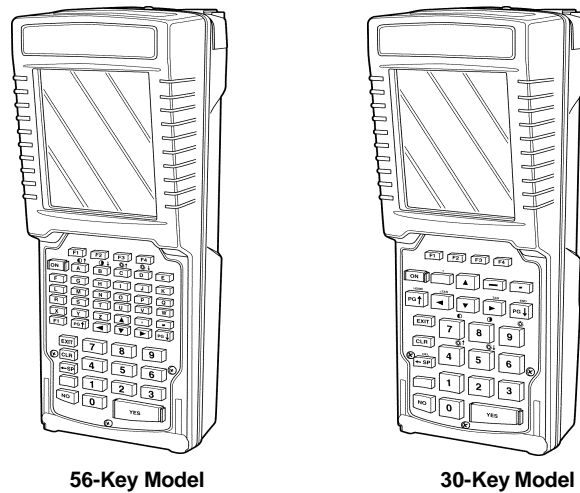


Figure 1-1
Hand-Held Computers (front view)

From the viewpoint of an application, the 6220 Computer is like other PC compatibles, with a few exceptions, as described in the following paragraphs.

Batteries

▼ CAUTION:

This computer has **ONE** primary power source, the **MAIN Battery**. The **Backup Battery** maintains power to RAM while changing Main Batteries. At all times during operation and while stored (overnight, long periods of time, and during suspend), the unit must have its Main Battery installed.

Main Battery

The main battery is a 1350 mAH lithium ion power pack, and has a useful life of 500 charge/discharge cycles. The battery pack contains a smart processor for the system software to report battery status.

In suspend mode, the main battery maintains RAM, RTC, and CMOS settings. When the computer is off charge, the main battery also supplies needed charge to the backup battery. The main battery must always be installed, except when replacing batteries.

Be sure the batteries are fully charged before leaving them off charge for long periods of time. A unit, with fully charged main and backup batteries, can be left off charge overnight or for a long weekend. However, if the computer goes into shutdown mode as a result of a low main battery, data remains protected by the backup battery.

Backup Battery

The backup battery has a small nickel-cadmium (NiCd) battery and is on trickle-charge when the unit is powered from the main battery or connected to a charger. It is unable to maintain DRAM contents for an extended time, with the expectation of supporting the system without a main battery. If the main battery is drained completely or is not reinstalled, DRAM data is lost. When fully charged, the backup battery retains data in the DRAM for about ten hours.

Battery Check

Check the main battery capacity by touching two contacts at the same time, as shown in Figure 1-2. Another method of checking the battery is to use a software fuel gauge based on the APM 1.1 specification.

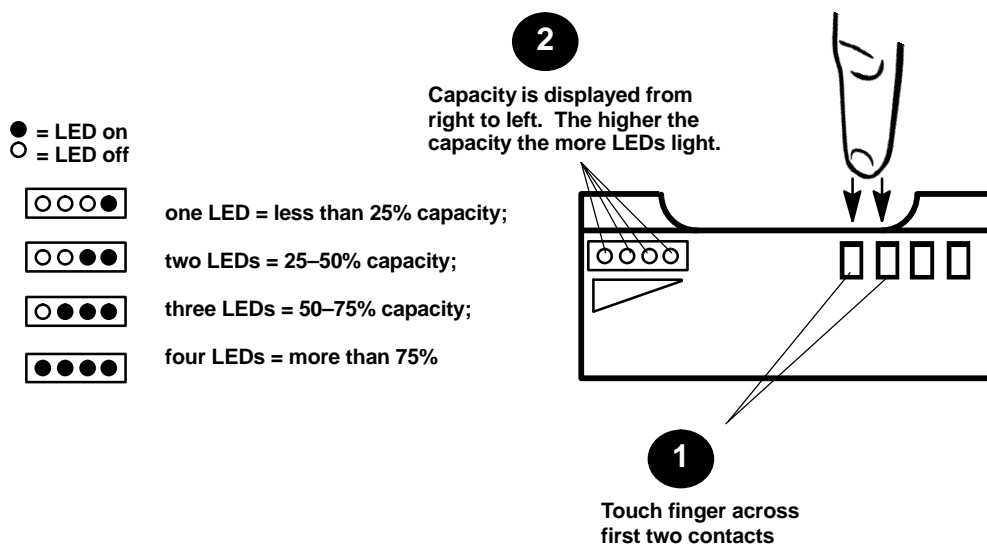


Figure 1-2
Main Battery Pack (bottom view) Showing Contacts

Connectors

There are connectors on both ends of the 6220 Computer:

- ▶ A 15-pin serial port, mounted on the top end.
- ▶ An RJ-11 4-pin phone jack, mounted on the top end.
- ▶ A 12-pin surface contact connector, mounted on the bottom end, and makes contact with a dock.

Display

This terminal features a backlight, with (optional) touch-sensitive LCD display and a resolution of 160 (wide) by 200 (long) pixels. Use a stylus or your finger on the touch screen. Do not use a ball point pen, since it may damage the screen. Application software can only operate the display in portrait mode and receives touch inputs as standard mouse messages.

Backlight

The backlight is useful in darkened conditions. However, use the backlight only as needed, since it reduces the battery life.

The amount of time the backlight stays on before shutting off is controlled through the Configuration Program, SC400CFG.COM. See page 2-2 for more information.

Touch Screen

The (optional) overlay resistive analog touch screen is controlled by a single chip touch screen controller. The touch screen controller provides x,y point information to the CPU whenever a touch is detected on the screen.

Docks and Chargers

The batteries in the 6220 Computer can be charged by placing the computer into a dock or connecting the batteries to a charger. The 6220 Computer fits many of the docks available for other hand-held computers. The following types of docks are available for the 6220 Computer:

- ▶ **Single Dock** Integrated AC power supply, holds one hand-held computer
- ▶ **Multidock** Integrated AC power supply, holds up to eight hand-held computers, secure storage, light status indicator.
- ▶ **Printer** Intermecc AC-powered printers generally provide a way to put a hand-held computer in a dock and charge it while docked.
- ▶ **Wall Charger**

Keyboard

The 6220 Computers have two keyboard configurations, 30-key or 56-key, as shown in Figure 1-1 on page 1-2.

Keyboard Remapping

All keys can be remapped, including shifted and unshifted keys, with the exception of the **ON** key (suspend/resume) and the **Yellow** shift key. Use SC400CFG.COM to remap the keypad, as described on page 2-2.

Suspend/Resume

To suspend operation, press and release the **ON** key. To resume operation, press and release the same key.

The 6220 Computer automatically suspends when there is no activity for a time, which conserves power, thereby protecting against loss of data while replacing the main battery or leaving the unit unattended. The time before automatically suspending is determined by the configuration program.

Memory

The following types of memory are available with this computer:

CMOS

CMOS memory retains configuration information (such as RAM drive parameters across reset cycles).

System Memory (DRAM)

The 6220 Computer supports 8 and 16 MB of DRAM memory which is ordered at the time the product is purchased. The computer can be upgraded to a larger memory size if Customer Support Center installs a new memory board.

Flash ROM

The 6220 Computer has 2 MB of field-programmable flash memory, with BIOS and DOS firmware. Applications stored in flash memory are copied into DRAM for execution. Flash memory does not require a battery to maintain its contents.

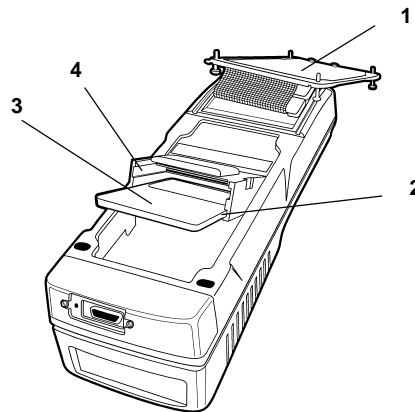
PC Cards

Use either a Type II card or Type III card in this computer. These cards work similar to a hard drive on a PC. Modems, radios, hard drives, and other options will also be supported as they become available. When inserting a PC Card into the drive, use the slot closest to the display.

Different brands of PC cards can be used in the 6220 Computer. Memory cards are available in a variety of sizes and types. Check with your Intermec Account Executive for specific options.

► **NOTE:** *SRAM cards are not supported as data storage devices on this computer.*

Figure 1-3 shows a rear view of the 6220 Computer, with the display facing down. Insert the card into the drive closest to the display. After inserting a boot card into a Type II slot, press the reset button before the card is recognized.



1. PC memory compartment door (removed).
2. Lower PC Card slot for Type II or Type III cards.
3. PC Card.
4. Upper PC Card slot for Type II or Type III cards.

Figure 1-3
Location of Components (rear, top)

► **NOTE:** *Cards that conform to the PC Card interface standard (new standard), or the PCMCIA interface standard (old standard), are referred to as: "PC Card" or "PC Card xxxxx" (where xxxxx consists of "modem," "drive," or other device type). This is because "PC Card" is rapidly becoming the accepted industry term for a storage medium that conforms to these standards.*

Open System Environment

One of the major benefits of an open-system approach is that development equipment and software can be acquired from many different vendors, including Intermec Technologies Corporation. This provides a wide latitude in selecting the equipment and software tools that are best suited to particular needs.

Ports

The 6220 Computer contains a variety of ports, which include:

- Two serial ports (standard COM1 and multiplexed COM2), with connections through the dock, using CMOS logic levels.
- An Infrared port for wireless communication with peripherals.

Power Management

Advanced Power Management (APM) software coordinates power-saving operations between applications and hardware. The APM code detects periods of inactivity and lowers power consumption by progressively reducing the processor clock speed, shutting down unused components, and finally suspending the unit. The APM software is responsible for handling suspend/resume requests via the **ON** key and notifies the system software when suspend/resume events occur.

The processor, the LCD screen, PC Cards, and the tethered scanner are components that are automatically power-managed.

Device drivers and applications can dynamically modify APM behavior through APM function calls, or certain configuration files can be modified to establish a default power management.

Printers

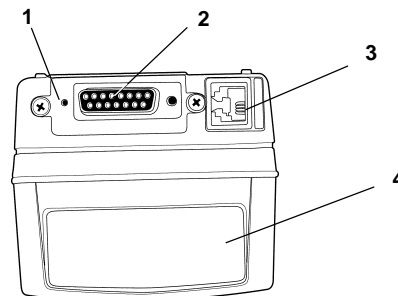
Tethered and IrDA printing are supported. Norand[®] Portable Communications Protocol (NPCP) printing is supported only if the NPCP printing toolkit is used.

Processor

The 6220 Computer is based on the ÉlanSC400 processor operating at a maximum speed of 99 MHz. This processor is modeled after the AMD Am486 CPU and contains an 8 KB write-back cache, integrated memory controller, PC/AT system logic, PC Card controller, LCD graphics controller, IrDA infrared port, and power management features in one integrated package. The integrated LCD controller is CGA register set compatible.

Reset Switch

Occasionally, the 6220 Computer needs to be reset. *Note that unsaved data will be lost when this occurs.* To reset the unit, push a small diameter wire, such as a paper clip, into a hole in one end of the 15-pin serial connector (as in Figure 1-4).



1. Reset switch
2. 15-pin serial connector
3. Phone jack (optional)
4. Display side of computer

Figure 1-4
Location of Components (top end)

Scanners

The tethered scanner connects to the standard 15-pin serial connector located on the top end of the terminal. RS-232 is used for data transfer. Intermec Technologies Corporation provides device drivers for all supported operating systems.

System BIOS

The 6220 BIOS is an implementation of the PC compatible BIOS with added features specific to INTERMEC[®] hand-held computers.

System BIOS performs power-on tests, system configuration, low-level power management, and run-time real-mode interrupt service handling. During the power-on sequence, the BIOS performs a Power On Self Test (POST), which checks all system components and configures them in preparation for booting the operating system. The system then boots the operating system from an flash memory.

System BIOS storage is provided by flash memory which is field programmable. The BIOS may occasionally need updating. See Section 4 for details.

Related Publications

The publications listed below contain additional information relating to the 6220 Computer or publications that may be good references for application development. Order Intermec publications from an Account Executive.

- ▶ **6220 Hand-Held Computer User's Guide** Intermec P/N: 961-028-104
- ▶ **APM BIOS Interface Specification 1.1**
Intel Corporation
Literature Distribution Center
P.O. Box 7641
Mt. Prospect, IL 60056-7641
(800) 548-4725
Publication Order Number: 241704-001
- ▶ **Handwriter Recognition System for Windows User's Guide**
Communication Intelligence Corp. (CIC) Intermec P/N: 961-054-001
- ▶ **Handwriter Recognition System for Windows Release Notes**
Communication Intelligence Corp. (CIC) Intermec P/N: 961-054-002
- ▶ **Handwriting Recognition System, HR-1200, User Manual**
Synaptics Intermec P/N: 961-054-007
- ▶ **Microsoft Developers CD**
A technical reference for developers; published quarterly on CD:
Microsoft Developers Network
One Microsoft Way
Redmond, WA 98052-6399
- ▶ **System BIOS for IBM PC's, Compatibles, and EISA Computers, Second Edition**
Phoenix Technologies LTD.
Addison-Wesley Publishing Company
ISBN: 0-201-57760-7
- ▶ **Writing TCOM Modules in PL/N**
Intermec Technologies Corporation P/N: 541-002-523
 - ▶ **PL/N 4000 Series Manual, Volume 1** Intermec P/N: 977-037-007
 - ▶ **PL/N 4000 Series Manual, Volume 2** Intermec P/N: 977-037-008

Customer Support

Factory Service

If your unit is faulty, you can ship it to the nearest authorized Service Center for factory-quality service.

Customer Support Center

The Intermec Customer Support Center (technical support) telephone number is 800-755-5505 (U.S.A. or Canada) or 425-356-1799. The facsimile number is 425-356-1688. Email is support@intermec.com.

If you email or fax a problem or question include the following information in your message: your name, your company name and address, phone number and email to respond to, and problem description or question (the more specific, the better). If the equipment was purchased through a Value-Added Reseller please include that information.

Web Site

The Customer Support File Libraries, including Hot Tips and Product Awareness Bulletins, are available via the Intermec Product Support page at this URL: <http://norbbs.norand.com/index.htm>. New users can sign up for a new account on this page.

PDF versions of Intermec manuals can be found at this URL: <http://corp.intermec.com/manuals/english.htm>.

Bulletin Board Service

The Customer Support Bulletin Board (BBS), maintained by Intermec Technologies Corporation, provides software and documentation:

- ▶ **Phone number:** 319-369-3515 (14.4 Kbps modem)
319-369-3516 (28.8 Kbps modem)
- ▶ **Protocol:** Full duplex, ANSI or ANSI-BBS; 300 to 28,800 bps; v.32bis; 8 bits, no parity, 1 stop bit. *For high-speed modems, disable XON/XOFF and enable RTS/CTS.*

This is the same location available via the web site. If your web access uses high-speed phone lines, the web interface provides a faster response.

DOS Device Drivers



This section contains installation and configuration information for the following DOS device drivers, including communications device drivers and DOS utilities.

APM Clock Driver

CLOCK.EXE is part of the Advanced Power Management (APM) system when running under DOS. CLOCK is a device driver that replaces the standard DOS clock driver. It must be installed as a device driver at system startup time.

Besides the normal clock driver functions, CLOCK makes sure that the date and time are maintained during suspends. It forces a read from the CMOS real-time clock upon Resume.

CLOCK is installed as a device driver in the CONFIG.SYS file. CLOCK should be installed after NORDOSPM.EXE. A typical entry in CONFIG.SYS to install CLOCK is:

```
DEVICE=NORDOSPM.EXE
DEVICE=CLOCK.EXE
```

After installation, CLOCK displays the following message:

```
CLOCK.EXE Vn.nn DOS APM Clock Driver
```

There are no installation switches or user-programmable parameters for CLOCK.EXE.

BGI Support

N6220.BGI is a custom Borland Graphics Interface (BGI) real-mode PENKEY.BGI driver, located in the DOS Toolkit. It develops Borland graphics-based applications for this display. One use in a DOS application is to separate the application and the BGI driver. This requires loading the driver from your application. The calls are documented in the Borland C++ DOS Reference book. Start with *installuserdriver*. This is also used by CALIB.EXE (Pen Calibration).

Using N6220.BGI Driver

The driver “plugs in” to the Borland BGI graphics library via the *installuserdriver()* function. Mode selection is performed through several custom user drivers provided in the BGI demo CUSTOM.C file.

► **NOTE:**

The N6220.BGI driver can be initialized and closed many times during a program run, but call installuserdriver() once and save the value returned for subsequent driver initializations.

Bitmap Text Output

The Borland BGI drivers do not fill the background color when rendering the bitmap font, thus erasing the text before writing over it. Use the `bar()` function to erase (clear) the text area (text cannot be written over with new text or spaces).

N6220.BGI fills the bitmap text background, eliminating text erasure gyrations. This new text mode is set using the BGI `setwritemode()` function and passing the `BGI_NIFTYTEXT` constant (defined in N6220.H). Pass `BGI_NORMALTEXT` to the `setwritemode()` function to return to normal BGI text rendering.

BGI Demonstration Files

If the `BGIDEMO.C` demo file is copied and modified for personal needs, copy other files included with `#include` lines found in the demo file.

Refer to *Borland C++ DOS Reference* for additional information.

DOS Utilities

Configuration Utility

`SC400CFG.COM` provides configuration for timers, activity monitors, keypad, and features such as: standby, powerfail, enable/disable suspend key, processor slowdown percent, system clock slowdown rate.

Remapping Keyboard

All keys on the keyboard, including the shifted and unshifted keys, can be remapped, except the **ON** key (suspend/resume) and the shift key.

► **NOTE:**

Before reprogramming certain keys, consider the effect it might have on other operations. For example, remapping the shifted [7] and [8] keys on a 30-key keyboard would lose the contrast adjustment.

Configuring Power Management

APM software is implemented to coordinate power-saving operations between applications and the hardware.

Using SC400CFG

Use `SC400CFG.COM` to reconfigure any part of the system, such as the keypad or power management. The example `SC400CFG.INI` file lists all the parameters that can be reconfigured.

First, make a copy of the example `SC400CFG.INI` file (located in the `\Utils` directory on the boot drive). The name of the INI copy can be changed.

The copy should include only those parameters that are to be changed. Delete all others. After modifying the configuration file, add the following line to the `AUTOEXEC.BAT` file on the PC Card, then reboot. Usage is as follows:

```
sc400cfg.com [options]
```

If your configuration file has a different name, specify that name in the command line, as described later in the *Command Line Switches* paragraph. This paragraph also contains other details concerning the [options]. If an INI file is *not* specified, `SC400CFG.INI` (if available) is used as the default.

SC400CFG.COM <options>

where **<options>** include the following:

-? Displays a help list similar to that shown here.

-F <filename>

Specifies file with setting changes. Default = SC400CFG.INI

-S<section>

Specifies section in file to use.

EXAMPLE:

```
SC400CFG -F:MyConfig.ini -S:NewConfigs
```

where *MyConfig.ini* is the name of your configuration file and the *[NewConfigs]* section of the configuration file contains the applicable settings.

Omitting the **-F** parameter defaults to the default configuration filename, SC400CFG.INI.

Omitting the **-S** parameter tells SC400CFG.COM to use the *[Default]* section of the configuration file.

Create Download Include File

Given a list of files, IPLFMT.EXE creates a download include file by concatenating a list of initial program files to be loaded onto a hand-held computer, prefixing each file with an appropriate DOSFIL header. See page 5-14 for header descriptions. Use the following format to create a network download file.

```
iplfmt.exe <list file> <include file>
```

list file

The name of the file that contains a list of files to include.

include file

The name of the download include file, such as DOWNLOAD.INC (this file is created by IPLFMT.EXE)

IPLFMT.EXE also extracts files from a network download file, using this format:

```
iplfmt.exe -r <include file> [<path>]
```

include file

The name of the download include file, such as DOWNLOAD.INC (this file is read by IPLFMT.EXE).

path The name of the output directory where the files are to be extracted; this parameter is optional.

A download list file contains the names of the files to be sent to your hand-held computer. Each line in the file specifies one name. The file names may include path information so that all of the files need not be in the same directory on the host. However, the file is transmitted without the source path information, and placed in the current directory of your hand-held computer.

To specify a destination path or to rename a file on your hand-held computer, use a line with the "newname=oldname" format. This causes the file "oldname" to be read from the PC and transmitted to your hand-held computer with the name "newname". Both "oldname" and "newname" may include path information.

EXAMPLE:

The following is a sample download list file:

```
app.lzh
config.sys
```

Drive Mapping Utility

MAPDRIVE.EXE is similar to SUBST.COM, in that drive letters can be assigned to other drives. MAPDRIVE:

- ▶ Maps two drives together so that the system allows access to only one device regardless which of the two drive letters was requested.
- ▶ Handles errors automatically within an application by invalidating any drives that would normally result in an interrupt vector requiring interaction at the DOS level.

To assign a path to a drive letter, use the `mapdrive.exe d=[path]` syntax.

EXAMPLE:

This sample assigns path (E:\temp) to drive A: `mapdrive.exe A=E:\temp`

To delete (or invalidate) an assignment for a drive letter, remove the assignment, using this syntax. "D" represents the drive letter assignment to be removed: `mapdrive.exe d=`

A drive can be reassigned to its original assignment, using an assignment such as:
`mapdrive.exe A=A:\`

Multiple drive mappings can be performed with a single command line. This sample deletes drive assignments for A: and B: `mapdrive.exe A= B=`

File Integrity Verification Utility

The DOS Toolkit contains a utility, CRC32.EXE, which computes a 32-bit Cyclic Redundancy Check (CRC-32) on a set of files. This check is computed using a complex mathematical equation that results in a unique CRC-32 value. This value guarantees accuracy of the file transfer so that even a minor change to a file results in a dramatically different CRC. This CRC value can validate whether system files are corrupt.

To verify file integrity, run CRC32.EXE to calculate the 32-bit CRC value. Then compare that value with the factory CRC value. All factory CRCs are listed in RELNOTES.TXT file in the Toolkit. Usage is as follows:

```
CRC32 [@][filename | pathname] [/s]
```

filename

The name of the file on which the CRC is calculated.

pathname

The location of the filename to be processed.

/s Search all subdirectories for matching filenames.

@ Optionally included to indicate that *filename* is an argument file, containing names of files to be checked. A valid argument file for CRC32.EXE has the same format as an argument file for IPLFMT.EXE.

▶ NOTE:

Wildcard processing in the filename is allowed only on the command line, but not within the argument file. CRC32.EXE processes directories with any attribute set, such as 'read-only'.

One or more files or directories can be processed at one time. For command line help, use the `/?` switch.

Get or Set ROM DOS Boot Flags

MMBFLAG.COM satisfies the following batch file requirements:

- ▶ The error level returned by MMBFLAG.COM indicates the current value of the DOS boot flags. The boot flags keep track of the default boot drive and serve as an indicator of Master Mode Booting.
- ▶ Change the default boot drive using a command parameter (*Note: Drives E and H are also supported*):

```
MMBFLAG.COM 2   (Drive C:)
MMBFLAG.COM 3   (Drive D:)
MMBFLAG.COM 4   (Drive E:)
```

Table 2-1
MMBFLAG.COM Error Levels

Error Level	Meaning
132	Master Mode Boot from Drive B (default boot drive is E)
131	Master Mode Boot from Drive B (default boot drive is D)
130	Master Mode Boot from Drive B (default boot drive is C)
4	Drive E is the default boot drive
3	Drive D is the default boot drive
2	Drive C is the default boot drive

Multipurpose Delay Utility

DELAY.EXE serves the following functions:

- ▶ Displays a message to the screen (like the Echo command)
- ▶ Waits for specified amount of time before continuing (similar to the Sleep command or DOS 6 Choice command with a time-out specified)
- ▶ Pauses system execution until key is pressed (like the Pause command)
- ▶ Returns an error level based on which key was pressed (like the DOS 6 Choice command)

The first three functions perform even when DELAY.EXE is loaded as a device driver. Since the error level concept does not apply to CONFIG.SYS processing, the fourth function applies to command line (batch file) execution.

1. To display one or more different messages, supply the new messages within quotes on the command line. Each quoted text string displays on a separate line. Use a null message (“”) to display a blank line. To keep the cursor positioned immediately after the last character displayed, do not supply the trailing quote (”). To display a message without waiting for either a timeout or a keypress, specify a delay timeout of zero (/0).
2. To force DELAY.EXE to exit after a period of time, even with no key press, specify the “/nnnn” switch anywhere on the command line. The decimal value of “nnnn” is the hundredths of seconds to elapse before DELAY.EXE exits. The maximum delay is about 640 seconds or ten minutes.
3. The /K parameter locks the keyboard during execution of DELAY.EXE.
4. When DELAY.EXE is executed from a command shell, the error level set (upon return) is based on the key, if any, that was pressed prior to exiting. The return value is geared to determine which numeric key was pressed.

Syntax: [device=]Delay[.exe] [/nnn [/K]] {"Message(s)"}

device=

Loads DELAY.EXE as a device driver. The full pathname for the executable must be specified, as follows: `device=c:\utils\delay.exe`

When executing DELAY.EXE on the command line, omit "device=" and only include the full pathname for the executable if that path is not in the PATH environment variable.

/nnn Forces DELAY.EXE to timeout, by specifying a timeout period. The timeout period (nnn) is specified in hundredths of a second. Thus, /100 causes DELAY.EXE to timeout after one second. Note if DELAY.EXE is from a DOS box under Windows, the timeout duration is longer due to how Windows implements the timeout function.

/K Disables the keyboard while pausing. Specify this switch *after* providing a timeout value or the delay cannot be terminated. Without this switch, the delay terminates with a key press, even with a specified timeout.

{"Messages(s)"

One or more messages can be specified on the command line. Each message starts with an undisplayed quote ("). If a trailing quote (") is not supplied, then the cursor remains positioned immediately following the message. Otherwise a carriage return and line feed is issued at the end of the message. The default message is "Press any key to continue..."

Errorlevel return values:

When DELAY.EXE is run as a command, use the errorlevel set (the delay is complete) to determine which key was pressed or whether a timeout occurred. The return value states which numeric key was pressed. Below are error levels and their common key pressed values.

- ▶ 0 through 9 "0" through "9"
- ▶ 221 The YES key
- ▶ 235 The NO key
- ▶ 255 No key was pressed, the time already expired
- ▶ 12 to 154 All other values are determined by subtracting 30h from the key value returned by reading the keyboard directly.

EXAMPLE:

Standard PC Function

Pause
Wait for "1" to be pressed

Echo "message"
Sleep 100 ms
Echo "Pause for 1 second"
Sleep 1 second (with keyboard disabled)
Choice /c:12 "message"

Echo "message two"
Pause

6220 Commands Required to Perform Function

```
DELAY.EXE
:Loop
    DELAY.EXE "Press '1' to continue . . ."
    If not errorlevel 1 goto Loop
    If errorlevel 2 goto Loop
DELAY.EXE /0 "message"
DELAY.EXE /10 ""
    DELAY.EXE /100 /K "Pause for 1 second"
DELAY.EXE "message"
    If errorlevel 3 goto InvalidDigit
    If errorlevel 2 goto PressedTwo
    If errorlevel 1 goto PressedOne
DELAY.EXE "message two" "Press any key to continue"
```

Reset the System

RESET.EXE resets the system for batch files or other software.

RAM Drive

RAMDFMT.EXE creates a RAM Drive on the HHC. This RAM drive is a block of system memory that is treated as DOS drive C:. The data stored on the RAM drive is maintained as long as power is supplied to the system and the drive is not reformatted.

When a RAM drive is created, this results in less extended memory available for programs. No additional conventional memory is required for the RAM drive. Because the support for accessing the RAM drive is embedded in the DOS BIOS, no additional drivers need to be loaded. The extended memory test (with the corresponding boot messages) reserves memory for the RAM drive. Only one RAM drive can be created.

The size selected for the RAM drive represents the total memory required by the RAM drive, not the free space that is available on the drive. DOS file system overhead reduces the specified memory by the size of the file allocation table (FAT) and root directory created.

If CONFIG.SYS loads HIMEM.SYS and sets DOS=HIGH, the first 64 KB of extended memory is used for the High Memory Area. As a result, the maximum size of the RAM drive is 64 KB less than total extended memory.

The following switches are allowed for RAMDFMT.EXE. All values are in decimal. Specify either size or address, but not both.

- a **address** of RAM disk (in kilobytes)
- d **drive number** (0 = A, 1 = B, ... etc.)
- e **number of directory entries**
- s **size of RAM drive** (in kilobytes)
- h **help**

► **NOTE:**

Reboot the HHC after the RAM drive is created (or altered), to make the change effective.

If a communications option other than Accessory Card is selected, the Norand Utilities program automatically creates a RAM drive if one does not already exist. If the accessory card option is selected, the startup files on the card must create a RAM Drive if one is needed.

For a system with 6 MB (or more) of total memory, Norand Utilities creates a RAM drive of size 3 MB plus 960 KB, that is, 64KB less than 4MB to ensure room for the High Memory Area. For lower system RAM capacities, only 1 MB plus 64 KB of RAM is reserved for conventional use and the remainder is allocated to the RAM drive.

The Norand Utilities program provides a menu that allows the RAM drive to be formatted in 64 KB increments. The RAM drive created by Norand Utilities allocates 128 directory entries, that is, it specifies -e128 as a parameter to RAMDFMT.EXE.

When NPCP, TTY, NRInet, TFTP, TCP/IP bootp, or Novell communication options are used, another choice exists for creating a RAM drive. If the RAMDFMT.CTL file is downloaded, Norand Utilities executes the program RAMDFMT.EXE to create a RAM drive. RAMDFMT.CTL is a text file that must contain the command-line parameters for RAMDFMT.EXE.

For example, if RAMDFMT.CTL contains the -s2048 line, then Norand Utilities executes "RAMDFMT.EXE -s2048", which creates a 2 MB RAM drive.

When processing RAMDFMT.CTL, if RAMDFMT.EXE cannot be executed or returns an error, the communication session fails with error "F 0." Also, if the parameters in RAMDFMT.CTL cause the RAM drive to be removed (such as "-s0"), the communication session fails with error "F 0."

IrDA Printer

The IrDA Lite device driver, NORIRDA.SYS, provides primary DOS character driver support for IrDA printers. Ensure NORIRDA.SYS is in the `\Drivers` directory on the system boot drive. These entries are required in CONFIG.SYS:

```
NORIRDA.SYS [-n:driverName] [-d:n] [-r:n]
```

- d:n** Specifies the disconnect time (in seconds) requested during negotiation, where “n” is 3, 8, 12, 16, 20, 25, 30, or 40. The parameters for this switch are cumulative and can add any or all of these disconnect times to the “3” default. The two systems negotiating select the largest common value.
- n:driverName**
(*Optional*) Specifies the new name to which this character device driver should respond. The default is IRDAPDRVR.
- r:n** Specifies the number of seconds discovery retries that are done at the first DOS write command by the driver. The default is five seconds.

LAN Communications

MININET.EXE provides communications over Local Area Networks (LANs) and automatically switches between supported speeds (500 KB, 460 KB, 115.2 KB) to locate a working speed. For details, see page NO TAG.

The default baud rate is 500 KB, but automatically switches to the secondary speed if necessary. If this switch is not used, the default baud rate is used, otherwise the speed can be set to any of the following values:

- ▶ 5000 500 KB (*default*)
- ▶ 4600 460 KB
- ▶ 1152 115.2 KB (*secondary default*)

To load MININET, use this format to add a command line to AUTOEXEC.BAT:

```
MININET.EXE [-s#] [-c#] [-t#]
```

All parameters are optional, as designated by the brackets []. Replace “#” with a value appropriate for each parameter, as described below:

-c# Sets the default COM port to be used.

▶ NOTE:

Supplying the -c# switch prohibits an application from selecting a COM port via the NCB CALL establishing a session. This makes it possible to force which COM port is to be used.

-s# Sets the primary speed of communication.

-t# Sets the secondary baud rate to automatically toggle between the primary and secondary baud rates, depending on the speed of the host. A value of zero indicates that only the default baud rate is used (set with **-s** switch). Any of the baud rate values listed previously are valid for this switch.

If neither **-s#** or **-t#** are set, MININET.EXE autoselects a baud rate from the three valid speeds.

If only **-s#** is set, MININET toggles the baud rate selection between the requested speed and 115.2 KB.

If both **-s#** and **-t#** are set to valid speeds, MININET toggles the baud rate selection between those two values.

If **-t#** is set to zero, MININET sets the baud rate to the value set by **-s#** or to the default value of 500 KB.

NPCP Printer

The DOS device driver, PC4800.SYS, provides Norand[®] Portable Communications Protocol (NPCP) printing support for DOS and PL/N applications written for the hand-held computer. This includes printing to Norand 4800 and 6800 Series Printers using NPCP.

Installation and Configuration

Required CONFIG.SYS Entry

PC4800.SYS is installed as a device driver in the CONFIG.SYS file.

EXAMPLE: `device=pc4800.sys [device name [port number [/I|lpt]]]`

device name

The name to use for this instance of the device driver. This name opens the device. It can be anything except PRN, and can be up to eight characters long. The default name is NP4800.

port number

The COM port used for output by the driver. Valid values: 1=COM1, 2=COM2, etc. Default: COM1

/I Redirects LPT output to the BIOS interrupt 17h API, and is required to support PL/N applications.

lpt The LPT port to be redirected. Valid values: 1=LPT1, 2=LPT2, etc. By default only output directed to the specified device name is captured.

► **NOTE:** *The order of the command line parameters is important. For example, you must specify a device name if a port number is specified.*

EXAMPLE 1: Redirect LPT2 to COM1: However, do not redirect any LPT2 output at the interrupt 17h (BIOS) interface.

```
device=pc4800.sys LPT2 1
install=4000api.exe ... { refer to 4000API.EXE for switches }
```

EXAMPLE 2: Redirect all LPT1 output to COM2.

```
device=pc4800.sys LPT1 2 /I
install=4000api.exe ... {refer to 4000API.EXE for switches}
```

If PC4800.SYS is not in the root directory, be sure to include the path. Be consistent; if NORIRDA.SYS is located in the \Drivers directory, so should PC4800.SYS.

► **NOTE:** *The /I switch is for PL/N application support. Do not use this switch for non-PL/N applications. The interface provided is not 100 percent PC-compatible.*

Required AUTOEXEC.BAT or CONFIG.SYS Entry

For additional information describing how to load PC4800.SYS, see page 3-2. Normally, 4000API.EXE is loaded in CONFIG.SYS using:

```
install=4000API ... {refer to 4000API.EXE for switches}
```

However, if it needs to be loaded high, load it in AUTOEXEC.BAT.

Functionality and Usage

Intermec Technologies Corporation supplies PL/N and C standard printer routines that handle critical errors and the printing of text to Norand printers. The following information is provided to create custom print routines.

Open PC4800.SYS before using it. For example, in the C programming language, use `fopen()`. Use the correct open call for the language used, to write to the device. Also, the open call used should return an error if the device does not exist. Some open calls create a file if the device driver is absent.

To write to the device, use any appropriate output function that can be directed to the device. For example, in C programming, use `fprintf()`.

Install a critical error handler that processes errors from the printer. The printer driver returns only “device not ready” errors to DOS for compatibility with the standard PRINT program supplied with DOS. The extended error can be retrieved by bypassing DOS and calling the driver directly from the error handler.

Ioctl calls (DOS Function AH = 44h, AL = 02h) are supported by the device driver (see a DOS reference manual). Calls require the printer device handle number to be loaded into register BX, a data item with the following structure pointed to by DS:DX, and the number of bytes to be read in register CX.

```
struct ioargs {
    far *unsigned char ioctl_cmd;
    far *unsigned char ioctl_buf;
};
```

The following commands may be pointed to by the `ioctl_command` field for execution by the driver:

```
ONLINE = 2
STATUS = 3
GET DEVICE FEATURES = 4
SELFTEST = 5
RESET = 6
GET CURRENT CONFIGURATION = 7
XTENDED ERROR = 255
```

When the ioctl call is made, data pointed to by the `ioctl_buf` is returned to the buffer for the number of bytes specified in register CX. The first byte of the returned data represents any error encountered during the call. If the call was successful, this byte is set to zero. If unsuccessful, this byte represents the extended error. Refer to printer documentation for information detailing the format for the rest of the data.

- **NOTE:** *Open and close the printer driver for each report transaction with the printer to take advantage of the NPCP line-loss detection features. Applications should status (ioctl 3) the printer after the last line of a report or a series of reports are sent to the printer to ensure that all lines were printed successfully.*
- **NOTE:** *The printer driver is multi-tasking. If the application needs to take full advantage of this feature, it must enable time slicing through the appropriate call to INT 15h (refer to the 4000API documentation).*
- **NOTE:** *If time-slicing is not activated, the last line buffered by the driver is not sent to the printer until the device is closed or an ioctl status call to the printer is made. Perform an ioctl status call before all closes to ensure the driver print buffer is sent to the printer successfully, since DOS closes to character devices do not perform output flushes as part of the close operation. The driver flushes the buffer upon a close, but no error is returned if the flush is unsuccessful.*

PC Card Services

NORMOD.SYS is a PC Card services client that handles identification and initialization of serial device PC Cards (such as modems and radios) and is installed in the CONFIG.SYS file, as follows:

```
DEVICE=NORMOD.SYS [-A] [-Bd] [-Cx] [-Dx x x] [-Ix] [-Nname] [-R] [-S] [-V]
```

-A Allows suspend requests, even if an open session is present. Normally, NORMOD rejects suspend requests when an application has an open connection.

-Bd Defines a battery cutoff level in millivolts. If a voltage reading for your main battery is below this level, then *open* requests fails. This protects against loss of data when power fails because of low batteries.

EXAMPLE: -B610 sets the cut off at 610 millivolts.

-Cx Overrides the default COM port. The default is COM3.

EXAMPLE: -C4 defines using COM4 instead of COM3.

-D x x x

Defines the timeout used for a particular manufacture's code. If a modem card has an unreliable ready line, this switch could define a timeout value instead of the application examining the ready line. The first two hexadecimal numbers must match both the code and the information values returned by a *GetConfigInfo* call before the third value is used as the timeout value. The timeout value needs to be specified in tics (tic=1/18th of a second).

EXAMPLE: "/D 16b 21 20" sets timeout to 20h tics, about 1.8 seconds. The "16b 21" identifies the Ericsson PIA radio.

-Ix Changes the interrupt request level to be used. The *x* parameter should be supplied in hexadecimal. The default value for *x* is *IRQ 5*.

EXAMPLE: -I7 uses *IRQ 7* instead of *IRQ 5*.

-Nname

Changes the installed device name. The default *name* value is *MODEM*.

EXAMPLE: -Ncom4 sets the device name to *COM4* instead of *MODEM*. *Nname* can be any valid 8-character name. However, do *not* use a name that is the same as the root name of any other current or future existing file or directory. Once a device name is defined, any files or directories with the same root name cannot be accessed.

-R Allows a card to be left powered on, with ring-indicate (RI) enabled during suspend. RI awakens terminal.

-S Saves UART registers when a suspend occurs, and restores upon resume.

-V Turns on 12-volt VPP whenever a modem session is open.

Pen Calibration

CALIB.EXE is the DOS Pen Calibration utility. It performs calibration of the pen interface for the hand-held computer.

Calibration aligns the location of the cursor to the same location where a finger or pen is placed on the touch screen. The hand-held computer comes with default settings normally acceptable for most applications. However, some applications, such as signature capturing, may require a calibration adjustment to compensate for variations in touch panel alignment and user preferences.

Configuration

Configuration settings are stored in CMOS. If CMOS contains invalid information, the program displays the calibration screen, regardless of the existence of the calibrate command-line option.

CALIB.EXE is a DOS application program and is invoked from AUTOEXEC.BAT or from the DOS command line. The appropriate mouse driver must be installed. The driver contains extensions to the standard Mouse API which provides digitizer information, screen resolution, and raw coordinate points, required for implementing CALIB.EXE.

Required Calibration Files

MOUSE.COM (pen driver)
CALIB.EXE (calibration utility)
N6220.BGI (BGI driver)

- ▶ **NOTE:** *Always load the pen driver before the pen calibration utility, as shown above.*
- ▶ **NOTE:** *For information about using the BGI driver, see page 2-1.*

Operation

CALIB.EXE begins calibration by placing one of four targets at a fixed location on the screen and waiting for three pen presses in the center of the target. Apply moderate pressure while pressing the target. Light taps, too much pressure, fingers, or other objects touching the screen generate incorrect results. The unit beeps after each successful press. If a beep is not heard, lift and press again. When three good data points are acquired, the next target is displayed.

When data for all four targets is acquired, the program displays a screen showing the calibration results. Check the match of the cursor with the location of the pen near the four corners of the screen. Do this by placing the pen down near each corner of the display and see if the cursor appears at the pen location. If the cursor is offset from the pen location by some fixed amount, click the appropriate **UP**, **DOWN**, **LEFT**, or **RIGHT** adjust buttons on the screen. You may prefer to add some amount of offset to place the cursor away from the pen using the same buttons. If the cursor position varies significantly from the pen location, the four-target calibration process was not successful. Restart the program and press the pen against the targets again.

When satisfied with the calibration, click **SAVE** on the screen to store the results. The results are then available for use by the pen/mouse driver. You may choose to not save the results by clicking **CANCEL**. In either case, the program exits. If the calibration program is unable to recognize the pen, press **Escape** to quit the program. In this case, the new calibration values are not stored.

Pen Driver

MOUSE.COM is actually a DOS mouse driver. It interprets the standard INT 33h mouse interface. Not all mouse functions are supported.

Refer to any reference manual that includes standard mouse interrupts or to the *Application Programming Interfaces Technical Reference* available on the internet for a list of available functions:

<http://partners.intermec.com/regions/devspt/index.html>

The pen driver installs as a TSR as a result of placing the following statement in the AUTOEXEC.BAT file:

```
MOUSE.COM
```

If the drive and directory for this driver is not in your path, include that information, as follows:

```
d:\drivers\MOUSE.COM
```

d: The driver and *\path* is the directory path to the driver.

After installation, the driver displays the following message:

```
Driver installed
Digitizer enabled
```

PenPrint

The PENPRINT.EXE program provides the capability of printing to Intermec printers with appropriate error handling, and is shipped with the DOS Toolkit.

Installation and Configuration

Command line usage:

```
penprint [options] <print-file> <print-device>
```

print-file

The name of the file that contains the data to be sent to the printer.

PenPrint does *not* modify this file. The data is sent “as is” to *print-device*.

print-device

The name of the DOS device driver that provides the interface to the printer (such as LPT1).

options:

- a** Cancels the printing of the report when an error message is displayed. If this parameter is not specified, *PenPrint* cancels printing only if a fatal error occurs.
- h** Displays a help screen.
- m** Suppresses the “Fix top-of-page” message.
- p** Disables the use of a pointing device. If this option is not specified, *PenPrint* attempts to auto-detect the presence of a touch pad. In some situations it cannot be reliably determined whether such a device is present. In these situations, this option can be used to make sure the proper prompts are used for error messages.

- r** Provides the option to restart the report, after a fatal error has occurred. If selected, *PenPrint* does not restart the report itself, but instead returns to the calling application with a return code indicating the choice made. It is then up to the calling application to restart the report. This is because a report may consist of a header and trailer, and the calling application may need to make two separate calls to *PenPrint* to print them, in order to prompt such as "TICKET OKAY?" before printing the trailer. If this parameter is not specified, the only option available after a fatal error occurs is to cancel the report.
- v<f>** The name of a file that contains the VOID message. This parameter overrides the default VOID message that is sent to the printer when either a fatal error occurs or the report is canceled. *PenPrint* sends the data in <f> to the printer in the same manner that it handles *print-file*, except that error processing is disabled.

If <f> is not specified, nothing is sent to the printer after a fatal error. This should only be done if the printer itself prints a VOID message when it detects a fatal error. Otherwise, it is always best to provide an indication that the report is invalid.

Return Codes

- 0** Report OK; report was *not* canceled.
- 2** Report was canceled by the user after a non-fatal printer error.
- 3** Report was canceled after a fatal printer error.
- 5** After a fatal printer error, user selected to restart report.
- 128** An error occurred on a device other than the printer.
- 253** The print file could not be opened.
- 254** The display could not be initialized.
- 255** Invalid parameter or incorrect number of parameters.

Usage Notes

1. *PenPrint* depends upon the BGI graphics drivers to implement graphics. Therefore, the appropriate BGI driver must reside in the current directory.
2. Recoverable errors require manual intervention. *PenPrint* will:
 - a. Display the error message.
 - b. Display the response options: "Continue" or (optionally) "Cancel" (*see -a parameter*).
 - c. Require an option to be selected.
 - d. Restore the display.

With "Continue," *PenPrint* continues printing. With "Cancel," *PenPrint* attempts to print a VOID message. If VOID message fails, *PenPrint* prompts for "Fix-top-of-page," then exits with a return code. The calling application prompts you to change the printer type or to restart the report (either choice is optional).

3. When handling fatal (unrecoverable) errors, PenPrint will:
 - a. Display the error message and response options: “Cancel” and (*optional*) “Restart Report” (*see “-r” parameter*).
 - b. Prompt for an option and restore the display.
 - c. Attempt to print the VOID message. If the VOID message fails, prompt for “Fix top-of-page.”
 - d. Exit with the return code.

With “Restart Report,” the calling application should restart the report. With “Cancel,” the calling application prompts you to change the printer type or restart the report (either choice is optional).

4. If the “-v” parameter is not specified, the following VOID message is printed by PenPrint (bold double-wide mode): **VOID * VOID * VOID**
 This message requires at least a 38-column printer. For printers that have more than 38 columns, the default message is printed at the left margin.
5. When printing the VOID message, error handling is disabled (i.e., you see only the error message for the original error). You might be in an infinite loop of error messages in PenPrint, if the original error was from a nonfunctional printer. PenPrint does not filter the data in the *print-file*. Data is sent “as is” to the *print-device*.
6. If a pointing device is detected and the “-p” switch is not used, PenPrint displays check boxes as “Continue,” “Restart Report,” or “Cancel” onscreen. Any touch within a box activates that response.
 If the “-p” parameter is specified or no pointing device is detected, the prompts display “[YES] Continue,” “[YES] Restart Report,” or “[NO] Cancel.”

Power Management

NORDOSPM.EXE, which is part of Advanced Power Management (APM) when running under DOS, is the DOS power management driver interface between applications and the APM BIOS. Install the power management driver as a device driver at system startup time.

This driver attempts to determine when the DOS environment is idle and saves power by idling the CPU and polls for APM events and broadcasts them to DOS drivers and applications. Mediates events that can be failed such as suspend, user suspend, standby, and user standby requests.

Figure 2-1 is a simplified diagram of Power Management Software:

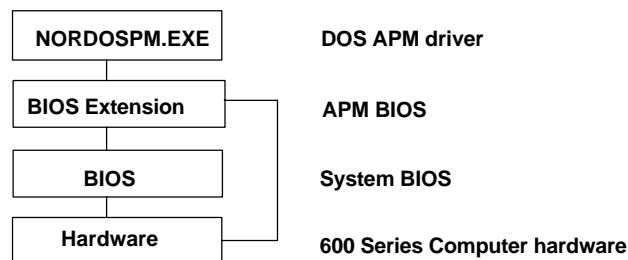


Figure 2-1
Power Management Software

Screen Emulation

FONTMAP.EXE is a 4000 Series screen emulation TSR for the 6220 Computer. This program ports applications written for a 4000 Series platform to the 6220 Computer without modifying the display logic. Screen features include:

- ▶ Character mapping
- ▶ Cursor fixed mode
- ▶ Variable font sizes (20, 21, or 16 characters across the screen)

FONTMAP replaces the standard VGA BIOS interface with a 4000 Series compatible video BIOS interface. Standard VGA BIOS calls are not available once FONTMAP is loaded. It works only with the 200x160-pixel display on the 6220 Computer. In the absence of command line switches, the default mode for FONTMAP is as follows:

- ▶ Cursor chase video mode
- ▶ 8x16 bitmap font
- ▶ Inverse mode on (characters above 127 are mapped to inverse video characters 0–127 when the high bit is turned off)
- ▶ Cursor turned off

Command Line Switches

```
INSTALL=FONTMAP.EXE -F <fontfile> -S -I
```

The following is a list of command line switches for FONTMAP:

- I** Character codes (128-255) displayed as extended characters. If this switch is not present, 128 is subtracted to map character codes (128-255) to (0-127); the remapped characters appear in inverse video.
- C** Sets the number of physical character columns in the display.
- R** Sets the number of physical character rows in the display.
- S** Defaults the screen to fixed mode.
- F <fontfile>**
Loads the font from a file represented by *<fontfile>*, such as GR12x16.FMT. The space between **-F** and the file name is required.

Font File Format

*.FMT files are font files used with FONTMAP.EXE. The following shows the format of a font file:

xsize	byte	Number of font pixels in x direction
ysize	byte	Number of font pixels in y direction
font[(xsize+7/8),ysize]	byte	The actual bitmap font

The largest bitmapped font that can be supported is 8 KB.

Interrupts Supported by FONTMAP.EXE

The 4000 Series interrupts supported by FONTMAP.EXE are listed in Table 2-2.

Table 2-2
FONTMAP Interrupts Supported

INT #	FONTMAP Interrupts	Register Values
10h	Display Services	
	Set Display Mode	AH=00h
	Set Cursor Type	AH=01h
	Set Cursor Position	AH=02h
	Read Cursor Position and Type	AH=03h
	Set Active Display Page	AH=05h
	Scroll Active Display Page Up	AH=06h
	Scroll Active Display Page Down	AH=07h
	Read Character and Attribute at Cursor Position	AH=08h
	Write Character and Attribute at Cursor Position	AH=09h
	Write Character (only) at Cursor Position	AH=0Ah
	Teletype Character Write	AH=0Eh
	Return Current Video State	AH=0Fh
	Set Physical Display Size	AH=12h BL=FAh
	Return Pointer to Current Display Parameters Packet and Font Information	AH=12h BL=FBh
	Return Invert Mode	AX=12DDh BL=FCh
	Enable Invert Mode	AX=12DEh BL=FCh
	Disable Invert Mode	AX=12DFh BL=FCh
	Set Chase Mode	AX=12E0h BL=FCh
	Set Fixed Mode	AX=12E1h BL=FCh
	Return Physical Display Size	AH=12h BL=FDh
	Read or Write Contrast	AH=12h BL=FEh
	Activate, Toggle, or Turn Backlight Off	AH=12h BL=FFh
	Load User Font	AX=1400h
	Load System Default Font	AX=1401h

Unsupported 4000 Series Video Interrupts

The following functions remain unimplemented within FONTMAP.EXE:

- ▶ Function AX = 12E2h BL = FCh: Select Setparms Mode
- ▶ Function AX = 12E3h BL = FCh: Set Graphics Mode
- ▶ Function AH = 12h BL = FDh: Return Physical Display Size
- ▶ Function AH = FFh: Refresh the Screen

Tethered Scanner

H2THRSCN is the tethered scanner driver. It collects and passes scanned data to an application. It also manages power for the scan operations and has the following features:

- ▶ The data is transferred through the serial connector.
- ▶ A key cannot be reprogrammed as a trigger.
- ▶ Nothing is displayed on the screen.
- ▶ If a fully decoded RS-232 scanner is needed, it provides that capability.
- ▶ It handles SE1223 hardware.
- ▶ Baud rates up to 19200 are available.

Installation and Configuration

The tethered scanner is generally installed from AUTOEXEC.BAT by the H2THRSCN statement. Make sure the NORDOSPM.EXE and H2THRSCN.EXE files are located in the root directory on the computer.

Load the NORDOSPM.EXE driver first, then H2THRSCN.EXE. Both drivers are located in the *\Drivers* on the flash. The following entry is required in the CONFIG.SYS file for DOS scanning:

```
device=nordospm.exe
```

Usage

After a resume, press the scanner button again to activate the scanner.

EXAMPLE:

This sample scanner testware enables the scanner, displays the DOS keystrokes until the return or newline character is encountered, then exits.

```
#include <dos.h>
#include <stdio.h>
#include <conio.h>
void main (void) {
    int c;

    _AH = 0x80;          // Collect only one scan, 4500 compatible
                        //
    geninterrupt(0x7a); // Thread thru scan tsr
                        //
    while ((c != '\n') && (c != '\r')) {
        while (kbhit()){
            c = getch();
            putchar(c);
            if (c == '\n' || c == '\r') break;
        }
    }
}
```


Command Line Switches

When using the get help switch (*H2ThrScn/?*), the following is displayed:

```
Norand Tethered Scanner Driver vx.xx
(C) 1995-1999 Intermec Technologies Corporation
All rights reserved.
```

The tethered scanner uses command line switches to drive the desired functions. The syntax is as follows:

```
H2THRSCN [<switches>]
```

where <switches> include combinations of the following:

- H** Displays this information
- A[n]** Enables the aiming beam for *n* 1/2 second-intervals. If the wand or hand-held scanner supports the aiming beam, use this switch to enable it. The message at load time is "Aiming beam set to Xms." where *x* is *n* (interval) times 500 ms.
- E** Causes the tethered bar code reader to always be enabled for used. The message is "Scanning permanently enabled."

► **NOTE:**

-SCANBIOS requires the program to use the API calls to enable and disable the scanner. The message at load time will be "SCANBIOS API enabled."

-SCANBIOS

Controlled using the SCANBIOS API

- U** The scanner driver may be unloaded if the TSR finds it can safely do so, thus freeing memory for other uses. If the unload is successful, a "disabled" message is displayed.
- R** Appends a carriage return (CR) to the end of the data string. Some devices supply this in the form of a postamble string. When using this switch, the driver message is "Enter appending enabled." If the hardware is in working order, the "Scanner initialized!" message appears; if not, then "Scanner HW initialization failed!" appears.

EXAMPLE:

The following is an example command line for the tethered scanner:

```
H2THRSCN [-E] [-R]
```

If the driver is successfully loaded, then "Existing driver reinitialized" appears and the command line options are updated. This way, driver operations can be changed from always enabled to using the SCANBIOS APIs.

If the driver is reloaded and is a different version, the command line switches are not updated. The "existing driver is an incorrect version" will appear.

Tethered bar code reader power is supplied through the ring indicate (RI) signal when the device is enabled. Scanner UART base of 0x3E8 and IRQ 4.

Upper Memory Provider

H2UMP.SYS is an *upper memory provider*. Since it is based on the Elan H2 processor, it supports the minimal set of XMS function calls required to provide upper memory to MS-DOS when you have a “DOS=UMB” statement in the CONFIG.SYS file. It does not support the XMS calls that deal with either the deallocation or reallocation of upper memory blocks.

H2UMP supplies up to 128 KB of upper memory block elements, in the range of C000–DFFF. Option ROM blocks are automatically scanned for and reserved (that is, untouched). It can be used with HIMEM.SYS, or as a stand-alone program. If used with HIMEM, H2UMP must be loaded after HIMEM.

H2UMP is similar to EMM386.EXE for loading devices into HMA. Unlike EMM386 with the NOEMS switch, H2UMP supplies DOS with memory from the first megabyte of the system memory. This is memory that would otherwise be available for use as shadow RAM.

H2UMP is much smaller and simpler than EMM386. Since EMM386 is a virtual mode system extension, it also slows down processing somewhat as it handles special exception interrupts that are required when providing virtual memory to DOS. H2UMP uses less system memory, therefore making more memory available to the system, but does not use expanded memory.

A single command line switch is provided to force H2UMP to reserve memory block elements, other than those with option ROM signatures in them. The syntax for this switch is: /X=aaaa[,aaaa]

where the addresses specified are hexadecimal starting addresses for each block of memory to be reserved. The H2UMP provides two 64 KB blocks of upper memory; one at C000, and the other at D000.

EXAMPLE 1: DEVICE=H2UMP.SYS
 DOS=UMB
 DEVICEHIGH= . . .

EXAMPLE 2: DEVICE=HIMEM.SYS
 DOS=HIGH
 DEVICE=H2UMP.SYS
 DOS=UMB
 DEVICEHIGH= . . .

EXAMPLE 3: DEVICE=HIMEM.SYS
 DOS=HIGH
 DEVICE=H2UMP.SYS /X=D000
 DOS=UMB
 DEVICEHIGH= . . .

Example 3 reserves the D000 address range for PC Card support.

Converting Applications



This section contains information applicable for converting 4000 Series Applications for use on a 6000 Series Computer.

Converting to the 6000 Series Computers

When applications are converted from a 4000 Series System to a 6000 Series Computer, consider the following information:

Files No Longer Supported

The following 4000 Series Files are not used in the 6000 Series Configurations:

File	Comments
10X16.EXT	Use FONTMAP.EXE (see page 3-3)
BOOT.SYS	
BOOTCFG.SYS	
BOOTPH0.SYS	
CATFILES.EXE	Use IPLFMT.EXE (see page 3-3)
CATMAKE.BAT	Use IPLFMT.EXE
CATPREAM.BIN	
DEXIO.BIN	Use PC-DEXIO.BIN
FONTBUF.COM	
FPNOP.COM	
IO.SYS	ROM DOS 5 is loaded from flash
MAXI-DOS.SYS	ROM DOS 5 is loaded from flash
MEMIO.EXT	XLMEMIO is not supported
MINI-DOS.SYS	ROM DOS 5 is loaded from flash
MINI-IO.SYS	ROM DOS 5 is loaded from flash
MINI-NET.COM	Use MININET.EXE (see page 3-3)
NOR-ANSI.SYS	
NORANDBB.EXE	
NP4805.EXT	The 4805 Endcap Printer is specific to 4000 Series systems.
NPRTBIOS.EXT	Use PC4800.SYS (see page 3-2)
NT4800.SYS	Use PC4800.SYS (see page 3-2)

File	Comments
PRTBIOS.EXT	RCT printers (NP207, NP111) are not supported
RAMCARD.SYS	
RAMCFMT.EXE	
RAMDISK.EXT	A RAM drive is created as drive C: by Norand [®] Utilities or the RAMDFMT utility. The RAM drive on 4000 Series Computer is Drive B.
RPLHOST.EXE	Use INTERLNK.EXE
SCAN4000.EXE	
SETDISP.EXE	Use FONTMAP.EXE (see page 3-3)
SOFTBIOS.EXT	BIOS is loaded from flash.
TNETBIOS.EXE	
XLMEMIO.EXE	XLMEMIO is not supported.

Changed Files

The following 4000 Series Files are used differently on a 6000 Series Computer. You can make the changes as specified.

CONFIG.SYS The extended CONFIG.SYS commands in the 4000 Series version of IO.SYS are no longer supported:

Command	Comments
EXT=	BIOS is loaded from flash. Remove these lines from CONFIG.SYS.
TSR=	Use INSTALL= or execute Terminate and Stay Residents (TSR) from AUTOEXEC.BAT.
VERIFY	The VERIFY ON command, a feature of COMMAND.COM, can be used. However, this command is not identical with its use on 4000 Series platforms, where MINI-DOS/MAXI-DOS wrote through the disk buffers, as well as to directory information to the media whenever it changed. These actions do not occur in DOS. The application is responsible for performing file commits.

See the CONFIG.SYS file in your DOS toolkit, to see how CONFIG.SYS is used for applications with the 6000 Series Computer.

CPLNI.COM Do not use the *-d* command line switch. Do not delete driver files from the RAM drive after they are loaded. All files needed to restart the application after a reset must remain on the RAM drive.

PC4800.SYS Use Version 1.80 or later for a 6000 Series Platform. This driver is used for Norand Portable Communications Protocol (NPCP) printer support for both PL/N and non-PL/N applications. 4000API.EXE must be loaded when PC4800.SYS is used.

PC-DEXIO.BIN Use Version 1.26 or later for a 6000 Series Configuration. 4000API.EXE must be loaded when PC-DEXIO.BIN is used.

New 6000 Series Files

The following files are new 6000 Series Files that were not used on the 4000 Series Platform:

AUTOEXEC.BAT

This is a standard DOS configuration file. See the AUTOEXEC.BAT file in the DOS toolkit to see examples of how this file is used.

4000API.EXE

This program provides functionality for certain applications written for the 4000 Series to run unmodified on the 6000 Series Platform. See page 3-4 for additional information.

FONTMAP.EXE

Use FONTMAP.EXE to change the screen font. This program also provides functionality expected by applications written for the 4000 Series so that these applications can run unmodified on the 6000 Series Computer. See page 2-16 for additional information.

***.FNT**

Font files for use with FONTMAP.EXE.

IPLFMT.EXE

This program concatenates a list of files into a single file suitable for download in a Norand communications session. For additional information, see page 2-3.

MININET.EXE

This program replaces the 4000 Series MINI-NET.COM on the 6000 Series Platforms. It provides the NET BIOS interface to the NPCP LAN. To use NPCP, load MININET.EXE, a TSR that supplies a NetBIOS compatible interface to access the NPCP LAN. For additional information, see page 2-8.

Unchanged Files

The following 6000 Series files have not changed from the 4000 Series.

- ▶ CLKIO.BIN
 - ▶ DELETE.COM
 - ▶ HOSTIO.BIN
 - ▶ KBDIO.BIN
 - ▶ LZEXE.DOC
 - ▶ LZEXE.EXE
 - ▶ MEMIO.BIN
 - ▶ INT15.EXE
 - ▶ MV.EXE
 - ▶ NORSESS.COM
 - ▶ PRN2COM.COM
 - ▶ PRN2COM.SYS
 - ▶ PRTIO.BIN
 - ▶ TTYIO.BIN
 - ▶ XYXFER.COM
- ▶ INT15.EXE (*For a PC only, do not run on PEN*KEY 6000 Series platform*)

4000 Series Programming Interfaces

4000API.EXE is a TSR that supplies Application Programming Interfaces (APIs) that are compatible with those previously supplied on 4000 Series Terminals. With the capability to port 4000 Series Applications to the 6000 Series Environment, the amount of work required to port an application is greatly reduced.

Note that 4000API functions are not industry standards. The use of certain 4000API functions limits the ability of the application to function on standard platforms. If you desire portability and standardization of applications, consider modifying old programs and developing new programs to conform to industry standards. The 6000 Series platforms are PC-compatible.

Installation and Configuration

To load 4000API.EXE for use with a C++ application (disabling the interrupt override) use the following command:

```
4000API.EXE /C3 /PC /10 /14 /16
```

To load 4000API.EXE for use with a PL/N application, use the following commands:

- ▶ If an internal modem is used:

```
4000API.EXE /C3 /16 /10 /PC
```

- ▶ If an internal modem is not used:

```
4000API.EXE /16 /10 /PC
```

▶ **NOTE:** *Either the slash (/) or the dash (-) can be used for command line switches.*

Command Line Switches

Typically, you should supply the following switches, unless your application is a PL/N application originally designed to run on a 4000 Series Computer.

- /C3** Do not redirect COM3 to COM1. Do not ignore modem control signals. Existing PL/N applications frequently use COM3 to communicate with COM1, only without using modem control signals.
- /PC** Change the value returned by INT 15h AX=01FBh. The default value is "SOFT BIOS V3.00". When this switch is supplied, the value returned is "PC BIOS V3.00".
- /10** Do not trap INT 10h (video BIOS interrupt). There are a few 4000-compatible video BIOS extensions enabled by default and disabled with this switch.
- /14** Do not trap INT 14h (COM port BIOS interrupt). By default, 4000API supplies 4000-compatible BIOS extensions to communicating with COM ports. Supplying this switch, the 4000API services are replaced by the COM port BIOS (from ROM BIOS).
- /16** Do not trap INT 16h (keyboard BIOS interrupt). By default, 4000API emulates the 4000 keyboard, complete with the ability to remap keys. Since a 6000 Series Computer has INT 16h support, you can apply this switch, resulting in PC-compatible key codes returned instead of 4000-style key codes.

System Information



This section contains system information for the PEN*KEY® 6220 Computer.

If the binary image files are downloaded from the Intermec Customer Support Bulletin Board System (BBS), go to page 4-2 to create four flash update disks from the image files.

If you received your flash update on four disks, then refer to the appropriate paragraph for one of the following methods for reflashing.

- ▶ Updating flash from an application supporting soft-downloading (page 4-2)
- ▶ Updating flash using INTERLNK (page 4-3)
- ▶ Updating flash using an ATA card (page 4-4)
- ▶ Updating flash using an Initial Flash Load (IFL) SRAM card (page 4-5)

Binary Image Files Posted to Bulletin Board

6220FL2A version 1.04 files are renamed before being posted to the BBS. If you are downloading these files, Table 4-1 has the files as they appear in 6220FL2A, their BBS equivalent, and descriptions. See 6220FL2A.TXT for instructions.

Table 4-1
6220FL2A BBS Equivalents

6220FL2A Filename	BBS Filename	Description
RELNOTES.TXT	FL20A104.TXT	PEN*KEY 6220FL2A Flash Release V1.04 Release notes, including annotated file listing.
README.TXT	6220FL2A.TXT	PEN*KEY 6220FL2A Flash Release V1.04 README.TXT Contains reflash instructions for 6220FL2A flash version 1.04.
6220FL2A.INC	6220FL2A.104	6220FL2A Flash Update, version 1.04 download file Reflashes from 6220FL2A version 1.03 to version 1.04. Download this file and rename it 6220FL2A.INC.
6220FF2A.INC	6220FF2A.104	6220FL2A Full Flash, version 1.04 download file Reflashes from any version of any 6220 flash to 6220FL2A version 1.04. Download this file and rename it 6220FF2A.INC.
230125-2.EXE	20A104-2.EXE	6220FL2A Flash, Disk #2 of 4, version 1.04 Self-extracting executable with DOS system files and 6220FL2A.EXE. Used to create a bootable ATA card to reflash the 6220 Computer or to reflash the 6220 Computer using INTERLNK.
230125-3.EXE	20A104-3.EXE	6220FL2A Flash, Disk #3 of 4, version 1.04 Self-extracting executable with 6220FL2A.IFL, a binary image that creates a 4 MB SRAM card to reflash a 6220 Computer.
230125-4.EXE	20A104-4.EXE	6220FL2A Flash, Disk #4 of 4, version 1.04 Self-extracting executable with 6220FL2A.BIN, a binary image that creates a temporary 2 MB SRAM flash card.

Flash Release Contents

Locate the following files on the four flash upgrade disks. If you downloaded these files from the BBS, see Table 4-1 for their equivalents. All diskettes contain a README.TXT file with information similar to what is in this section.

► **NOTE:** Use an SRAM card when the 6220 Computer flash is corrupted or was never programmed.

Non-Ethernet 6220FL2A

Diskette #1 (230125-1) Soft-download files:

- ▶ 6220FL2A.INC Application soft-download file (update).
- ▶ 6220FF2A.INC Application soft-download file (full reflash).

Diskette #2 (230125-2) MS-DOS system and ATA card files:

- ▶ 230125-2.EXE Self-extracting executable with MS-DOS system files and Initial Program Load (IPL) files required to build a bootable ATA card used to reflash a terminal. This also reflashes with INTERLNK.

Diskette #3 (230125-3) 4 MB SRAM card files:

- ▶ 230125-3.EXE Self-extracting executable with DD.EXE and an IPL binary file required to build a 4 MB bootable SRAM card to reflash a terminal.

Diskette #4 (230125-4) 2 MB SRAM card files:

- ▶ 230125-4.EXE Self-extracting executable with DD.EXE and a flash binary file that can be programmed to a 2 MB SRAM card, then booted directly.

Ethernet 6220FL2B

Diskette #1 (230126-1) Soft-download files:

- ▶ 6220FL2B.INC Application soft-download file (update).
- ▶ 6220FF2B.INC Application soft-download file (full reflash).

Diskette #2 (230126-2) MS-DOS system and ATA card files:

- ▶ 230126-2.EXE Same as 230125-2.EXE.

Diskette #3 (230126-3) 4 MB SRAM card files:

- ▶ 230126-3.EXE Same as 230125-3.EXE.

Diskette #4 (230126-4) 4 MB SRAM card files:

- ▶ 230126-4.EXE Same as 230125-4.EXE

Updating Flash Using an Application Soft-Download

Use this procedure to reflash a 6220 Computer as part of a normal application download session.

*The following example procedures are written for the 6220FL2A flash release, file-names are from the flash release diskettes, the floppy disk drive is assumed to be **A:**, and the 6220 PC Card slot is assumed to be **E:**.*

If applying flash release 6220FL2B, replace all “A” files with “B” files and replace all “125” files with “126.”

Prerequisites

- ▶ AN Intermec host PC and associated network.
- ▶ An application that supports the flash soft downloading.
- ▶ A soft download include file contained on diskette #1 of the flash release kit: either 6220FL2A.INC or 6620FF2A.INC.

▶ NOTE:

If certain that all the terminals being reflashed currently contain the previous release of the 6220FL2A flash, then use 6620FL2A.INC to quickly update to this version. Otherwise, use the 6220FF2A.INC full flash soft-download file.

Procedure

Use this procedure to reflash a 6220 Computer via an application soft-download.

1. Copy the appropriate soft-download file to the network download directory for the host PC.
2. Set up the download specifications for the application, such that the appropriate soft-download file is downloaded when a different version of flash is detected by the application. Details for this step should be documented with the application.
3. Perform a typical application download. During the reboot that occurs after that TCOM, the flash will be updated automatically.

Updating Flash using INTERLNK

Use this procedure to reflash a 6220 Computer using a serial connection between the 6220 Computer and a DOS-based PC.

Prerequisites

- ▶ A PC with MS-DOS 5 (or greater) installed. Note that PCs with Windows 95 and Windows 98 do not come with INTERLNK and do not work well with it.
- ▶ A serial cable connection between the PC and the 6220 Computer. This could be a connection to the 15-pin D-Sub or to a single dock in which the 6220 Computer is resting.
- ▶ The self-extracting executable contained on diskette #2 of the flash release kit, 230125-2.EXE. *Note the BBS file equivalent has a different name.*

Procedure

1. On the 6220 Computer, press and release the reset button, then press and hold **F4**. Once booted to Norand Utilities, release **F4**.
2. Press **Yes** to go to the main menu of Norand Utilities.
3. Press [**2**], then [**6**], to configure INTERSVR for the COMM SETTING.
4. Press [**1**] to begin a COMM session. If INTERLNK is not loaded on the PC, you first need to add the following line to CONFIG.SYS, then reboot.

```
DEVICE=C:\DOS\INTERLNK.EXE /NOSCAN
```

5. Using a cabled connection to the PC, run INTERLNK on the PC to connect to drive C of the 6220 Computer. For example, mapping a non-existent PC drive (such as E), sets up drive E as the PC drive connection to drive C of the 6220 Computer. This will connect PC drive E: with 6220 drive C:.

```
INTERLNK E=C
```

► NOTE:

If *INTERLNK* is run with no drive letters first, the available drive letters will be displayed.

- Build a temporary directory on the PC to contain files required to reflash the 6220 Computers.

```
C:
CD C:\
MD 6220FLSH
CD 6220FLSH
..\230125-2
```

- Copy the reflash code to the 6220 RAM drive.

```
Copy IFL\*. * E:\ /Y
```

- Reset the 6220 Computer. Place it on charge if not already done so. Press [5] to reflash as prompted.

Updating Flash Using an ATA Card

Use this procedure to prepare a bootable ATA card to reflash the 6220 Computer.

Prerequisites

- ▶ A PC with either Windows 95 or Windows 98 installed.
- ▶ A PC Card slot (in the PC). *For this example, drive D:*.
- ▶ A partitioned ATA card (*usually partitioned, formatted by manufacturer*).
- ▶ A directory with the flash release files contained within. *For this example, C:\6220FLSH.*
- ▶ The appropriate self-extracting executable from diskette #2 of the flash release kit, 230125-2.EXE.

Procedure to Build an ATA Card

- Build a temporary directory to hold the files required to build a boot card.

```
C:
CD C:\
MD 6220FLSH
CD 6220FLSH
..\230125-2
```

- Format the ATA card without any operating system components. *The /U switch ensure that all previous data is destroyed and that read and write errors are checked. However, this does prevent the card being unformatted afterwards.*

```
FORMAT /U d:
```

- Add the MS-DOS 6.22 components to the card. *Note that COMMAND.COM, IO.SYS, and MSDOS.SYS must reside in the current directory with the SYS6.COM program.*

```
SYS6 d:
```

- Copy the additional files required to perform the reflash process onto the formatted ATA card.

```
COPY IFL\*. * d:
```

Procedure to Reflash with an ATA Card

To boot the ATA card, place the card in either 6220 PC card slot, then press and release the reset button. Follow the display prompts to update the 6220 flash.

Updating Flash Using a 4 MB SRAM Card

Do this procedure to prepare a 4 MB bootable SRAM card to reflash a 6220 Computer. You can use an SRAM card to reprogram the 6220 Computer flash even if the flash is corrupted or perhaps never programmed.

Prerequisites

- ▶ A PC with either Windows 95 or Windows 98.
- ▶ A PC Card slot (in the PC). *For this example, drive D:*
- ▶ A directory with the flash release files contained within. *For this example, C:\6220FLSH.*
- ▶ A 4 MB SRAM card.
- ▶ Self-extracting executable from diskette #3 of the flash release kit, 230125-3.EXE.

Procedure to Build IFL SRAM Card

Use this procedure to build an SRAM IFL card:

1. Build a temporary directory to hold the files required to build a boot card.


```
C:
CD C:\
MD \6220FLSH
CD \6220FLSH
..\230125-3
```
2. Format the SRAM card and verify that FORMAT indicates a 4 MB card.


```
FORMAT /U d:
```
3. Reprogram the card with the IFL card image:


```
DD 6220FL2A.IFL d:
```
4. Remove the temporary directory.


```
CD ..
DELTREE 6220FLSH
```

▶ **NOTE:** *It is possible for verify to fail even though the write operation succeeded. If that happens, use the CRC32 program on all the individual files and check the results against what is shown in the RELNOTES.TXT file.*

▶ **NOTE:** *Do not use the DOS COPY or XCOPY command to create the IFL card. Master mode booting requires certain files at specific locations on the SRAM card. DD.EXE places every byte in a specific location on the media; the DOS commands do not attempt to place files in the same locations from one media to another.*

Procedure to Reflash with an IFL Card

Place the IFL card in the 6220 PC Card slot closest to the display, then press and release the reset button. Follow the prompts on the display to update the 6220 flash.

Recommended Methods

Partial Flash Download

6220FL2A.INC and 6220FL2B.INC are small (54 KB) download include files that can be used during an application TCOM session to soft-download the flash update.

For this method to work, all reflashed 6220 Computers must start with the same version of the same flash. If there is a mix of 6220FL2A.IFL and 6220FL2B.IFL then this method does not work. The best approach is to use this method to update the majority of the terminals. An ATA card can then be used on any terminals that fail during this process because of which flash they had initially.

ATA Boot Card

You can place the 230125-2.EXE or 230126-2.EXE reflash code on an ATA card and boot the code from either PC card slot to reflash a 6220 Computer. After re-flashing and removing the card, boot the 6220 Computer back to the application. The only major drawback to this approach is the amount of involvement required per unit.

SRAM Boot Card

You can use a 4 MB SRAM card (230125-3.EXE or 230126-3.EXE) to reflash your 6220 Computer. The only advantage to using an SRAM card over an ATA card is that an SRAM card can be used even when flash in the 6220 is corrupted. Since that is unlikely and SRAM cards are more expensive, the ATA card is the preferred method.

Full Flash Download

6220FF2A.INC and 6620FF2B.INC are large (900 KB) download include files that can be used during an application TCOM session to soft-download the full new flash. It can be used to reflash regardless the current flash version.

Section 5

Communications



This section covers INTERLNK and INTERSVR products from Microsoft Corporation and are bundled with MS-DOS. This also includes Norand[®] Utilities from Intermec Technologies Corporation and a telecommunications session overview.

Using *INTERLNK* and *INTERSVR*

INTERLNK connects a PEN*KEY[®] 6000 Series Computer and a PC through serial ports. INTERSVR is the INTERLNK server, a communication option in Norand Utilities. These two resources are shipped with the DOS Toolkit. The cable that connects the PC to your hand-held computer can be either a standard null modem cable or a TTY TCOM cable. You also need a dock.

INTERLNK makes the hand-held computer drives appear as virtual drives on the PC, with drive letters beyond the highest drive letter used on the PC.

After installing INTERLNK, type "INTERLNK" from the PC command line to display the designations for the redirected drives. See the diagram below. For more details, refer to the DOS online help text.

```
Port=COM2

This Computer          Other Computer
  (Client)              (Server)
-----
I:                    equals   D: (519Kb)
```

Norand Utilities

The Norand Utilities program, PSROM0C.EXE and PSROB0C.EXE, provides the basic functions required to prepare the 6000 Series Computer for use. It can also be called from applications to perform data communications.

Initial Program Load

Most of the communication options for performing an initial program load (IPL) require a download include file on the host computer. See page 2-3 for details on creating a download include file.

The following file names cannot be used in a download list file because they are used by Norand Utilities to perform communications:

- ▶ NRTLOG.DAT ▶ PSROM0C.DAT
- ▶ NRUPLD.CTL ▶ PSROM0C.INI

The Norand Utilities program specifically requests a <workgroup>.INC file, where <workgroup>.INC is the WORKGROUP field value in the UNIT ID menu.

NPCP Network

NPCP is the Norand Portable Communications Protocol supported by the 6920 Communications Server, the 4920 Series Telecommunications Packages, 498x Communication Controllers, and 6980/6985 Network Managers.

1. Create a download include file, as described on page 2-3.
2. Create a 4920 boot disk, if necessary. The only file on a 4920 boot disk is a download include file. The include file is copied to the DOWNLOAD subdirectory of the 4920.

► **NOTE:**

*4000 Series applications are copied to a BOOT subdirectory.
6000 Series applications are copied to the DOWNLOAD subdirectory.*

3. Create a 4980 boot disk, if necessary (there is limited space on the 4980 system). Place the 4980 system files on the 4980 boot disk as is done for 4000 Series applications (such as using the 4980 Boot Toolkit). Place the download include file on the disk by executing the following commands:

```
MD A:\DATA
COPY <include file> A:\DATA
NCDIR.EXE A:\DATA
```

► **NOTE:**

NCDIR.EXE is included in the 4980 Boot Toolkit.

► **NOTE:**

Norand Utilities first attempts a session to NORAND_SERVER, which allows the application to be retrieved from a 498x Communications Controller or 6980/6985 Network Manager. If the application is not stored on the network manager, a session is initiated to NORAND_HOST, which connects to the host.

NRInet

NRInet is a protocol that performs a Norand file transfer session over TCP/IP Ethernet. It is supported by the 6920 Communications Server.

1. Create a download include file, as described in the preceding paragraphs, and place it in the download directory of the 6920 Communications Server.
2. Optionally, configure a DHCP server to provide information required by the hand-held computer, including the IP addresses of the client, router, and subnet mask. Any information not provided by DHCP must be entered manually by the user on the hand-held computer. A DHCP server can also provide a domain name and IP addresses of domain name servers, which allows you to enter a host name rather than an IP address. If the SERVER_NAME field on the hand-held computer is left blank, a connection to the name “Norand6920” is attempted, as a default.

Version 2.xx of the Norand Utilities program creates a NET.CFG file and a PCTCP.INI file on the RAM drive. The NET.CFG and PCTCP.INI files are not erased, so they may be used by applications.

TFTP

TFTP (Trivial File Transfer Protocol) is a standard TCP/IP protocol supported on many TCP/IP servers.

1. A TFTP service must be running on a TCP/IP server.
2. Optionally, configure a DHCP server to provide information required by the hand-held computer (IP addresses of the client, router, and subnet mask). Any information not provided by DHCP must be entered manually by the user on the hand-held computer. A DHCP server can also provide a domain name and IP addresses of domain name servers, which allows the hand-held computer user to enter a server name rather than an IP address. If the SERVER NAME field on the hand-held computer is left blank, a connection is attempted to the name “NorandTftp” as a default.

3. Create a download list file, as described previously, and place it in the default working directory for the tftpd service. The list file must be named <workgroup>.BCF where <workgroup> is the value of the WORKGROUP field in the UNIT ID menu.

TTY

TTY is a protocol that is supported by the 6920 Communications Server and 4920 Series Telecommunications Package.

- ▶ Create a 4920/6920 boot disk, as described in the preceding paragraphs for the NPCP Network.

▶ NOTE:

See the *6220 Hand-Held Computer User's Guide (P/N: 961-028-104)* for a list of errors that may be displayed during an IPL.

Calling Norand Utilities from Application

The following paragraphs describe how an application can use the communications facilities of the Norand Utilities to perform data communications.

PSROM0C.EXE is the primary program module that supports data communications between a 6000 Series Computer and a host computer. It supports the two-way TTY, NPCP Local Area Network (LAN), and NRInet protocols.

PSROB0C.EXE is not called directly by applications, but is called by PSROM0C.EXE. Use the PSROM0C.EXE *ctl-file* syntax on the command line, where *ctl-file* is the name of a text file containing parameters that control the telecommunications session. Valid parameters are described in the following paragraphs. Parameters that are not applicable may be omitted.

The return value from PSROM0C.EXE is a session status code. Values for this code are described on page 5-8.

System Setup Requirements

To use PSROM0C.EXE, the file NRTCERR.TBL must exist in the current working directory or in the PATH.

When using PSROM0C.EXE Ver. 2.00 or later, if the application uses CardSoft device drivers to access PC Cards, it must reassign the drive letters A and B.

For example: ASSIGN.COM A:=E: B:=F:

NPCP

To use NPCP, you must load the program MININET.EXE. You can install this program by inserting the INSTALL=MININET.EXE statement into the CONFIG.SYS file or by placing the <path>\MININET.EXE statement into the ROMINIT.BAT or AUTOEXEC.BAT file, where <path> \ is the drive letter and pathname where MININET.EXE is located.

TTY

To use the TTY protocol, the NRTTYM.TBL file must exist in the current directory or in the Path. NRTTYM.TBL is the modem table. For information about creating a customized modem table, see the *PSMDM0C, DOS HHC MODEM TABLES* product, P/N: 215-968-001.

If the application uses NORMOD.SYS or NGENMOD.SYS to support PC Card modems but uses different command line settings than the default Norand Utilities configuration, the command line for the driver must include “-NMODEMn” where “n” is the COM port number used by the driver.

NRInet Using PSROM0C Version 3.xx

To use the NRInet protocol version 3.xx, the PATH must include the directories of the following drivers:

- ▶ LAN9000.COM ▶ ODIPKT.COM
- ▶ LSL.COM ▶ RS485ODI.COM

Optionally, configure a DHCP server to provide IP information required by the computer, such as the IP addresses of the client, server, router and domain name servers.

The NET.CFG and WATTCP.CFG files are overwritten by PSROM0C.

Some computers support two network interfaces for NRInet: Ethernet and RS-485. The network interface is selected in a menu in Norand Utilities during the initial program load sequence. This setting is stored on the RAM drive so it is lost any time the RAM drive is removed or formatted. For example, applications that boot from a PC Card often format the RAM drive before installing the application. In these cases, return to the Norand Utilities menu to reselect the setting in order for NRInet sessions to work correctly.

NRInet Using PSROM0C Version 2.xx

To use the NRInet protocol version 2.xx, the PATH must include the directories of the following drivers:

- ▶ BOOTP.EXE ▶ LAN9000.COM
- ▶ DHCP.EXE ▶ LSL.COM
- ▶ ETHDRV.EXE ▶ ODIPKT.COM
- ▶ INET.EXE ▶ RS485ODI.COM

BOOTP.EXE and DHCP.EXE are required only if IP information is retrieved from a BOOTP server or a DHCP server as described below. PSROM0C.EXE executes BOOTP.EXE first. DHCP.EXE is executed only if BOOTP.EXE fails.

The files NET.CFG and PCTCP.INI are required to load these drivers.

NET.CFG must exist in the current working directory and must include the following, indented as shown:

```

Link driver LAN9000
  INT 5
  PORT 300
  MEDIA_TYPE AUTO
  Frame Ethernet_II

Link driver RS485ODI
  Frame Ethernet_II

```

PCTCP.INI must include:

```

[pctcp ifcust 0]
async-send = yes
ip-address = nnn.nnn.nnn.nnn
subnet-mask = nnn.nnn.nnn.nnn
router = nnn.nnn.nnn.nnn

[pctcp general]
etc-dir = d:\tcp\etc
domain = xxxxxxxx.com

[pctcp addresses]
domain-name-server = nnn.nnn.nnn.nnn

[pctcp kernel]
interface = ifcust 0

```


The “etc-dir” field must specify a directory that contains a SERVICES file. A minimal SERVICES file exists in the flash at d:\tcp\etc. For NRInet, SERVICES must contain a protocol entry for “nrinet” with a protocol type of “tcp”. For example: nrinet 44965/tcp #Norand Inet File Transfer

The fields “domain” and “domain-name-server” are not required if the host is specified by IP address rather than domain name. Also, the fields that specify IP addresses (nnn.nnn.nnn.nnn) and the “domain” field can be omitted if BOOTP.EXE or DHCP.EXE loads the information from a BOOTP server or a DHCP server.

Set the environment variable PCTCP to the path of PCTCP.INI. For example, in AUTOEXEC.BAT: SET PCTCP=C:\PCTCP.INI

Valid Control File Parameters

- ▶ **For all protocols:**
PROTOCOL, TRIES
- ▶ **For TTY:**
COM, CONFIG, DATABITS, MODEMSELECT, MODEMTYPE, PARITY, PHONE, SPEED, STOPBITS
- ▶ **For NPCP:**
COM, NPCPHOST
- ▶ **For NRInet:**
IPCLIENT, IPHOST, NETMASK, ROUTER

The following is a list of control file parameters and their descriptions.

COM=

The COM port to use for two-way TTY or NPCP communications.
(Example: COM=4)

Default Values:

- 1 (COM1) = Two-way TTY
- 2 (COM2) = NPCP

CONFIG=

Modem command string to configure a Hayes-compatible modem.
(Example: CONFIG=ATE0V0Q0X4&C1&D2&M0&RS0=0)

DATABITS=

Valid only if two-way TTY protocol was specified. (Example: DATABITS=7)

Valid values:

- 7, 8

IPCLIENT=

Supported in PSROM0C V3.xx only.

This is the IP address of the client computer. If this parameter is specified, the NETMASK parameter is also required. If this parameter is omitted, PSROM0C attempts to retrieve IP information from a DHCP server or a BOOTP server.

IPHOST=

This is the IP address or domain name of the Ethernet host. The value can be at most 18 characters long. If this parameter specifies a domain name, the IPCLIENT parameter must be omitted because DHCP information about domain name servers is required to resolve the host name. For PSROM0C V3.xx, if IPHOST and IPCLIENT are both omitted, the host name is retrieved from DHCP if available. Otherwise, this parameter defaults to “Norand6920.”

MODEMSELECT=

Valid only if two-way TTY protocol was specified. This is the modem ID of a record in the modem table file NRTTYM.TBL. If this parameter is specified, the COM= and CONFIG= parameters may be omitted. The parameters MODEMSELECT= and MODEMTYPE= are mutually exclusive; therefore only one may be specified in a control file. (Example: MODEMSELECT=30)

MODEMTYPE= Valid only if two-way TTY protocol was specified. It indicates the type of modem, if any, to which the hand-held computer is connected. The parameters **MODEMSELECT=** and **MODEMTYPE=** are mutually exclusive; therefore only one may be specified in a control file. (Example: *MODEMTYPE=1*)

Valid Values:

- 0 = No modem, i.e. direct connect (*default*)
- 1 = Norand modem, or other Hayes-compatible modem

NETMASK= *Supported in PSROM0C V3.xx only.*
This is the local subnet mask and is required if **IPCLIENT** is specified.
(Example: *NETMASK=255.255.240.0*)

NPCPHOST= NPCP host name.

Valid values:

- NORAND_HOST
- NORAND_SERVER

PARITY= Valid only if two-way TTY protocol was specified. (Example: *PARITY=2*)

Valid values:

- 0 = None (*default*)
- 1 = Odd
- 2 = Even

PHONE= Dials a Hayes-compatible modem. This value can be at most 20 characters long.
(Example: *PHONE=ATDT3693361*)

PROTOCOL= Indicates the protocol to be used. (Example: *PROTOCOL=3*)

- ▶ 3 = TTY
- ▶ 4 = NPCP (LAN)
- ▶ 14 = NRInet

ROUTER= *Supported in PSROM0C V3.xx only.*
This is the router IP address.

SPEED= TCOM speed for two-way TTY. Valid only if two-way TTY protocol was specified.
(Example: *SPEED=2400*)

Valid values:

- 1200, 2400, 9600, 19200, 38400, 57600, or 115200.

STOPBITS= Valid only if two-way TTY protocol was specified. (Example: *STOPBITS=1*)

Valid values:

- 1 = One stop bit
- 2 = Two stop bits
- 3 = 1 1/2 stop bits

TRIES= Specifies the number of times to attempt a successful communication session. If **TRIES** is exhausted before a session is successful, **PSROM0C.EXE** returns the error of the last session. A parameter value of zero indicates that the retry is to continue until a session is successful or the user aborts.

EXAMPLE: Example control file for NPCP:
 PROTOCOL=4
 Example control file for NRInet
 PROTOCOL=14
 IPHOST=nnn.nnn.nnn.nnn

Example control file for TTY
 PROTOCOL=3
 COM=1
 SPEED=9600
 PARITY=0
 DATABITS=8
 STOPBITS=1

Upload Control File Parameters

Information about files to be transferred must be contained in a NRUPLD.CTL file. Valid parameters for this file are described in the following paragraphs.

- FILE=** Name of upload file, uploaded according to the preceding header information. Multiple FILE parameters may follow a single HEADER parameter, if the header applies to all the specified files. (*Example: FILE=BYPRD.DAT*)
- FORMAT=** Defines the format of variable-length records whose record types match the preceding RECTYPE parameter. The format of a record consists of the data type (“t”) and length of each field within the record (“nnn”), specified as follows:
 FORMAT=t_{nn}n_{tt}n_{nn} . . .
 The field length is right-justified and zero-padded. Although the Norand file transfer protocol supports a number of data types, many of these are specific to the PL/N language from Intermecc Technologies Corporation. For the sake of simplicity, just specify a data type of X (for character) and the record length, which does not include the record type character. If you need more information on PL/N file formats, see the host TCOM manual.
Example: FORMAT=X020
 FORMAT=N012X016N004
- HEADER=** Precedes the files subsequently specified in FILE parameters. This header record provides the host information on data formatting.
 If the HEADER parameter is omitted or blank, the files subsequently specified are transferred *as is*; that is, no header precedes the file. Any header information is assumed to be embedded in the file itself.
 When the binary file descriptor (DOSFIL) is specified, a complete specification of the header is not needed. If the size is not specified or is zero, the entire file is uploaded, and the actual size is inserted into the header uploaded to the host. If the file name is not specified, the FILE parameter file name is inserted into the header uploaded to the host. (*Example: HEADER=<DBYPRD 00000x040>*)
- RECTYPE=** RECTYPE and FORMAT parameters upload files that contain variable-length records, in which the first character of each record identifies the type of record. These parameters remain in effect only until the next HEADER parameter is encountered. (*Example: RECTYPE=A*)
 RECTYPE is a single, printable ASCII character; it indicates the record type to which the following FORMAT applies.

► **NOTE:** *At a minimum, the NRUPLD.CTL file must specify a session control file or a download request file. For a description of these files, see page 5-12.*

EXAMPLE: Example NRUPLD.CTL
 HEADER=<DSCNTR100001X042>
 FILE=SCNTRL.DAT
 HEADER=<DBYPRD 00000X040>
 FILE=BYPRD.DAT
 HEADER=<DBYTRXN00000X001>
 RECTYPE=A
 FORMAT=N004N005
 RECTYPE=B
 FORMAT=X010
 RECTYPE=C
 FORMAT=N004X005
 FILE=BYTR00.DAT
 FILE=BYTR01.DAT
 file=bytr02.dat

```
FILE=BYTR03.DAT
HEADER=<DDOSFIL00001X010>FIMAGE.PCX
FILE=IMAGE.PCX
HEADER=<DDOSFIL00001X020>F\RT00001\COMMON.DAT
FILE=COMMON.DAT
HEADER=<DDOSFIL00001X000>
FILE=MYFILE.DAT
```

Communications Log File

A log of the communications is output to a text file named NRTLOG.DAT. Information is continually appended to the file until it is deleted. It is not cleared by PSROM0C.EXE. Your application is responsible for deleting NRTLOG.DAT.

The format and syntax are described in the following paragraphs.

BEGS=nnnn Indicates start of TCOM session. “nnnn” equals the session number (currently always 1). *EXAMPLE: BEGS=1*

DOWN=d:\pathname\filename.nnn

Indicates that a file download was attempted. “nnn” is the error code. Zero is a valid value that indicates a successful download.

EXAMPLE: DOWN=C:\CUST.DAT,23

ENDS=x,m,nnn Indicates the end of a TCOM session. “x” is the session status (a single character code), “m” is the stage of the communications session, and “nnn” is the protocol error code. The code is also the return value of PSROM0C.EXE.

The following table shows the valid values for (x).

Code	Meaning
“G”	Good session.
“T”	Unexpected end of transmission.
“H”	An incorrect file header was encountered.
“F”	A file error was encountered.
“L”	Communication aborted before first file header

The following table shows the valid values for (m).

Value	Meaning
5	Sign-on started.
4	Data send started.
3	Turn-around started.
2	Data receive started.
1	Sign-off started.
0	Session complete.

See the following *Protocol Errors* for valid values for (nnn).

UP=d:\pathname\filename,nnn

Indicates that a file upload was attempted where:
nnn = Error code. (*Example: UP=C:BYTRXN.DAT,0*)

Valid values:

0 indicates a successful upload.

Protocol Errors

With PSROM0C.EXE returning the session status code, the Communications Log file can be interrogated for a specific protocol error. The valid Protocol error codes for (n), as in the End of TCOM Session parameter "ENDS=", are listed.

Table 5-1
NPCP Protocol Errors

Error #	Meaning
0	No error.
1	MININET.EXE is not installed.
6	User aborted communications by pressing EXIT.
11	An invalid parameter was specified in the control file.

The following errors are returned by MININET.EXE. 100 is added to the error returned by MININET.EXE to avoid conflict with other defined errors.

Table 5-2
MININET Protocol Errors

Error #	Meaning
101	Illegal buffer length
103	Invalid command
105	Command timed out
106	Message incomplete
108	Illegal local session number
109	No resource available
110	Session closed
111	Command canceled
113	Duplicate name in local name table
114	Name table is full
115	Name is deregistered, command completed
117	Local session table full
118	Session open rejected
119	Invalid name number
120	No answer
121	Name not found
122	Name in use on remote adapter
123	Name deleted
124	Session ended abnormally
125	Name conflict
126	Incompatible remote device
133	Network interface is busy
134	Too many commands outstanding
135	Invalid LAN adapter number
136	Command completed while cancel occurring
138	Command not valid to cancel
164–179	Unusual network condition
180–354	Adapter malfunction

Table 5-3
NRInet Protocol Errors

Error #	Meaning
0	No error.
1	PSROBOC.EXE could not be loaded. Ensure the directory for PSROBOC.EXE is in the PATH.
6	User aborted communications by pressing [NO].
11	An invalid parameter was specified in the control file.
800	TCP/IP kernel could not be loaded. Possible causes for this are: <ul style="list-style-type: none"> ◆ (PSROM0C V2.XX only) NET.CFG or PCTCP.INI does not exist. ◆ (PSROM0C V2.XX only) The PCTCP environment variable is not set. ◆ The PATH does not include the directories of the drivers. ◆ The computer does not have an Ethernet ID or the Ethernet ID could not be accessed. ◆ (PSROM0C V3.XX only) Invalid client IP address. Verify the IPCLIENT parameter, or make sure the DHCP or BOOTP server is running.
801	Invalid client IP address. For PSROM0C V2.XX, verify the entry for 'ip-address' in PCTCP.INI, or ensure the DHCP or BOOTP server is running. For PSROM0C V3.XX, verify the IPCLIENT parameter, or ensure the DHCP or BOOTP server is running.
802	(PSROM0C V2.XX only) Invalid Service or Service Type, or invalid port number. Verify the 'etc-dir' entry in PCTCP.INI specifies the correct path for the SERVICES file, and that an entry exists for 'nrinet.'
803	Invalid host name or IP Address. Ensure the host computer is running. For PSROM0C V2.XX only, verify the IPHOST parameter and the entries for 'subnet-mask,' 'router,' 'domain,' and 'domain-name-server' in PCTCP.INI. For PSROM0C V3.XX, verify the IPHOST, NETMASK, and ROUTER parameters.
804	(PSROM0C V2.XX only) Could not create socket. Check all cables and network connections.
806	Block sent was incomplete, or block received was incomplete.
807	Client/server negotiation failed.
808	Server specified an unsupported block size.
809	Invalid buffer pointer.
810	All server connections are already in use. Try again later.
811	Timeout while sending data. Connection to remote machine dropped. Ensure host is still running; check all cables and network connections.
812	Timeout while receiving data. Connection to remote machine dropped. Ensure host is still running; check all cables and network connections.
813	(PSROM0C V2.XX only) Attempt to send data to the server failed due to a closed connection. Ensure server is still running, and check all cables and network connections.
814	Attempt to receive data from the server failed due to a closed connection. Ensure server is still running, check all cables and network connections.
815	Could not access the network attach information. Ensure the Access Point, such as 6710, is connected and running.
816	An error occurred reading the network attach information. Ensure the Access Point, such as 6710, is connected and running.
817	Server did not respond to the connect request. Ensure the server is still running, and check all cables and network connections.
818	An error occurred reading the TCP/IP kernel information.
935	Client/server negotiation failed.
939	Destination address required.
940	Message too long.
948	Address already in use.

Table 5-3 (Continued)
NRInet Protocol Errors

Error #	Meaning
950	Network is down.
951	Network is unreachable.
952	Network dropped connection or reset.
954	Connection reset by peer.
955	No buffer space available.
960	Connection timed out.
961	Connection refused.
962	Too many levels of symbolic links.
963	File name is too long.
964	Host is down.
965	Host is unreachable.
966	Directory not empty.

Table 5-4
TTY Protocol Errors

Error #	Meaning
0	No error.
6	User aborted communications by pressing EXIT.
11	An invalid parameter was specified in the control file.
23	End of transmission.
101	Line lost.
102	Parity error.
103	Character gap too long.
104	Data loss.
105	Excessive NAKs.
106	Block count error.
107	Block check error.
108	Block framing error.
109	Control character error.
2xx	Modem error. xx is Hayes response code or program-defined code:
03	No carrier.
04	Command not recognized.
06	No dial tone.
07	Dialed number is busy.
08	No answer.
86	Error sending command to modem.
87	Expected numeric response was not numeric.
88	Invalid response format.
89	No significant response from modem.
97	System disabled COM port — low battery, PC Card modem removed.
98	Unrecognized English response.
99	Memory allocation error.

TCOM Session

Under the Norand standard file transfer session, the hand-held computer always uploads first. Once all upload files are sent, the line is turned around and the host then sends any download files. Each file is preceded by a header record that gives the receiving computer information on the format used by that file.

The hand-held computer must first send a session control file, which identifies the hand-held computer to the host computer. The hand-held computer may then send a download request file. This prepares the host computer to download files to the hand-held computer. Upload files can then be sent to the host.

Session Control File

To identify itself to the host, the hand-held computer sends the session control file at the beginning of a TCOM session. The host uses the application-dependent terminal identification to identify the proper files for downloading to the hand-held computer. The format of the session control file is as follows:

```
<DSCNTRL00001Xnnn>PPPPPPPPPPPTTTTTTTTTTTTTTTTTTTTTTYMMDDHHMMSS . . .
  < = Beginning of file header
  D = File type (fixed)
  SCNTRL = File name (fixed)
  00001 = Decimal number of records in file (fixed)
  X = Data type (ignored by host)
  nnn = Decimal number of bytes in file record
  > = End of file header
  PPPPPPPPPPPPP = Program identification
  TTTTTTTTTTTTTTTT = Terminal identification (determined by application)
  YYMMDD = Date
  HHMMSS = Time
  . . . = Additional hand-held computer information
```

The application determines the actual data in the session control file. The data fields shown in the preceding list are simply a convention in Norand applications. However, the session control header always consists of 18 bytes. Further, Norand host communication packages usually expect to find the terminal identification starting in byte 33 of the data stream; this location is configurable, however.

Download Request File

A 6220 Computer may directly request specific files from the host by sending a download request file to the host just after sending the session control file. The files are downloaded after the line has been turned around. If a download request file is sent, the host does *not* use the terminal identification in the session control file to determine which files to download to the computer. Include as many filename records as specified in the number of records field of the header. This permits requests for more than one file to be batched together.

► **NOTE:**

The download request file must be the first or second file sent; otherwise it is not treated as a special file by the host.

```
<DDWNLRQnnnnnX016>[--filename1---][--filename2---]. . .
  < = Beginning of file header
  D = File type (fixed)
  DWNLRQ = File name (fixed)
  nnnnn = Decimal number of records in file
  X = Data type (X for character)
  016 = Decimal number of bytes in record (fixed)
  > = End of file header
  [--filename?---] = File to download, left-justified, blank-padded to 16 characters
```


Upload and Download Files

While the Norand file transfer protocol supports a number of data types and file types, many of these are specific to the Norand PL/N language. If more information is needed on PL/N file formats, see the host TCOM manual. For the sake of simplicity, DOS programmers can use one basic file header for all upload and download files. The general format of a file header is as follows:

```
<tffffffmmmmmtnnntnnntnnn>data ...
    < = Beginning of file header
    t = File type
        "D" for unpacked data
        "E" for unpacked executable
        "P" for packed data
        "B" for packed executable
    ffffff = File name
    mmmmm = Decimal number of records in file, right-justified and zero-padded
        t = Field data type (described below)
    nnn = Unpacked length of the field, right-justified and zero-padded
    > = End of file header
    . . . = Additional hand-held computer information
```

The actual name of the file that is created on the hand-held computer has ".DAT" or "P.PL6" appended to the ffffff file name. File types "D" and "P" have ".DAT" extensions. File types "E" and "B" have "P.PL6" appended to the name.

Valid field data types are as follows:

Type	C Data Type	Comments
X	char[]	Specifies a character buffer that does <i>not</i> have a NULL terminator.
B	unsigned char	File header field length can be 001 to 003. Valid field values are 0 to 255.
I	signed int	File header field length can be 001 to 006. Valid field values are -32767 to 32767. "+" is not uploaded for positive values. Leading zeroes are uploaded if necessary to meet the specified field length.
D	signed long	File header field length can be 001 to 011. Valid field values are -2147483647 to 2147483647. "+" is not uploaded for positive values. Leading zeros are uploaded if necessary to meet the specified field length.
S	char[]	Specifies a NULL terminated string. The field length does not include the NULL, as the NULL is not included in the unpacked file. The NULL is inserted when the field is written to the hand-held computer file.
N	not applicable	Specific to the Norand PL/N programming language. For further information, see the PL/N Reference Manuals.
W	not applicable	Specific to the Norand PL/N programming language. For further information, see the PL/N Reference Manuals. This type is equivalent to an unsigned integer in field size, but is packed in the opposite byte order.
(N/A	Marks the beginning of a repeated field descriptor sequence. The field length is the number of repetitions of the sequence. For example, the sequence: B002X004B002X004 can be written as: (002B002X004)000 "(002" marks the beginning of a pattern repeating twice. ")000" marks its end.
)	N/A	Marks the end of a repeated sequence begun with a "(nnn" descriptor. Field length is the number of repetitions of the sequence. For example, the sequence: B002X004B002X004 can be written as: (002B002X004)000 "(002" marks the beginning of a pattern repeating twice. ")000" marks its end.

In Norand host communication packages, a record length of one indicates the file contains variable length records, with the first character of each record identifying the record type. These single character record types are used by the formatting utilities to format the file into logical records.

PL/N File Descriptor for Binary Files

For the support of full DOS file names and for better support of non-PL/N binary files, an expanded header structure is defined:

```
<DDOSFIL00001Xmmm>F[--dosfilename--] S[filesize]data. . .
    < = Beginning of file header
    D = File type (fixed)
    DOSFIL = File name (fixed)
    00001 = Fixed
    X = Data type (fixed)
    mmm = Decimal number of bytes of file information between EOF header and
start of file
    > = End of file (EOF) header
    F = File name parameter
[--dosfilename--] = Complete DOS file name (may be of any length)
<space> = Separates the parameters
    S = File size parameter
[filesize] = Exact number of bytes in binary file (may be up to eight digits long)
data. . . = Binary file starts immediately after filesize parameter
. . . = Additional hand-held computer information
```

For example, to send a file named `\DATA\DATAFILE.DAT` that has a size of 102,000 bytes, the following header would precede the file:

```
<DDOSFIL00001X027>F\DATA\DATAFILE.DAT S102000
```

It is the responsibility of the 6000 Series Application Program to ensure that the file name specified is unique on the host. This can be done by including the hand-held computer ID as part of the file name or path name:

```
<DDOSFIL00001X027>F\DATA\DATA0001.DAT S102000
<DDOSFIL00001X031>F\HH000001\DATAFILE.DAT S102000
```


Flash Card

A memory storage PC Card that meets the ATA standard. (See also: ATA Card, SanDisk, Spinning Media, or Hardcard)

GND (Ground)

The common pin on the communications connectors, provided for electrical return path for the signals.

Host

A customer's host computer.

A computer that provides services directly to users, such as the user's computer. In TCP/IP, an IP addressed device.

A large computer that serves many users, such as a minicomputer or mainframe.

IFL Card (Initial Flash Load)

An SRAM card that installs the flash contents on most PEN*KEY systems.

IOCTL (I/O Control)

UNIX function call used to control a device.

IP Address

An IP defined 32-bit address usually in dotted decimal notation. See also: dot address, IP, network address, subnet address, host address.

IRQ (Interrupt Request)

A method involving a set of special address lines in PCs, connecting peripherals (such as a serial port or network adapter) to the processor and other computer architectures so the peripherals can request service from the processor.

LAN (Local Area Network)

A group of network devices in which each device can communicate through a wired or wireless link. The wired link may have several segments joined by repeaters and bridges. The LAN is characterized by the relatively short distance it is designed to cover, a high speed of operation, and relatively low error rates. The geographic scope of LANs is limited to thousands of feet or closely-spaced building complexes.

Landscape Mode

In terms of the display, it has an orientation such that the information displayed on the screen is viewed by rotating the image ninety degrees clockwise, resulting in a wide display.

NetBIOS (Network Basic Input/Output System)

An application programming interface (API) that augments the DOS BIOS by adding special functions for local area networks (LANs). Almost all LANs for PCs are based on the NetBIOS. Some LAN manufacturers have even extended it, adding additional network capabilities.

NetBIOS relies on a message format called Server Message Book (SMB).

Norand Utilities

A program that provides the basic functions needed to prepare the PEN*KEY for use, including program load and data communications.

NPCP (Norand Portable Communications Protocol)

A Norand proprietary protocol that provides session, network, and datalink services for Norand hand-held computers in the Norand LAN environment used with printers and data communications.

NRInet

A Norand protocol that transfers data over TCP/IP.

ODI (Open Data-Link Interface)

An Application Programming Interface (API) developed by Novell for writing network drivers. ODI separates the physical network layer (the Data-Link Layer in the OSI model) from the network protocol layer (the Transport Layer). As a result, the same network interface card (NIC) can carry data for different protocols. For example, ODI allows a computer with just one NIC to be simultaneously connected to both an IPX/SPX network and a TCP/IP network.

OEM (Original Equipment Manufacturer)

A misleading term for a company that has a special relationship with computer producers. OEMs buy computers in bulk and customize them for a particular application. They then sell the customized computer under their own name. The term is really a misnomer because OEMs are not the original manufacturers, they are the customizers.

To provide equipment to another company, an OEM, which customizes and markets the equipment.

PC (Personal Computer)

1. A desktop computer developed by IBM or a clone based on the same architecture developed by a third party vendor. 2. Sometimes used more generically to refer to other desktop systems, such as the Apple Macintosh. 3. The original IBM computer using an Intel 8088 CPU and an 8-bit internal bus.

PC Card

A device that fits in the card slots of an Intermec or other computer. In some PEN*KEY computers, the smaller Type II card occupies one slot, while in some cases a Type III card may block the second slot. The card may be used for data storage, modem, printer, wireless, or other purposes.

Generally, PC Card is used in reference to devices, whereas PCMCIA is a reference to the standard.

PC Card Icon

An icon located in the System Tray, which indicates a PC Card is plugged into the PC Card slot. (See also: System Tray)

PCMCIA (Personal Computer Memory Card International Association)

An industry group responsible for creation of the standard for devices (slightly larger than a credit card) for small computers, such as laptop and hand-held computers.

Formerly, devices were referred to as PCMCIA cards, but are now generally referred to as PC Cards, while PCMCIA is a reference to the standard.

PDF (Portable Document Format)

A file format developed by Adobe Systems. PDF captures formatting information from a variety of desktop publishing applications, making it possible to send formatted documents and have them appear on the recipient's monitor or printer as they were intended. To view a file in PDF format, you need Adobe Acrobat Reader 3.0 or greater. Adobe Reader is a free application distributed by Adobe Systems.

PL/N (Programming Language/Norand)

A third generation high level programming language proprietary to Intermecc. Many PLN applications, especially route accounting applications are ported to the 6100 and 62XX models from 4000 Series Hand-Held Computers.

Portrait Mode

In terms of the display, it has an orientation such that the information displayed on the screen is viewed in a normal manner, without rotating the image, resulting in a narrow display.

Protocol

A formal description of message formats and the rules computers must follow to exchange those messages.

RAM (Random-Access Memory)

Dynamic memory, sometimes known as main memory or core. When used by itself, "RAM" refers to read and write memory; you can both write data into RAM and read data from RAM. This is in contrast to ROM, which permits you only to read data. Most RAM is volatile, requiring a steady flow of electricity to maintain its contents. When power is turned off, data in RAM is lost.

RS-232 C (Recommended Standard 232)

An electrical interface standard approved by the Electronic Industries Association (EIA) for connecting serial devices. In 1987, the EIA released a new version of the standard and changed the name to EIA-323-D. And in 1991, the EIA teamed up with Telecommunications Industry Association (TIA) and issued a new version of the standard called EIA/TIA-232-E. Many people still refer to the standard RS-232C or just RS-232.

Almost all modems conform to the EIA-232 standard and most personal computers have an EIA-232 port for connecting a modem or other device. In addition to modems, many display screens, mice, and serial printers are designed to connect to a EIA-232 port. In EIA-232 parlance, the device that connects to the interface is called a Data Communications Equipment (DCE) and the device to which it connects (such as the computer) is called a Data Terminal Equipment (DTE).

RS-485 (Recommended Standard 485)

An electrical interface standard approved by the Electronic Industries Association (EIA) for connecting serial devices. RS-485 is used for multipoint communications. RS-485 is similar to RS-422 but can support more nodes per line because it uses lower-impedance drivers and receivers.

RTC (Real Time Control) Protocol (ADK)

A sophisticated multiplexing system using contention rules to open communication yielding faster and more efficient use of the radio channel.

RTS (Request To Send or Ready To Send)

A modem control signal on a standard RS-232-C connector that puts the modem in originate mode to start sending data.

RXD

Received Data.

SanDisk

A brand of flash memory card. (See also: ATA Card, Flash Card, Spinning Media, or Hardcard)

SG

Signal ground pin (or signal) on the communications connectors.

SIN

Signal input pin (or signal) on the communications connectors.

SOUT

Signal output pin (or signal) on the communications connectors.

SRAM (Static Random Access Memory)

A type of memory that is faster and more reliable than the more common DRAM. This memory is nonvolatile and does not need to be refreshed like DRAM.

While DRAM supports access times of about 60 nanoseconds, SRAM can give access times as low as 10 nanoseconds. In addition, its cycle time is much shorter than that of DRAM because it does not need to pause between accesses. Unfortunately, it also is much more expensive to produce than DRAM. Due to its high cost, SRAM is often used only as a memory cache.

SRAM Card

A memory storage PC Card which functions like a floppy disk.

TCP/IP (Transmission Control Protocol, Internet Protocol)

Most networks combine IP with a higher-level protocol called Transport Control Protocol (TCP), which establishes a virtual connection between a destination and a source.

IP by itself is something like a postal system. It allows you to address a package and drop it in the system, but there is no direct link between you and the recipient. TCP/IP establishes a connection between two hosts so they can send messages back and forth for a period of time.

Tethered

A scanning method requiring a cable between the computer and the scanner.

TTY

NORAND two-way TTY asynchronous data link protocol.

TXD

Transmit data pin (or signal) on the communications connectors.

WAN (Wide Area Network)

A network that covers a large geographic area. It is often used to connect two or more LANs together. This usually involves a variety of methods to maintain communications between all nodes in the network. For example, microwave communications, fiber-optic connections or leased telephone lines.

General Index

NOTE:

This index covers all topics. Those in italics are figures, those in bold are tables.

NUMBERS

12-pin, dock, connector location, 1-3
15-pin, connector location, *1-6*
 serial, 1-3
30-key keyboard, *1-2*
 remapping, 2-2
4-pin, connector location, RJ-11 phone
 jack, 1-3
4000 Series
 COM port BIOS interrupt, 3-4
 programming interfaces, 3-4
 screen emulation, 2-16
4920 Series Tcom Packages, 5-2
498x Communication Controllers, 5-2
56-key keyboard, *1-2*
6220 computer
 rear, top view, *1-5*
 top end, *1-6*
6220FF2B.INC, 4-6
6920 Communications Server, 5-2
6980/6985 Network Managers, 5-2

A

AC powered docks, 1-4
Accessory cards, creating a RAM drive,
 2-7
Advanced Power Management, 1-6
 clock driver, 2-1
 configuration, 2-2
Assign/unassign drive letters, drive map-
 ping utility, 2-4
ATA, cards, RAM drive, creating, 2-7
ATA flash card, updating flash, 4-4

B

Backlight, 1-3
Battery
 backup battery, 1-2
 charging in dock, 1-4
 main, checking, 1-3
 main battery, 1-2
 measuring capacity, 1-3
BGI support, 2-1
BIOS, 1-7
 in flash, 1-4
 updating, 1-7
Boot flags, get/set ROM DOS, DOS utility,
 2-5

Booting, from ATA card to create a RAM
 drive, 2-7

C

C++, loading 4000API.EXE for C++ ap-
 plication, 3-4
Calibration
 pen, DOS utility, CALIB.EXE, 2-12
 pen operation, 2-12
CardSoft, address range for PC Card sup-
 port, reserved, 2-7
Charger
 compatibility with hand-held comput-
 ers, 1-4
 list of types, 1-4
Clock driver (APM), 2-1
CMOS, 1-4
COM port
 15-pin serial location, *1-5*
 4000 series BIOS interrupt, 3-4
Command line switches, tethered scanner,
 2-19
Communications
 INTERLNK, INTERSVR, 5-1
 LAN device driver configuration, 2-8
 Norand Utilities, 5-1
 options, creating RAM drive, 2-7
Components, locations, *1-5, 1-6*
Configuration
 DOS drivers
 IrDA printer, 2-8
 keyboard, 2-2
 NPCP Printer, 2-9
 pen (mouse), 2-13
 pen calibration, CALIB.EXE, 2-12
 power management, 2-2
 scanner, 2-18
 utility, SC400CFG.COM, 2-2
Configuring
 keyboard, 2-2
 power management, 2-2
Connectors
 12-pin location, dock, 1-3
 15-pin location, serial, 1-3
 4-pin location, RJ-11 phone jack, 1-3
 top end of 6220 Computer, *1-6*
Control file
 example for
 NPCP, 5-6
 NRInet, 5-6
 TTY, 5-6
 upload
 definition, 5-7
 example file, 5-13
Converting 4000 series applications, 3-1
CPU (processor specifications), 1-6
Creating, an IFL SRAM card, 4-5

D

Delay, multipurpose utility, 2-5
Disk media contents
 Ethernet, 6220FL2B, 4-2
 non-Ethernet, 6220FL2A, 4-2
Display
 backlight, 1-3
 message, multipurpose DOS utility, 2-5
 orientation, 1-3
 touch screen, 1-3
Docks
 12-pin, connector location, 1-3
 charging battery in, 1-4
Documentation, related publications, 1-7
DOS
 firmware in flash, 1-4
 utilities
 CALIB.EXE, pen calibration, 2-12
 CRC32.EXE, file integrity verifica-
 tion, 2-4
 DELAY.EXE, multipurpose delay, 2-5
 drive mapping, 2-4
 get/set ROM DOS boot flags, 2-5
 RESET.EXE, reset system, 2-6
Download, RAMDFMT.CTL, 2-7
Download list file
 contents, 2-3
 create, IPLFMT, 2-3
 sample, 2-3
 upload and download files, 5-13
DRAM
 backup battery, data stored, 1-2
 maintaining power to memory, 1-4
 memory, 1-4
 sizes, 1-4
Drive letters, assign/unassign, DOS util-
 ity, 2-4
Drivers
 BGI support, 2-1
 communications, LAN, 2-8
 DOS
 calibration utility, CALIB.EXE, 2-12
 mouse (pen), 2-13
 power management, 2-15
 DOS utilities, 2-2
 PC Card services, 2-11
 printers
 IrDA printer, 2-8
 NPCP printer, 2-9
 PenPrint, 2-13
 scanner, tethered, 2-18
 screen emulation, 2-16
 upper memory driver, 2-20
Drives
 letters assigned automatically, 2-5
 supported for use, 2-5

E

Emulation, screen, 2-16
Environment, open systems, 1-5

Errors

- numeric responses, modem, **5-11**
- protocol errors
 - MININET, **5-9**
 - NPCP, **5-9**

Example files

- control file
 - NPCP, 5-6
 - NRInet, 5-6
 - NRUPLD.CTL, upload, 5-7
 - TTY, 5-6
- scanner application, 2-18

Executing, RAMDFMT, 2-7

F

- Features, listed, 1-1
- File integrity verification, DOS utility, 2-4
- Files. *See* Files Index
- Firmware, 1-6
- Flash
 - BIOS in, 1-4
 - BIOS storage, 1-7
 - DOS firmware in, 1-4
 - memory, 1-4
 - size of, 1-4
 - updating BIOS, 1-7

Flashing

- creating an IFL SRAM card, 4-5
- updating
 - from an application, 4-2
 - using INTERLNK, 4-3
 - with a 4 MB SRAM card, 4-5
 - with an ATA card, 4-4
 - with an IFL card, 4-5

Fonts

- file format, 2-16
- screen emulation, 2-16

I

- Include files, download include files, RAMDFMT.CTL, 2-7

InfraRed. *See* IrDA

Initial flash load (IFL)

- creating an IFL SRAM card, 4-5
- updating flash with an IFL card, 4-5

Initial program load (IPL), device driver,

- IPLFMT.EXE, 2-3

Installing, DOS drivers

- IrDA printer, 2-8
- NPCP printer, 2-9
- pen (mouse), 2-13
- pen calibration, 2-12
- scanner, tethered, 2-18

Interfaces, 4000 Series programming, 3-4

INTERLNK

- device driver, 5-1
- server, INTERSVR, 5-1
- updating flash, 4-3
- using, 5-1

INTERMEC resources, 2-7

- creating RAM drive, 2-7

Interrupts, 4000 Series video

- supported, **2-17**
- unsupported, 2-17

INTERSVR

- configuring, 4-3
- using, 5-1

Introduction, 1-1

IrDA, printing, 2-8

K

Keyboard, 1-4

- configurations, 1-4
- interrupts, 4000 Series emulation, 3-4
- redefinition, 2-2
- remapping, 1-4, 2-2
- suspend/resume key, 1-4

L

LCD screen, power managed, 1-6

Location

- connectors
 - 12-pin to dock, 1-3
 - 15-pin serial, 1-3
 - 4-pin RJ-11 phone jack, 1-3
- PC Card slots, 1-5

Log file, communications, 5-8

M

Main battery, 1-2

Master Mode, ROM DOS boot flags, 2-5

Memory

- adding more, 2-20
- CMOS, 1-4
- DRAM, 1-4
- flash ROM, 1-4
- PC Cards, 1-5
- types, 1-4
- upper memory provider, 2-20

MININET

- installing for NPCP, 5-3
- protocol errors, **5-9**

Modem, null modem cable, 5-1

Mouse (pen), driver, 2-13

N

NORAND Utilities

- purpose, 3-1
- TCOM sessions, 3-1

Norand Utilities

- COM= parameter, 5-5
- CONFIG= parameter, 5-5
- DATABITS= parameter, 5-5
- download request file, 5-12
- IPCLIENT= parameter, 5-5
- IPHOST= parameter, 5-5
- MODEMSELECT= parameter, 5-5
- MODEMSTYPE= parameter, 5-6
- NETMASK= parameter, 5-6
- NPCPHOST= parameter, 5-6
- PARITY= parameter, 5-6
- PHONE= parameter, 5-6
- PROTOCOL= parameter, 5-6
- purpose, 5-1
- ROUTER= parameter, 5-6
- session control file, 5-12
- SPEED= parameter, 5-6
- STOPBITS= parameter, 5-6

system setup requirements, 5-3

- TCOM sessions, 5-1
- TRIES= parameter, 5-6
- updating flash using INTERLNK, 4-3

Norand Utilities, PSROM0C.EXE

- installing MININET for NPCP, 5-3
- protocols, NPCP, 5-3
- RAM drive, create, 2-7

NPCP

Norand Utilities

- NPCPHOST= parameter, 5-6
- PROTOCOL= parameter, 5-6
- TRIES= parameter, 5-6
- printing, 2-9

NPCP network, 5-2

NRInet, 5-3

Norand Utilities

- IPCLIENT= parameter, 5-5
- IPHOST= parameter, 5-5
- NETMASK= parameter, 5-6
- PROTOCOL= parameter, 5-6
- ROUTER= parameter, 5-6
- TRIES= parameter, 5-6
- using PSROM0C version 2.xx, 5-4
- using PSROM0C version 3.xx, 5-4

Null modem cable, 5-1

O

ON key, 1-4

Open systems environment, 1-5

P

Pause, multipurpose delay utility, 2-5

PC Card

- how used, 1-5
- power managed, 1-6
- services, command line switches, 2-11
- slot location, 1-5
- support for SystemSoft, reserved address range, 2-7
- with Stackcr, booting from RAM drive, 2-7

PC Card services, 2-11

PCMCIA. *See* PC Card

Pen (mouse), driver, 2-13

Pen calibration, 2-12

Phone jack, RJ-11, 4-pin connector loca-

- tion, 1-3, 1-6

PL/N

- file descriptor for binary files, 5-14
- loading 4000API.EXE for application, 3-4

Pod, serial scanner, command line

- switches, 2-19

Ports

- 15-pin serial location, 1-6
- COM, 1-5
- BIOS interrupt, 4000 Series, 3-4

POST (Power-On Self-Test), 1-7

Power

- batteries, 1-2
- management, 1-6

Power management, 1-6, 2-15

- clock driver, 2-1
- configuration, 2-2

Power-On Self-Test (POST), 1-7

Printers, 1-6

4800, 6800, 2-9

IrDA, 2-8

NORAND printers (PenPrint), 2-13

NPCP, 2-9

Processor, 1-6

power managed, 1-6

Programming interfaces, 4000 Series, 3-4

Protocols, NPCP, 5-3

Publications, related documentation, 1-7

Purpose for this book, 1-1

R

RAM

drive

creating, 2-7

size versus total HHC memory, 2-7

maintaining power via batteries, 1-2

preserved by smart battery, in suspend mode, 1-2

Reflashing

creating an IFL SRAM card, 4-5

updating

from an application, 4-2

using INTERLNK, 4-3

with a 4 MB SRAM card, 4-5

with an ATA card, 4-4

with an IFL card, 4-5

Related publications, technical documentation, 1-7

Remapping, keyboard, 1-4, 2-2

Reset switch, 1-6

Reset system, DOS utility, 2-6

Resume (suspend/resume) ON key, remapping keyboard, 1-4, 2-2

RJ-11 phone jack, 4-pin, connector location, 1-3

ROM

flash memory, 1-4

get/set ROM DOS boot flags, DOS utility, 2-5

RS-232, data transfer, 1-6

S

Scanner, serial pod, command line switches, 2-19

Scanner, tethered, 1-6, 2-18

example application, 2-18

power managed, 1-6

Screen emulation, 2-16

Serial, 15-pin connector, location, 1-3, 1-6

Session control file

format, 5-12

header, 5-12

terminal identification, 5-12

Speed of processor, 1-6

Suspend/resume (ON key), 1-4, 2-2

Switch, reset, 1-6

Switches

command line, RAMDFMT.EXE, 2-7

installation, MININET.EXE, 5-3

System BIOS

in flash, 1-7

updating, 4-1

System memory

CMOS, 1-4

DRAM, 1-4

flash ROM, 1-4

PC Cards, 1-4

types, 1-4

System reset, DOS utility, 2-6

T

TCOM session overview, 5-12

Technical documentation, related publications, 1-7

Terminal identification, 5-12, 5-14

Tethered scanner, 1-6, 2-18

command line switches, 2-19

TFTP

communications, 2-7

Norand Utilities uses, 2-7

Touch screen, 1-3

Troubleshooting, main battery, 1-3

TTY

Norand Utilities

COM= parameter, 5-5

CONFIG= parameter, 5-5

DATABITS= parameter, 5-5

MODEMSELECT= parameter, 5-5

MODEMTYPE= parameter, 5-6

PARITY= parameter, 5-6

PHONE= parameter, 5-6

PROTOCOL= parameter, 5-6

SPEED= parameter, 5-6

STOPBITS= parameter, 5-6

TRIES= parameter, 5-6

supported by, 5-3

U

Updating BIOS, 1-7

Updating flash

from an application, 4-2

using INTERLNK, 4-3

with a 4 MB SRAM card, 4-5

with an ATA card, 4-4

Upload and download files, 5-13

Upper Memory Provider (UMP), 2-20

advantages, 2-20

command line switch, 2-20

Utilities

configuration, SC400CFG.COM, 2-2

drive mapping, MAPDRIVE.EXE, 2-4

file integrity verification, CRC32.EXE, 2-4

get/set ROM DOS boot flags,

MMBFLAG.COM, 2-5

multipurpose delay, DELAY.EXE, 2-5

pen calibration, CALIB.EXE, 2-12

reset system, RESET.EXE, 2-6

V

Verification utility, for integrity of toolkit files, 2-4

Video BIOS extensions, 4000 Series, 3-4

W

Wait, multipurpose delay, DOS utility, 2-5

Files Index

NOTE:

This index section is provided to assist you in locating descriptions for device drivers, applications, utilities, batch files, or other files within this publication.

NUMBERS

20A104-2.EXE, 4-1
20A104-3.EXE, 4-1
20A104-4.EXE, 4-1
230125-2.EXE, 4-1, 4-2, 4-3, 4-4, 4-6
230125-3.EXE, 4-1, 4-2, 4-5, 4-6
230125-4.EXE, 4-1, 4-2
230126-2.EXE, 4-2, 4-4, 4-6
230126-3.EXE, 4-2, 4-5, 4-6
4000API.EXE, 3-2, 3-3, 3-4
6220FF2A.104, 4-1
6220FF2A.INC, 4-1, 4-2, 4-3, 4-6
6220FF2B.INC, 4-2
6220FL2A, contents, 4-2
6220FL2A.104, 4-1
6220FL2A.IFL, 4-5, 4-6
6220FL2A.INC, 4-1, 4-2, 4-3
6220FL2A.TXT, 4-1
6220FL2B, contents, 4-2
6220FL2B.IFL, 4-6
6220FL2B.INC, 4-2

A

AUTOEXEC.BAT, 2-2, 3-2, 3-3, 5-3, 5-5

B

BGIDEMO.C, 2-2
BOOTP.EXE, 5-4

C

CALIB.EXE, 2-12
CLOCK.EXE, 2-1
COMMAND.COM, 3-2, 4-4
CONFIG.SYS, 2-1, 2-5, 2-8, 2-9, 2-18, 2-20, 3-2, 5-3
CPLNI.COM, 3-2
CRC32.EXE, 2-4
CUSTOM.C, 2-1

D

DD.EXE, 4-2, 4-5
DELAY.EXE, 2-5
DHCP.EXE, 5-4, 5-5

E

EMM386.EXE, 2-20
ETHDRV.EXE, 5-4

F

FL20A104.TXT, 4-1
FONTMAP.EXE, 2-16, 2-17, 3-3
FORMAT.COM, 4-5

G

GR12x16.FMT, 2-16

H

H2THRSCN.EXE, 2-18
H2UMPSYS, 2-20
HIMEM.SYS, 2-7, 2-20

I

INET.EXE, 5-4
INTERLNK.EXE, 4-3
INTERSVR.EXE, 5-1
IO.SYS, 3-2, 4-4
IPLFMT.EXE, 2-3, 2-4, 3-3

L

LAN9000.COM, 5-4
LSL.COM, 5-4

M

MAPDRIVE.EXE, 2-4
MININET.EXE, 2-8, 3-3, 5-3, 5-9
MMBFLAG.COM, 2-5
MOUSE.COM, 2-13
MSDOS.SYS, 4-4

N

N6220.BGI, 2-1, 2-12

N6220.H, 2-2
NCDIR.EXE, 5-2
NET.CFG, 5-2, 5-4
NGENMOD.SYS, 5-3
NORDOSPM.EXE, 2-1, 2-15, 2-18
NORIRDA.SYS, 2-8
NORMOD.SYS, 2-11, 5-3
NRTCMMERR.TBL, 5-3
NRTLOG.DAT, 5-1, 5-8
NRTTYM.TBL, 5-3
NRUPLD.CTL, 5-1, 5-7

O

ODIPKT.COM, 5-4

P

PC-DEXIO.BIN, 3-2
PC4800.SYS, 2-9, 2-10, 3-2
PCTCP.INI, 5-2, 5-4, 5-10
PENPRINT.EXE, 2-13
PSROB0C.EXE, 5-1, 5-3
PSROM0C.DAT, 5-1
PSROM0C.EXE, 5-1, 5-3, 5-4, 5-6, 5-8
 version 2.xx, 5-4
 version 3.xx, 5-4
PSROM0C.INI, 5-1

R

RAMDFMT.CTL, 2-7
RAMDFMT.EXE, 2-7
README.TXT, 4-1, 4-2
RELNOTES.TXT, 2-4, 4-1
RESET.EXE, 2-6
ROMINIT.BAT, 5-3
RS485ODI.COM, 5-4

S

SC400CFG.COM, 1-3, 1-4, 2-2
SC400CFG.INI, 2-2
SERVICES, 5-5
SUBST.COM, 2-4
SYS6.COM, 4-4

W

WATTCP.CFG, 5-4



Corporate Headquarters
6001 36th Avenue West
Everett, Washington 98203
U.S.A.

tel 425.348.2600

fax 425.355.9551

www.intermec.com

6220 Hand-Held Computer Technical Reference



P/N 977-054-010D