FMT1000 Series
TCP/IP Remote Boot
Setup Guide

70-16673-01
Revision A
October 1995

symbol®

SYMBOL TECHNOLOGIES KNOW IT NOW

Symbol Technologies, Inc.
116 Wilbur Place
Bohemia, N.Y. 11716

Symbol Support Center

For service information, warranty information, or technical assistance in the U.S.A., call:

SYMBOL SUPPORT CENTER
1-800-653-5350

If you purchased your Symbol product from a Symbol Business Partner, contact that Business Partner for service.

Canada

Mississauga, Ontario
Canadian Headquarters
(905) 629-7226

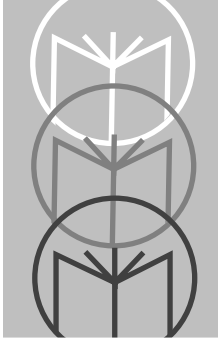
Europe

Wokingham, England
European Headquarters
734-771-222 (Inside UK)
011-44-734-771222 (Outside UK)

Asia

Singapore
Symbol Technologies Asia Inc.
337-6588 (Inside Singapore)
+65-337-6588 (Outside Singapore)

Contents



About This Guide

Introduction

This installation guide is intended to provide the information necessary to make the FMT1000 Series computers capable of booting on a TCP / IP network.

Chapter 1, *Boot Host Setup,* describes techniques for communicating with diskless workstations and explains how to build a bootable image on the file server.

Chapter 2, *Host Setup Options,* details steps for booting your computer.

Chapter 3, *Using PCTCP with an FMT1000 Series Computer,* lists the files needed to boot a DOS image from the server when running FTP Software's PCTCP package.

Chapter 4, *BIOS Setup,* takes you through the setup sequence.

Appendix A, *Boot Support Utilities,* lists programs that need to be run as part of boot preparation or during the boot process.

Refer to the *FMT1000 Series Programmer's Guide* (Symbol P / N 70-16674-01) for information on FMT1020 programming and functions. Refer also to the *FMT1000 Series Remote Boot Setup Guide* (Symbol P / N 70-16672-01) for other network-specific connection information.

Notation Conventions

Keys and key sequences are printed in <UPPERCASE> and enclosed in broken brackets. For example:

<ENTER>	Enter or return key.
<TAB>	Tab key.
<CTRL+Z>	Key combination of <CTRL> and <Z>, the PC equivalent of holding down the control key while pressing the <Z> key.
<CTRL>-<Z>	FMT1000 key sequence equivalent of the standard IBM PC <CTRL+Z> key combination.
<F6>	F6 function key.

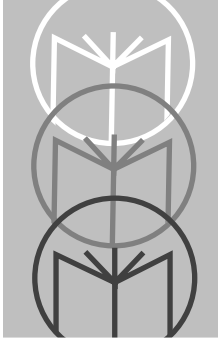
If you are using the FMT as your workstation, you must press and release the <CTRL> key, then press and release the <Z> key to achieve the same result as when using the standard IBM PC keyboard. You cannot press two or more keys at the same time in operating the FMT1000 computers.

Command lines to be entered from the keyboard will appear in **`BOLD UPPERCASE COURIER`**, e.g., **`FORMAT A:`**.

The prompt will appear in `regular courier`, e.g., `C:\>` or a simple colon (:).

Commands entered in DOS will be shown using a DOS style prompt with the correct drive and subdirectory, e.g., `F:\LOGIN>`.

Commands entered on the Novell NetWare server will be shown using the Novell prompt **:** (colon).



Chapter 1
Boot Host Setup

Overview

There are two major aspects to making FMT1000 series computers capable of booting on a Transmission Control Protocol / Internet Protocol (TCP / IP) network:

1. Building a DOS disk image that will be booted over the network.
2. Configuring the host so that it recognizes the FMT1000 series computer and permits it access to the system. The host must support either Reverse Address Resolution Protocol (RARP) or Bootstrap Protocol (BOOTP) daemons, as well as NFS.

Theory of Operation

Because FMT1000 series computers are diskless PCs, their use with a UNIX-based host on a TCP / IP network provides two common techniques for communicating with diskless workstations:

- RARP
- BOOTP.

The choice of protocol is made during setup, TCP / IP Boot Option, and may be saved in EEPROM thereafter. (See Chapter 4, Power-Up Setup Sequence, Step 11.)

RARP

When this boot option is selected, the FMT1000 series computer transmits an RARP request to find a server that can provide an IP address based on the computer's unique hardware address. This requires a host on the network with an active RARP daemon and appropriate routing tables established, usually in the files **/etc/hosts** and **/etc/ethers**.

Once the server has responded with the IP address, the FMT1000 series computer solicits a boot image file from the server. It does this by issuing a TFTP request for a filename based on hexadecimal representation of the FMT1000 series computer 's IP address located on the host TFTP directory. This directory is specified by the host in the **tftpd** initialization. (See Section 2, page 2-3, step 7.)

The boot image is copied to a RAM disk in the FMT1000 series computer , using conventional memory, and execution of the image begins as though the computer had been booted with a boot disk in drive A.

BOOTP

When this boot option is selected, the FMT1000 series computer broadcasts a BOOTP request to find BOOTP server on the network. Note that the BOOTP packet is transmitted on the internetwork and may, in fact, hop across routers to find a host. The RARP boot option will not hop across routers.

The BOOTP host contains a setup file that includes the FMT1000 series computer IP address, complete path, and file name for the boot image file. Once this file information is transmitted to the FMT1000 series computer , the specified file is downloaded via TFTP and further execution is the same as the RARP option described above.

Make sure that COMMAND.COM is in that disk directory and the COMSPEC environment variable is changed to point to the new location for COMMAND.COM.

Next, run a batch file from the file server. This should be a normal run (specify just the name to run) and not a CALL. This removes all open files from the RAM disk so it can be freed.

Once the boot image is loaded and executed, it is necessary to free the conventional memory consumed by the RAM disk. Use the utility RMVNTBT.COM. For more details, see the Appendix to this guide.

You are now ready to run your application program.

Building a Bootable Image

To build a bootable image you must create a DOS bootable floppy with all the files needed. First, copy the floppy image to a disk file by running the following command:

```
COPYDISK A: DISK.DOS
```

The first argument must be the letter designating the floppy drive where the bootable disk is located, and the second argument must be the output file name.

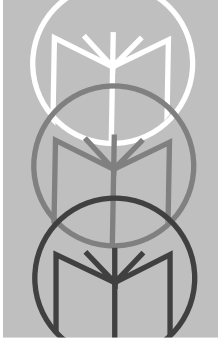
Once the image file has been created, issue the following command:

```
MKDOS DISK.DOS c8010162.dos
```

This creates a compressed disk image that will be loaded into a temporary RAM disk by the TCP/IP boot process. (Note: The name *c8010162.dos* should be changed to match the IP address of the specific FMT1000 series computer computer and should be entered in lower case.)

You must now copy the boot file to the **/tftpboot** directory on the file server or to the directory specified by the RARP daemon setup. If you are running a TCP/IP support package, you can use FTP to copy the file to the server. Again, note that on UNIX systems, file names are case-sensitive. Some systems require IMAGE mode to be set when performing a binary file transfer. Be sure that you have IMAGE mode set when you do the transfer.

The boot file name must be the hexadecimal value of the IP address for the remote FMT1000 series computer . For example, if the IP address is 200.1.1.98, the file name is c8010162.dos. The hexadecimal value of the IP address is used so that the file name will be eight characters or less to accommodate the file name limitations of some computer systems.



Chapter 2
Host Setup Options

SUN File Server

The following files must be edited on the SUN file server:

1. ***/etc/exports*** – To add access for the remotely mounted file system. See exportfs in the SUN documentation for configuration information.

   Sample Exports

   ```
   /
   /usr
   /export
   /export/swap
   ```

2. ***/etc/hosts*** – To add the host name and IP address for the FMT1000 Series computer to be available.

   Sample Hosts

   ```
   #
   # Sun Host Database
   #
   # If the NIS is running, this file is only consulted when booting
   #
   200.1.1.6  sparc
   200.1.1.99  fred
   200.1.1.3  bruce
   200.1.1.98 fmt
   ```

3. ***/etc/ethers*** – To add the mapping between the hardware address and the host name.

   Sample Ethers

   ```
   8:0:20:1:00:00  fred
   0:0:c0:a:a6:12  bruce
   0:C0:52:0:0:1a fmt
   ```

SCO UNIX System

Pre-Setup Checks and Configurations

Following are the pre-setup checks and configuration steps necessary to configure the SCO UNIX system for use with the FMT1000 Series computer.

1. Make sure the version of the TCP/IP runtime package you have is version 1.2.0 or later and includes the Reverse Address Resolution Protocol Daemon (RARPD).
   a. You must have the NFS Server Runtime Package, Version 1.2.0 or later to be able to mount the SCO disk system from the FMT1000 Series computer. Use the SWCONFIG command to confirm the existence of the SCO disk system on your runtime package.
   b. You must have a DOS-based TCP/IP and NFS Client package to access the SCO UNIX disk from DOS-based programs.
2. Edit the file /**etc/rc2.d/S85tcp** and remove the **#** from the front of the lines that comment out the RARPD from running. The lines will be a little after the section that begins: “Choice of routing daemons.”
3. Make sure that NFS is installed and running. A file called **/etc/rc2.d/S89nfs** starts the NFS support.
4. Edit the file **/etc/hosts** to add the IP address and name for the FMT1000 Series computers to your systems access list. The lines will look like this:

   ```
   192.1.1.20 fmt1

   192.1.1.21 fmt2
   ```

   Be sure to use an appropriate IP address for your site.

5. Edit the file **/etc/ethers** to add the hardware address and names for the FMT1000 Series computers so the RARPD program can tell the FMT1000 Series computer its IP address. The lines will look like this:

   ```
   00:c0:52:00:00:3f fmt1

   00:c0:52:00:00:40 fmt2
   ```

   These names must match the names used in the **/etc/hosts** file.

6. Edit the file **/etc/exports** to allow NFS access from the FMT1000 Series computer to the directories needed on the SCO UNIX host. The lines will look like this:

   ```
   /u/lpdos

   /u/lpapp
   ```

   Whatever is used here should match the entries in the NFS mount files on the FMT1000 Series computer. In the case of FTP Software it would be the **LPTCP.INI** file.

7. Edit the **/etc/inetd.conf** file to uncomment and / or modify the **tftp** line in this file to match as follows:

   ```
   tftp dgram udp wait root /etc/tftpd tftpd -s /tftpboot
   ```

   The **-s /tftpboot** option on the command line above tells tftpd to allow access to this directory only where the boot images for the FMT1000 Series computer PC will reside. The file names for the boots files are the hexadecimal representation of the FMT1000 Series computer's IP address with the extension *.dos*. This shortens the name to fit the eight-letter limit for DOS filenames. The boot name must be in lower case and must have the .dos extension. For example, if the IP address is 192.30.147.100, the boot file is named **c01e9364.dos**.

   Where $192_{10}=c0_{16}$

   and $30_{10}=1e_{16}$

   and $147_{10}=93_{16}$

   and $100_{10}=64_{16}$.

8. As necessary, create the **/tftpboot** and **/u/lpdos** directories and invoke **chmod** to change the permissions for global access.
9. If you are using the **LPTCP.INI** file that we are sending, edit the file and change all references of 192.30.147.42 to your host IP address. Also change any reference of **/u/lpdos** to the directory you will use for your DOS files.

   Be sure that all text files on your SCO UNIX system that will be used by the FMT1000 Series computer under DOS have both a Carriage Return and a Line Feed at the end of each line. It is expected that all the programs specific to this particular FMT1000 Series computer will be run from the host by means of a batch file using the hexadecimal IP address as the name for the batch file on the host, e.g., *c01e9364.bat*.

10. After making the changes described in Step 11, build a bootable image. Use **doscp** to copy the DOS image file to **/tftpboo**t and copy **rmvntbt.com** to / **u/lpdos**.

    Specifically, while in the **/u/lpdos** directory, use **doscp** to copy from diskette to hard drive:

    command.com

    rmvntbt.com

    Use **doscp -r** to copy:

    lptcp.ini

    c01e937b.bat

11. Reboot the SCO UNIX system in order for the changes to the system startup to take place.

HP9000 Running HP-UX BOOTP

Note: Preparation of the HP9000 to allow remote booting of FMT1000 Series computers follows many of the same steps required for installation of an HP700/RX X Station. We recommend that you read Section 8, Using BOOTP, in the HP System Administrator's Guide for the HP 700IRX.

Booting the FMT1000 Series computer

To boot the FMT1000 Series computer, follow these steps:

1. Note the hardware address of the FMT1000 Series computer as displayed on the 2x40 LCD during the power-on test or on the label on the rear of the unit. The six-digit address printed there is the last six digits of the unit's MAC address and is combined with the 6 digit OUI (00-c0-52) to determine the entire MAC address (00-c0-52-xx-xx-xx).
2. Obtain a unique IP address from your network administrator for the FMT1000 Series computer. Edit **/etc/hosts** and include the FMT1000 Series computer IP address and name.
3. Edit **/etc/exports** and include the directory or directories you wish FMT1000 Series computer to use for its files. Be sure to create these directories if they do not already exist, and use **chmod** to set desired access privileges. Note that these directories will correspond to "drive letters" in the nfs client setup (see **lptcp.ini** for examples).
4. Edit **/etc/bootptab** and include an entry for the FMT1000 Series computer(s), e.g.:

```
fmt1:\

ht=ether:\

ip=192.30.147.39:\

bf=/tftpboot/fmt.dos:\

ha=00c052xxxxxx:\

vm=rfc 1048
```

where **fmt1** and the ip address agree with those recorded in **/etc/hosts**, **ha** is the MAC address determined in Step 1 above, and **bf** is the filename (with path) of the boot image file that was created following the instructions of this manual.

5. Type **ps -e** to get a list of active processes and verify that the following daemons are running:

   tftpd, bootpd, lockd, nfsd.

   The first two daemons are required for the FMT1000 Series computer to boot; if necessary, edit **/etc/inetd.conf** and uncomment references to these daemons. The last two daemons are configured as part of **/etc/netnfsrc**, and this file may be edited if required to start nfs. If any of the daemons are not running, perform the required edits, restart the HP9000, and verify (using **ps -e**) that they started successfully.

6. If you have not already done so, prepare a boot image per the instructions on page 3. Copy this file (using FTP or NFS) into the **directory/ftlename** that was specified in step 6.
7. Reset the FMT1000 Series computer, enter Setup, and select TCP / IP as the boot type, BOOTP as the TCP / IP option, and ETHERNET-II as the protocol. Save this setup (using the F1 option), and the 2x40 LCD will now display a status screen indicating progress: the entire boot process should take only a few seconds to complete.

   Note: It may be desirable to connect an external CGA monitor during initial setup and application debugging.

RS6000 Running AIX

Note: Preparing a FMT1000 Series computer for remote booting under AIX is similar to the steps described in the AIX documentation for booting diskless SPARC stations. We recommend that you read these sections prior to setting up the FMT1000 Series computer.

To boot the FMT1000 Series computer, follow these steps:

1. Note the hardware address of the FMT1000 Series computer as displayed on the 2x40 LCD during the power-on test or on the label on the rear of the unit: the six-digit address printed there is the last six digits of the unit's MAC address and may be combined with the 6 digit OUI (00-c0-52) to determine the entire MAC address (00-c0-52-xx-xx-xx).
2. Obtain a unique IP address from your network administrator for the FMT1000 Series computer unit. Edit **/etc/hosts** and include the FMT1000 Series computer EP address and name.
3. Edit **/etc/exports** and include the directory or directories you wish FMT1000 Series computer to use for its files. Be sure to create these directories if they do not already exist, and use **chmod** to set desired access privileges. Note that these directories will correspond to "drive letters" in the nfs client setup (see ***lptcp.ini*** for examples).
4. Edit **/etc/bootptab** and include an entry for the FMT1000 Series computer(s) similar to the following:

```
fmt1:\
ip=192.30.147.39:\
hn:\
bf=/tftpboot/fmt.dos:\
ht=ether:\
ha=00c052xxxxxx
```

   where **fmt1** and the ip address agree with those recorded in **/etc/hosts**, **ha** is the MAC address determined in step 1 above, and bf is the filename (with path) of the boot image file that was created following the instructions in this manual.

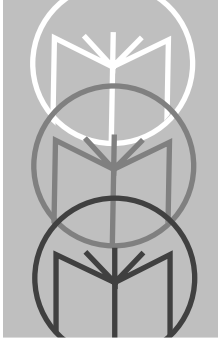
1. Type **ps -e** to get a list of active processes and verify that the following daemons are running:

   tftpd, bootpd, lockd, nfsd.

   The first two daemons are required for the FMT1000 Series computer to boot; while the last two daemons are required for the FMT1000 Series computer to have access to a remote disk drive. If any of the selected daemons are not running, invoke the diskless workstation management tool **smit -m** and follow through the menu selections for starting daemons on a server (**bootp and tftp**) and selections for communications applications on servers (NFS). When finished, run **ps -a** to reverify that the selected daemons have started. Note that restarting the RS6000 is not required.

2. If you have not already done so, prepare a boot image per the instructions in the beginning of this manual. Copy this file (using FTP or NFS) into the directory / filename that was specified in Step 6.
3. Reset the FMT1000 Series computer unit, enter setup, and select TCP / IP as the boot type, BOOTP as the TCP / IP option, and ETHERNET-II as the protocol. Save this setup (using the Fl option), and the 2x40 LCD will now display a status screen indicating progress. The entire boot process should take only a few seconds to complete.

   Note: It may be desirable to connect an external CGA monitor during initial setup and application debugging.



Chapter 3 Using PCTCP with an FMT1000 Series Computer

This section lists the files necessary to boot a DOS image from the server when running FTP Software's PCTCP package using the ODI (Novell) and NDIS (Microsoft) drivers.

This package allows you have a standard boot image that is shared by each FMT1000 Series computer in the network. The only unique files needed are the batch files to load the specific programs needed on a specific FMT1000 Series computer.

The files listed in the following sections are needed on the boot floppy. Copy them there after formatting the disk with the command:

```
FORMAT A: /S
```

Standard Boot Image Files with ODI Driver Support

Table 3-1. Files Provided by You

AUTOEXEC.BAT	The shared Batch startup file, calls the FMT1000 Series computer specific module.
COMMAND.COM	Command / Batch processor
CONFIG.SYS	Set the DOS startup parameters

Table 3-2. Files Provided with the FMT1000 Series Computer

definip.com	Define the three environment variables **ipaddr, snmask, and snbits**
lpnice.com	ODI Ethernet interface driver
lsl.com	Needed to support ODI driver
net.cfg	Needed by LSL and LPNICE
setcfg.com	Set the configuration in the specified file
lptcp.ini	TCP / IP initialization file
rmvntbt.com	Remove Bootup RAM DISK A: drive and return the memory

Table 3-3. Files Provided by FTP Software

ethdrv.exe	Generic TCP / IP suite
idmnt.exe	NFS file system mounter
idrive.exe	NFS files system support package
odipkt.com	ODI to Packet driver converter
idrive.msg	NFS files system support package

Sample CONFIG.SYS

```
FILES=40
BUFFERS=50
STACKS=9,256
LASTDRIVE=F
```

Sample AUTOEXEC.BAT

```
@echo off
REM ********************************************
REM * Generic startup file *
REM for an FMT mounting a NFS volume
REM using the FTP Software TCP/IP, NFS
REM package
REM ********************************************
REM
REM Set environment variable where the
REM temporary initialization file lives
SET PCTCP=A:\LPTCP.INI
REM
REM Setup environment variables for ipaddr,
REM snmask, and snbits
DEFINIP
REM
REM modify the initialization file to have
REM our address and sub-net mask
SETCFG LPTCP.INI %ipaddr% %snmask% %snbits%
REM
REM Load Link Support Layer
LSL
REM
REM Load FMT Ethernet driver
LPNICE
REM
REM Load the Novell ODI to Packet driver
REM translator
ODIPKT
```


```
REM
REM Load the TCP/IP support suite
ETHDRV
REM
REM Load the NFS support suite
IDRIVE
REM
REM Mount the network drive
IDMNT -a
REM
REM Set over to the new drive and path
C:
REM
REM Tell DOS where to get COMMAND.COM from
REM now on
SET COMSPEC=C:
REM
REM Set environment variable where the
REM permanent initialization file lives
SET PCTCP=C:\LPTCP.INI
REM
REM
REM Run our FMT specific batch file
|%ipaddr%
```

Sample Batch File for FMT 192.30.147.100

```
REM FILE NAME: c01e9364.bat|
REM Remove Bootup RAM DISK A: drive and return the memory
to us
RMVNTBT
```

Sample NET.CFG

```
LINK DRIVER LPNICE
FRAME ETHERNET_II
```

Files Needed in Standard Boot Image with NDIS Driver

The following files are needed, as well as a floppy formatted with DOS installed.

Table 3-4. Files Provided by You

AUTOEXEC.BAT	The shared Batch startup file, calls the FMT1000 Series computer specific module.
COMMAND.COM	Command / Batch processor
CONFIG.SYS	Set the DOS startup parameters

Table 3-5. Files Provided with the FMT1000 Computer

definip.com	Define the three environment variables **ipaddr, snmask, and snbits**
lp_ndis.sys	FMT1000 Series computer NDIS driver
protocol.ini	Define the interface for the NDIS protocol manager
setcfg.com	Set the configuration in the specified file
lptcp.ini	TCP / IP initialization file
rmvntbt.com	Remove Bootup RAM DISK A: drive and return the memory
netbind.exe	NDIS driver for network interface

Table 3-6. Files Provided by FTP Software

ethdrv.exe	Generic TCP / IP suite
idmnt.exe	NFS file system mounter
idrive.exe	NFS files system support package
dis_pkt.gup	NDIS to packet driver translation
protman.sys	Protocol manager
idrive.msg	NFS files system support package

Sample CONFIG.SYS

```
DEVICE=A:\PROTMAN.SYS /I:A:\
DEVICE=A:\DIS_PKT.GUP
DEVICE=A:\LP_NDIS.SYS
FILES=40
BUFFERS=50
STACKS=9,256
LASTDRIVE=F
```

Sample AUTOEXEC.BAT

```
@echo off
*******************************************
REM * Generic startup file *
REM for an FMT mounting a NFS volumeR
EM* on a SUN Sparc Station *
REM using the FTP Software TCP/IP, NFS
REM package.
REM *******************************************
REM
REM Setup network bindings for NDIS
NETBIND
REM
REM Set environment variable where the
REM temporary initialization file lives
SET PCTCP=A:\LPTCP.INI
REM
REM Setup environment variables for ipaddr,
REM snmask, and snbits
DEFINIP
REM
REM modify the initialization file to have
REM our address and sub-net mask
SETCFG LPTCP.INI %ipaddr% %snmask% %snbits%
REM
REM Load the TCP/IP support suite
ETHDRV
```

```
REM
REM Load the NFS support suite
IDRIVE

REM
REM Mount the network drive
IDMNT -a
REM
REM Set over to the new drive and path
C:
CD \
REM
REM Tell DOS where to get COMMAND.COM from
REM now on
SET COMSPEC=C:\COMMAND.COM
REM
REM Set environment variable where the
REM permanent initialization file lives
SET PCTCP=C:\LPTCP.INI
REM
REM Remove Bootup RAM DISK A: drive and
REM return the memory to us
%ipaddr%
```

Sample Batch File for FMT 192.30.147.100

```
REM FILE NAME: c01e9364.bat
RMVNTBT
REM
REM Run our FMT PC specific batch file
```

Sample Protocol.ini

```
; [PROTMAN]
; DRIVERNAME= PROTMAN$
; DYNAMIC= YES
```


```
; this is for FMT, none of these items
; should be changed

[LP_NIF]
DRIVERNAME = LPND$$$$
MAXTRANSMITS= 10
IOADDRESS = 0x300

INTERRUPT = 5
RAMSIZE(K)= 32
IOWORDSIZE= 8

;****************;
; Protocol       ;
;****************;

[PKTDRV]
DRIVERNAME= PKTDRV$
BINDINGS= LP_NIF
INTVEC= 0x66
; CHAINVEC= 0x66
```

Sample PCTCP Initialization File

```
[pctcp general]
host-name               = FMT
domain                  = bb.domain
user                    = FMT
full-name               = Woody
office                  =
office-phone            =
time-zone               = MST
time-zone-offset        = 420
use-old-init-scheme     = no
; pfile                 = %<string opt>

[pctcp kernel]
interface               = ifcust 0
```

```
serial-number           = xxxx-xxxx-xxxx
authentication-key      = xxxx-xxxx-xxxx
window = 1024
; use-emm               = %<boolean opt>
host-table =
; kernel-int            = %<string opt>
; ip-precedence         = %<string opt>
; ip-precedence-matching= %<string opt>
; ip-security           = %<string opt>
large-packets           = 5
small-packets           = 5

tcp-connections         = 6
udp-connections         = 6
[pctcp addresses]
domain-name-server      = 192.30.147.32
; time-server           = %<ip-addr opt>
; mail-relay            = %<ip-addr opt>
; cookie-server         = %<ip-addr opt>
; log-server            = %<ip-addr opt>
; imagen-print-server   = %<ip-addr opt>

[pctcp ip-security]
; basic-classification  = %<string opt>
; basic-authority       = %<string opt>
; matching              = %<string opt>
; extended              = %<string opt>
; ports                 = %<string opt>

[pctcp bootp]
; server-address        = %<ip-addr opt>
; backup-path           = %<string opt>

[pctcp tar]
; user                  = %<string opt>
; host                  = %<string opt>
```

```
; file                    = %<string opt>
; date-file               = %<string opt>

[pctcp host]
; umask                   = %<number opt>

[pctcp smtp]
; default-host            = %<string opt>


[pctcp vmail]
; client-section          = %<string opt>
; reply-to                = %<string opt>
; indent-reply            = %<string opt>
; fullname                = %<string opt>
; signature               = %<string opt>
; aliases                 = %<string opt>
ignore-fields             = Received Message-        Id
Sender Originating-Client Path
; new-mbox                = %<boolean opt>
; post                    = %<boolean opt>
highlight-fields          = Subject From:
; editor                  = %<string opt>
; printer                 = %<string opt>
; record                  = %<string opt>
; folder                  = %<string opt>
; ask-overwrite           = %<boolean opt>
; print-nonprintable      = %<boolean opt>


[pctcp pcmail]
; toplevel-directory      = %<string opt>
; user                    = %<string opt>
; password                = %<string opt>
; server                  = %<string opt>
; msglimit                = %<number opt>
; max-sync-descs          = %<number opt>
```

```
; log-level              = %<string opt>
; net-time-out           = %<number opt>
; repository-port        = %<number opt>

 [pctcp pop3]
; toplevel-directory     = %<string opt>
; user                   = %<string opt>
; client                 = %<string opt>
; server                 = %<string opt
; rpop                   = %<boolean opt>
; pop3-port              = %<number opt>
default-mbox            = default
; bboard                 = %<boolean opt>
; net-time-out           = %<number opt>
; xmit                   = %<boolean opt>
; max-sync-descs         = %<number opt>
; log-level              = %<string opt>

[pctcp pop2]
; toplevel-directory     = %<string opt>
; user                   = %<string opt>
; client                 = %<string opt>
; server                 = %<string opt>
; rpop                   = %<boolean opt>
; pop2-port              = %<number opt>
default-mbox            = main_mbox
; net-time-out           = %<number opt>
; max-sync-descs         = %<number opt>
; log-level              = %<string opt>

[pctcp nntp]
; toplevel-directory     = %<string opt>
; user                   = %<string opt>
; client                 = %<string opt>
; server                 = %<string opt>
; nntp-port              = %<number opt>
; post                   = %<boolean opt>
```


```
; net-time-out          = %<number opt>
; max-sync-descs        = %<number opt>
; log-level             = %<string opt>
expire-all              = 4 days
join-max-msgs           = 20

[pctcp screen]
; ftp-attr              = %<string opt>
; ftp-displ             = %<string opt>

[pctcp tn]
; ftpsrv                = %<boolean opt>
cmdline                 =  -?
; status                = %<boolean opt>
back-arrow-key = del
; screen-saver          = %<string opt>

[pctcp 3270]
; model-3/4             = %<number opt>
; yale                  = %<boolean opt>
; 3270-keymap           = %<string opt>
; ascii/ebcdic          = %<string opt>
; ftp-sfe-attr          = %<string opt>
; ftp-sfe-rv-attr       = %<string opt>

[pctcp vt]
; def-em-mode           = %<string opt>
wrap-line = on

[pctcp print]
; prog                  = %<string opt>
; spool                 = %<string opt>
; swap                  = %<string opt>
; printer               = %<string opt>
; hotkey                = <boolean opt>
; timeout               = %<number opt>
; onexit                = %<boolean opt>
```

```
[pctcp idrive]
cache-tmo= 5
default= hp9000a
lock-tmo= 5
; map-char = %<string opt>
; mapping = %<string opt>
; retry-x= %<number opt>
; stream= %<number opt>
; sym-del= %<boolean opt>
umask = 777
nis-server = 192.30.147.42

[pctcp idrive hp9000a]
host = 192.30.147.42
path = /u/lpdos
drive = c:
sec-key = pcnfs
sec-arg = nobody
lock = Y
mtu = 1000
reads = 512
timeout = 5
writes = 512
umask = 777

[pctcp kerberos]
directory = c:\etc

 [pctcp lpr]
server = 192.30.147.42
; printer = %<string opt>
; class = %<string opt>
; banner = %<boolean opt>

[pctcp comscrpt freds-host]
; dialup = %<string opt>
; hangup = %<string opt>
```


```
[pctcp connect freds-host]

local-ip-address        =  0.0.0.0
remote-ip-address       =  0.0.0.0

[pctcp time]
; dst-begins            = %<number opt>
; dst-ends              = %<number opt>

[pctcp ifcust 0]

ip-address              = ?aaaaaaaaaaaaa
subnet-mask             = ?mmmmmmmmmmmmmm
```

Note: On these pages, on lines beginning with *domain-name-server*, *nis-server*, *host*, and *server*, the address should be changed from *192.30.147.42* to the IP address of the host computer.

Chapter 4
BIOS Setup

Once the FMT1000 Series computer is booted, the Ethernet address and BIOS revision level appear on the first line of the 2-line by 40-character display. The second line displays **"[ENTER]=setup"** even if setup has been disabled. Figure A-1 shows a sample after-boot display.

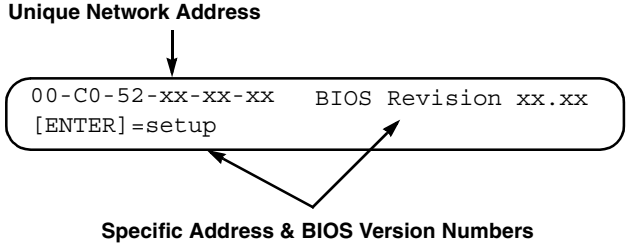


Figure D-1. First Screen Display after Bootup

A delay of a few seconds occurs before the system finishes the boot process. To enter into the setup process, the <ENTER> key must be pressed while the address and revision are displayed.

Figure A-2 shows a sample setup screen. Line 1 of the display indicates which setup screen and option are currently selected. Line 2 shows the currently saved options. The <ENTER> key accepts a setting and advances to the next setup screen.

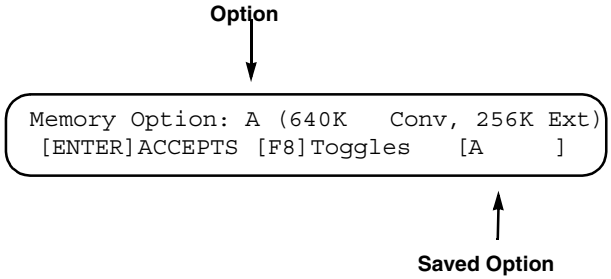


Figure D-2. Memory Options Screen Using <F8> Key To Toggle Between Options

Pressing the following keys moves you through the setup screens and their option lists:

<F5>	Moves you backward through setup screens without change.
<F6>	Moves you forward through setup screens without change.
<F7>	Moves you backward through option list.
<F8>	Moves you forward through option list.
<ENTER>	Accepts option setting and advances you to next setup screen.

BIOS Setup Screens

After pressing <ENTER> to get into the setup mode, you are presented with numerous setup screens to select options affecting display operation, memory configuration, boot type, etc. Details about the setup screens may be found on the network driver disk in a file named “Readme.txt”. After all setup screens have been viewed and choices made, the save options screen will appear as shown in Figure A-3.

Figure D-3. The Save Options Screen

Setup is now complete. Several options are available:

Press <F1>: Exit and save changes to EEPROM. (Changes appear every time FMT1000 Series computer boots up.)

Press <F2>: Abandon changes. (Setup returns immediately to status in effect prior to entering setup.)

Press <F8> Exit without saving changes to EEPROM. (Setup returns to status in effect prior to entering setup but remains in effect for current work session.)

A confirming message is displayed, as shown in Figure A-4, and the computer proceeds with boot.

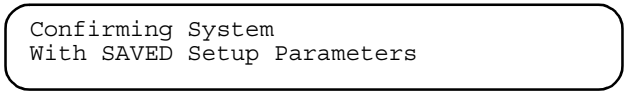
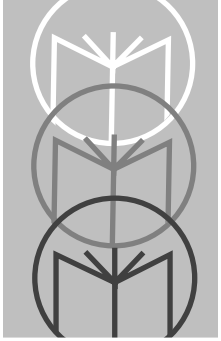


Figure D-4. Confirmation Message



Appendix A
Boot Support Utilities

These utilities support remote booting of FMT1000 Series computers. To this end they do minimal error-checking to keep the program size small. It is assumed that these programs will be run as part of boot preparation or in the boot process in a well defined environment. Any error messages that can be output are listed at the end of each program description with an explanation of possible cause.

DEFINIP.COM

This program obtains the IP address and sub-net mask values from the bootstrap after a DOS image has been loaded, and defines the following environment variables:

- **IPADDR** – Eight-character hex representation of the FMT1000 Series computer's IP address.
- **SNMASK** – Eight-character hex representation of the sub-net mask for the network if it is known.
- **SNBITS** – Two-character hex representation of the number of trailing 0 bits in the sub-net mask. This value is needed as a parameter for some PC/TCP commands.

Possible DEFINIP messages are as follows:

```
? DEFINIP: Cannot get network data
```

This is usually the result of not running this on a remotely booted FMT1000 Series computer.

```
? DEFINIP: Environment area is too small
```

This is the result of not having enough environment space allocated in the bootable image. Modify your **CONFIG.SYS** file and give **COMMAND.COM** more environment space.

SETCFG.COM

This program modifies a PC / TCP (or other) configuration file to include the IP address or sub-net mask values as obtained from the bootstrap. It is invoked as follows:

```
SETCFG file.ext aaaaaaaa mmmmmmmm bb
```

where:

file.ext = Name of file to modify. (Usually **lptcp.ini**)

aaaaaaaa = IP address (must be exactly 8 hex digits)

mmmmmmmm = Sub-net mask (must be exactly 8 hex digits)

bb = Sub-net mask trailing bits value (must be exactly 2 hex digits)

This command is intended to be invoked from AUTOEXEC.BAT after DEFINIP, using the environment strings set by DEFINIP as arguments:

```
SETCFG file.ext %IPADDR% %SNMASK% %SNBITS%
```

The file to be modified must contain special fields indicating where data is to be placed as follows:

?aaaaaaaaaaaaaa = Replace with IP address

?mmmmmmmmmmmmmm = Replace with sub-net mask value

?bb = Replace with sub-net mask trailing bits value

The **?aaaaaaaaaaaaaa** field must contain exactly 14 **a**'s after the question mark.The ?**mmmmmmmmmmmmmm** field must contain exactly 14 **m**'s after the question mark. The **?bb** field must contain exactly 2 **b**'s after the question mark. The value inserted in the file is the dotted decimal format representation of the IP address and the sub-net mask value and the decimal representation of the sub-net mask trailing bits value.

Possible messages from SETCFG are as follows:

```
? SET CFG: Illegal argument value

? SETCFG: Command line too short

? SETCFG: Cannot open configuration file

? SETCFG: Error reading configuration file

? SETCFG: Error writing configuration file

% SETCFG: Complete
```

RMVNTBT.COM

This program removes the boot RAM-disk from memory. It is normally executed from a batch file read from the file server after the DOS program area has been mounted as an NFS disk.

RMVNTBT and the command to redefine the COMSPEC environment string to point to the NFS disk where DOS programs are run from:

```
%IPADDR%
```

The first line of the specific batch file, with the hexadecimal IP address, which would start with the following command:

```
RMVNTBT
```

This would be followed by whatever additional machine-specific startup commands are needed. Once the RMVNTBT command completes, the IP bootstrap is completely disabled and all memory used is returned to DOS. If there is a local drive A, it can also be accessed at this point. (It cannot be accessed before the IP bootstrap is removed, because the bootstrap operates by emulating drive A.)

COPYDISK.EXE

This is a DOS utility which copies a floppy image to a disk file.

%% COPYDISK: Copying xxxx bytes

This message lets you know how much disk space will be consumed by this file.

% COPYDISK: Complete

? COPYDISK: Error yyyyyy reading floppy disk

? COPYDISK: Error writing output file

? COPYDISK: Not enough memory

MKDOS.EXE

This is a DOS utility which creates an IP bootable image from an image disk file.

To create an IP bootable image under DOS, first create a bootable floppy containing all desired files. The floppy can be any size, but should be packed. Excess space above the last file on the disk does not affect the size of the loadable image, but extra space between files does. A packed disk can be created by initializing a new disk and copying all files to it without deleting any files or by using a disk compression utility on an existing disk.

The floppy must first be copied to a disk file as follows:

```
COPYDISK A: DISK.DOS
```

The first argument must be a floppy drive letter, and the second argument must be the output file name.

Once the image file has been created, issue the command:

```
MKDOS DISK.DOS BOOT.DOS
```

The first argument is the name of an image file created by **COPYDISK** and the second argument is the name for the bootable image to be created. When this command completes, the bootable image created can be copied to the server using NFS or FTP or by floppy (if the server can read DOS disks). If FTP is used, IMAGE transfer mode must be used. If MKDOS has been ported to the server, the bootable image can be created completely on the server, assuming that the server can read the bootable DOS disk and create a disk file which is an image copy of it.

% MKDOS: Image size is xxxx bytes

```
? MKDOS: Unexpected end of file on input from "File name"

? MKDOS: Not enough memory
```