

# MORE ON UNIX

## Overview

This file provides additional information for Unix environments. It should be used if you cannot find what you need in the M202Plus product manual.

Within this file, you will find information about:

- M202Plus configuration through a manual ARP method - *Using ARP* on page 2.
- M202Plus configuration using RARP - *Using RARP* on page 4.
- M202Plus configuration using BOOTP - *Using BOOTP* on page 5.
- communicating between different subnets - *Communicating Across Routers* on page 6.
- a manually configured print setup under System V Unix - *Manual System V Host Setup* on page 7.
- a manually configured LPR/LPD under BSD Unix - *Manual LPR/LPD Host Setup* on page 10.
- AIX print setups - *Printing from AIX* on page 12.
- using FTP to print - *Printing with FTP* on page 12.
- printing directly to TCP ports - *Direct Socket Printing* on page 14.
- using Unix devices to print - *Device Printing - NPD* on page 14.

## M202Plus Configuration

As mentioned in the M202Plus product manual, there are mandatory TCP/IP settings (i.e. an IP address and subnet mask) needed before the print server can be detected on the network. There are also some additional settings that you may want to configure but these are optional. This section offers alternative methods for configuring your M202Plus within a Unix environment and mentions some of the more common optional settings available.

### Using ARP

To configure the M202Plus with its TCP/IP settings using a manual “**arp**” command, you will need to:

- 1 Log on to a Unix station as superuser or root. This station must be located on the *same* subnet as the M202Plus since ARP is limited to subnets.
- 2 Find the Ethernet address for the M202Plus on the bottom of the device. It must be entered as part of this procedure.
- 3 Use the “**arp**” command to add an entry into the Unix station’s ARP table for this M202Plus. This is the most common syntax for this command:

Syntax:

```
arp -s ipaddress ethernetaddress
```

Example for Microsoft TCP stacks:

```
arp -s 192.75.11.9 00:80:72:07:00:60
```

This example specifies an M202Plus using IP address **192.75.11.9** and Ethernet address **00:80:72:07:00:60**.

- 4 Check to see if the ARP entry took properly.

```
arp -a
```

You should see an entry in the listed ARP table with

the IP address and Ethernet address specified in Step 3.

- 5 Try to “ping” this IP address to see if the M202Plus can be seen on your network.

At this point, you should be able to communicate with the M202Plus from your *local* Unix station. This means the print server knows about an IP address and subnet mask to abide by and has these settings in its *current* memory. However, if the M202Plus is power cycled, these settings will disappear unless you store them into Flash. To do this, you will need to:

- 1 Load a Web browser on your Unix station and direct it to the URL “`http://M202PlusIPaddress/networkConf.html`” (e.g. “`http://192.75.11.9/networkConf.html`”).

**Note:** If prompted for a “User ID” and password first, type in “root” for the ID and press ENTER at the password prompt since there’s no password by default.

- 2 At the “Network Configuration” HTML form that displays, click in the field below the “IP Address” heading and type in the IP address for the M202Plus.
- 3 Under the “Subnet Mask” heading, enter the M202Plus’s subnet mask.

**Note:** If you would like to communicate with the M202Plus from across routers, you will need to fill in an entry within the “Routing” section. Please see *Communicating Across Routers* on page 6 for further details.

- 4 Click on the SUBMIT button when done and physically repower the print server to make the new settings take effect

## Using RARP

RARP allows you to dynamically assign an IP address to the M202Plus upon bootup. To configure the M202Plus with its TCP/IP settings using RARP, you will need to:

- 1 On your RARP server, make an entry in the `/etc/ethers` file for the M202Plus.

Syntax:

```
ethernetaddress M202PlusIPaddress
```

Example:

```
00:80:72:07:00:60 192.75.11.9
```

- 2 Start the RARP daemon on the RARP server if it isn't running already. If it is running, send a HUP signal to it with the "`kill`" command:

Syntax:

```
kill -HUP pid
```

where *pid* is the process ID of the RARP daemon.

- 3 Power on the M202Plus so that it immediately sends out a RARP request. Wait one minute to allow the address assignment process to complete. You should see the STAT LED on the front of the M202Plus slow down indicating it knows about an IP address to use.
- 4 Try to "`ping`" the M202Plus's IP address from a local Unix station to see if it can be seen on your network.

**Note:** You will only be able to communicate with the M202Plus from a network station on the *same* subnet as the print server. The M202Plus needs to be told about a default router/gateway so any packets that come in from a different subnet can be returned via this gateway. Please see *Communicating Across Routers* on page 6 to set this up.

## Using BOOTP

BOOTP allows you to dynamically assign an IP address to the M202Plus upon bootup. In addition, the BOOTP server can provide additional details like a default router/gateway address. To configure the M202Plus with its TCP/IP settings using BOOTP, you will need to:

- 1 On your BOOTP server, make an entry in the `/etc/bootptab` file for the M202Plus.

Syntax:

```
M202PlusIPname: \
  :sm=netmask: \
  :hd=homedirectory: \
  :bf=null: \
  :gw=defaultgateway: \
  :ht=ethernet: \
  :ha=ethernetaddress: \
  :ip=ipaddress: \
  :ts=timeserver: \
  :to=timezone:
```

Example:

```
spike: \
  :sm=255.255.255.0: \
  :hd=/usr/null: \
  :bf=null: \
  :gw=192.75.11.1: \
  :ht=ethernet: \
  :ha=008072070060: \
  :ip=192.75.11.9: \
  :ts=192.75.11.7: \
  :to=25200:
```

- 2 Turn the M202Plus on so that it immediately sends out a BOOTP request. Wait one minute to allow the IP address assignment process to complete. You should see the STAT LED on the front of the M202Plus slow down indicating it knows about an IP address to use.

- 3 Try to “ping” the M202Plus’s IP address from a network station to see if it can be seen on your network.

**Note:** You will be able to communicate with the M202Plus from network stations on different subnets if the print server has been told about its default router/gateway in the BOOTP response from the BOOTP server.

## Communicating Across Routers

Since Unix environments rely on TCP/IP to communicate with the M202Plus, crossing routers becomes an issue.

After following one of the M202Plus configuration methods mentioned within the product manual and within this file (besides the BOOTP method), you will most likely only be able to communicate with the print server from the *same* subnet. This means any hosts across a router will *not* be able to see your M202Plus. For this to happen, it’s best to store a default router/gateway within the print server so that any packets destined for another subnet get forwarded to this router automatically. The router (or series of routers) can then take over ensuring the packets get to their final destination on another subnet on your network.

To configure a default router/gateway within the M202Plus, you will need to:

- 1 Load a Web browser on your Unix station and direct it to the URL “`http://M202PlusIPaddress/networkConf.html`” (e.g. “`http://192.75.11.9/networkConf.html`”).

**Note:** If prompted for a “User ID” and password first, type in “root” for the ID and press ENTER at the password prompt since there’s no password by default.

- 2 At the “Network Configuration” HTML form that displays, click in the first field below the “Gateway” head-

ing and type in the IP address of the default router/gateway for the M202Plus's subnet.

- 3 Click on the SUBMIT button when done and physically repower the print server to make the new setting take effect.

## Host Configuration

The M202Plus product manual covers a Unix setup using Microplex's installation script, Ezsetup. However, there may be times when you'd prefer or have to perform the manual setup steps for a new printer definition. There may also be times when you require a less common print solution such as direct socket capabilities. This section covers these additional setups for you.

### Manual System V Host Setup

This setup involves any Unix operating system utilizing System V print spooling (e.g. SCO, Solaris, HP/UX, DG/UX, etc.). This means an interface file is used for each defined printer and in the case of any System V print setup involving an M202Plus, the RSHD protocol is used to transfer data from the host to the print server.

To manually configure a new System V printer on a Unix station, you will need to:

- 1 Create a dummy device file that acts as a locking mechanism between contending print jobs.

Syntax:

```
touch /dev/M202Plusname.portname
```

where *M202Plusname* is the host or IP name of the print server and *portname* is one of its four I/O ports.

Example:

```
touch /dev/spike.prn1
```

- 2 Change permissions on this file to “777”.

Example:

```
chmod 777 /dev/spike.prn1
```

- 3 Copy the interface script, **rawsysv.if**, to the model directory (i.e. usually /usr/spool/lp/model). This script comes on the Microplex diskette labeled “Unix Host Software” within the **npdist/hostsrc** directory.

Example:

```
cp npdist/hostsrc/rawsysv.if /usr/spool/lp/model
```

- 4 Change permissions on this interface script to “777”.

Example:

```
chmod 777 /usr/spool/lp/model/rawsysv.if
```

- 5 Open this interface script for editing. Look for these variables several lines down from the top and define accordingly:

COMM_PROG	Communication program used to transfer data to the M202Plus (i.e. <b>rsh</b> , <b>rcmd</b> , <b>remsh</b> depending on what your Unix system uses for the remote command.)
UNIT_NAME	Host or IP name of the M202Plus.
UNIT_PORT	Leave blank.
LP_DEST	Pre-defined destination/queue on the M202Plus telling the print server which I/O port to send the print jobs to. This will usually be “ <b>d1prn1</b> ” for the PRN1 port, “ <b>d2prn2</b> ” for the PRN2 port, “ <b>d3com1</b> ” for the COM1 port, and “ <b>d4com2</b> ” for the COM2 port.

Example:

```
COMM_PROG=rsh
UNIT_NAME=spike
UNIT_PORT=
LP_DEST=d1prn1
```

This defines a new printer using “**rsh**” to send print jobs to the destination, **d1prn1**, on an M202Plus named **spike**.

- 6 Save this interface file once the variables have been defined.
- 7 Create this new printer on your Unix host using the “**lpadmin**” command.

Syntax:

```
/usr/lib/lpadmin -pprintername
-mscriptname -vdummydevicefile
```

where *printername* is any name you’d like to give to this new printer, *scriptname* is the name of the interface file edited in the last few steps, and *dummydevicefile* is the file created in Step 1.

Example:

```
/usr/lib/lpadmin -plaser -mrawsysv.if
-v/dev/spike.prn1
```

- 8 Enable this new printer.

Example:

```
/usr/bin/enable laser
```

- 9 Tell this new printer to accept print job requests.

Example:

```
/usr/lib/accept laser
```

You now have a new printer which relies on RSHD and an interface script to print to an M202Plus printer. Use the fol-

following command to test printing:

Syntax:

```
lp -dprintername filename
```

Example:

```
lp -dlaser /etc/hosts
```

## Manual LPR/LPD Host Setup

The M202Plus also supports Unix operating systems which follow a BSD print system (e.g. SunOS 4.1.x). This means a printcap file is used to define all printers available on the host and interface scripts are not commonly used. Instead, LPR/LPD printers are defined.

To manually configure an LPR/LPD printer on a BSD Unix system, you will need to:

- 1 Define an entry for this new printer within the printcap file, /etc/printcap.

Syntax:

```
printername:
:sh:pw#80:mx#0:\
:rm=M202Plusname:\
:rp=M202Plusdestination:\
:lp=:\
:sd=/usr/spool/lpd/printername:\
:lf=/usr/spool/lpd/printername/log:\
:af=/usr/spool/lpd/printername/acct:
```

where *printername* is any name you'd like to give to this new printer, *M202Plusname* is the host or IP name of the print server, and *M202Plusdestination* is one of the pre-defined destinations/queues on the device.

Example:

```
laser:
:sh:pw#80:mx#0:\
:rm=spike:\
:rp=d1prn1:\
:lp=:\
:sd=/usr/spool/lpd/laser:\
:lf=/usr/spool/lpd/laser/log:\
:af=/usr/spool/lpd/laser/acct:
```

- 2 Create the spool directory specified by the “sd=” entry in the printcap entry for this printer.

Example:

```
touch /usr/spool/lpd/laser
```

- 3 Create the log and account files specified by the “lf=” and “af=” entries in the printcap entry for this printer.

Example:

```
touch /usr/spool/lpd/laser/log
touch /usr/spool/lpd/laser/acct
```

- 4 Change permissions and ownership on the spool directory and everything within it.

Example:

```
chmod -R g+rwX,o+rX /usr/spool/lpd/laser
chmod -R daemon.daemon /usr/spool/lpd/
laser
```

- 5 Start the printer daemon for this new printer.

Example:

```
lpc start laser
```

You now have a new printer which relies on LPR/LPD to print to an M202Plus printer. Use the following command to test printing:

Syntax:

```
lpr -Pprintername filename
```

Example:

```
lpr -Plaser /etc/hosts
```

## Printing from AIX

Version 4.0 and higher of AIX offers LPR/LPD support for network printing. This means to set up a new printer under this operating system, you only need to define these two parameters:

- 1 **M202Plus IP address** - the IP address or host name of the print server you want to print to.
- 2 **M202Plus destination/queue** - a pre-defined name on the print server telling the device which I/O port to send the jobs to. You will most likely use “**d1prn1**” for the PRN1 port, “**d2prn2**” for the PRN2 port, “**d3com1**” for the COM1 port, and “**d4com2**” for the COM2 port.

## Printing with FTP

Print jobs can also be sent to the M202Plus using the File Transfer Protocol (FTP). To do this, you will need to:

- 1 On your network station, change to the directory containing the file you want to print.
- 2 Open an FTP session with the M202Plus.

Syntax:

```
ftp M202PlusIPAddress
```

- 3 Log in as “**root**”.
- 4 At the password prompt, just press ENTER since there is no password by default.
- 5 Change to the **dest** directory.

Syntax:

**cd dest**

- 6 List the contents of this directory using the “**dir**” command and determine which destination/queue you’d like to send the print job through. Most likely you will select “**d1prn1**” for the PRN1 port, “**d2prn2**” for the PRN2 port, “**d3com1**” for the COM1 port, or “**d4com2**” for the COM2 port.
- 7 Change to this destination directory.

Example:

**cd d1prn1**

- 8 Change the mode to correspond with the type of file you are going to print. By default, ASCII is set for text-only files but if you need to print both text and graphics, select binary using the “**bin**” command.
- 9 Place the file to be printed within this directory on the M202Plus so it can be spooled and printed.

Syntax:

**put filename**

- 10 Close the FTP session unless you have more jobs to print.

Syntax:

**quit**

## Direct Socket Printing

Within TCP/IP environments, there are some setups which require the host to print directly to a TCP port number on the M202Plus bypassing all of the higher level destination/queue names. If you run into a setup which requires this, you will need to define two parameters for a new print setup:

- 1 **M202Plus IP address** - the IP address or host name of the print server you want to print to.
- 2 **M202Plus TCP port number** - a pre-defined number for each of the four I/O ports on the print server. The choices are **4000** for COM1, **4200** for COM2, **4400** for PRN1, and **4600** for PRN2 depending upon which port your printer is attached to.

Though the host will be sending directly to a TCP port number on the M202Plus, the port numbers are still mapped to destinations/queues in case there is some type of special processing or logging to be done to the job. However, the destination name doesn't come into play within this type of printer definition. Just the two parameters mentioned above need to be defined for printing to work.

## Device Printing - NPD

When printing from Unix, there is sometimes a need to send data directly to a device on the Unix station. The M202Plus will work with these print setups as well using proprietary software called "NPD".

NPD is a pseudo-tty daemon that continuously runs on the Unix station watching over a pre-defined device. Anytime data goes to this device, NPD grabs it and passes it over the network to the M202Plus for printing.

## Finding NPD

NPD comes on the Microplex diskette labeled "Unix Host Software". By default, binaries are provided for SunOS 4.1.x and for System V derivatives like Solaris and SCO.

These can be found within the **npdist/bin** path. Source code and a makefile for compilation are also provided within **npdist/hostsrc**.

### Starting NPD

Starting the NPD daemon is all that is needed to utilize NPD. The syntax is:

```
npd devicefile M202Plusname portname
```

where *devicefile* is the name of a pseudo-tty's slave device acquired by NPD, *M202Plusname* is the host or IP name of the print server, and *portname* is one of the four I/O ports on the device. (e.g. prn1, prn2, com1, com2).

**Note:** The file specified by *devicefile* cannot already exist on your Unix station or NPD will fail to start.

Example:

```
npd /dev/spike.prn1 spike prn1
```

Starts an NPD daemon to watch over the device file, **/dev/spike.prn1**, looking for data to send to the **PRN1** port off of the M202Plus named **spike**.

### Testing NPD

Once NPD has been started, it's a good idea to test and see if it is working properly. The best test method is to send data directly to the device once you know the printer attached to the M202Plus is ready to go.

Syntax:

```
cat filename > devicefile
```

Example:

```
cat /etc/hosts > /dev/spike.prn1
```

## Stopping NPD

Whenever an NPD daemon starts, it makes a log entry in the `/usr/adm/npdlog` file. The contents of this file are:

```
pid M202Plusname portname masterfile
slavefile devicefile
```

where:

<i>pid</i>	Process ID of the NPD daemon.
<i>M202Plusname</i>	Host or IP name of the M202Plus.
<i>portname</i>	One of the four I/O ports on the print server.
<i>masterfile</i>	Master device used by NPD.
<i>slavefile</i>	Slave device used by NPD.
<i>devicefile</i>	Name linked to <i>slavefile</i> by NPD.

By acquiring the process ID of the NPD daemon, you can kill the process using this command syntax:

```
kill -TERM pid
```

Example:

```
kill -TERM 153
```