



APPLICATION NOTE 612

Getting Started with the TINIm400 (DS80C400) Verification Module

Abstract: The purpose of this application note is to guide users through the initial setup process for the TINIm400 Verification Module. Following the directions provided, a user should be able establish communications with the verification module and load a simple example application. A list of required hardware and software components is included.

Also see: ["Getting Started with TINI"](#)

TINIm400 Verification Module Features

- Hosts the TINI® Runtime Environment in validated hardware design
- DS80C400 hardware reference design
- Three serial ports
- Integrated 1-Wire® net master
- CAN 2.0B port
- Real-Time Clock for time stamping
- Direct connection to 10/100 Base-T Ethernet
- 1MB flash ROM
- 1MB NV SRAM

Requirements

The minimum hardware and software requirements are:

- TINI Hardware Requirements
 - TINIm400 Verification Module or equivalent
 - TINI400 Sockets Board or equivalent
- TINI Software Requirements
 - [TINI Software Development Kit](#) Version 1.1 or later
 - [JDK™ \(Java Development Kit\)](#) - from Sun Microsystems or equivalent¹
 - [Java Communications API](#) - from Sun Microsystems or <http://www.rxtx.org>
- Development System Requirements
 - Operating system meeting requirements of the JDK from Sun Microsystems
 - 1 RS232 (COM) port - 115200 baud recommended
 - [RS232C serial cable DB9 male to DB9 female](#) - Radio Shack Catalog Number 26-117 or equivalent
 - [Crossover Ethernet cable](#) - Radio Shack Catalog Number 950-0368 or equivalent for direct connection to a host computer, a straight through cable should be used for connecting to a router or hub
 - [5V DC Power Supply](#) capable of supplying 150mA
 - Radio Shack Catalog Number 900-2740 or equivalent (Center Positive)

Important Note

The DSTINIs400 and DSTINI400 incorporate several LEDs. The three LEDs on the DSTINIs400 are attached to the PHY and indicate the speed, link, and receive status of the network connection. The LED on the DSTINI400 is controlled by the application software, and should be ignored for debug purposes.

Quick Start

1. Prior to beginning, verify power is not connected to the TINIs400 sockets board.
2. Insert the TINIm400 Verification Module into the 144-pin connector on the TINIs400 sockets board.
3. Verify the jumper labeled DTR Reset (J14) is placed on the sockets board.
4. Attach an RS232 serial cable to the connector labeled "Loader - Serial 0" on the TINIm400 sockets board as shown in **Figure 1**.
5. Connect the serial cable to a serial port on your PC.
6. Next, attach the crossover Ethernet cable between the PC and the TINIs400 sockets board.
NOTE: If the TINI sockets board is not being connected directly to a network interface card, but instead to a separate network port on a hub or switch, a straight through Ethernet patch cable should be used.
7. Connect the power adapter to your TINIs400 sockets board as shown in Figure 1. A center positive 5V DC regulated power supply must be used with the TINIs400 sockets board. Other sockets boards may have different power requirements. Refer to the sockets board documentation for more information.
8. Plug the power adapter into a wall socket.

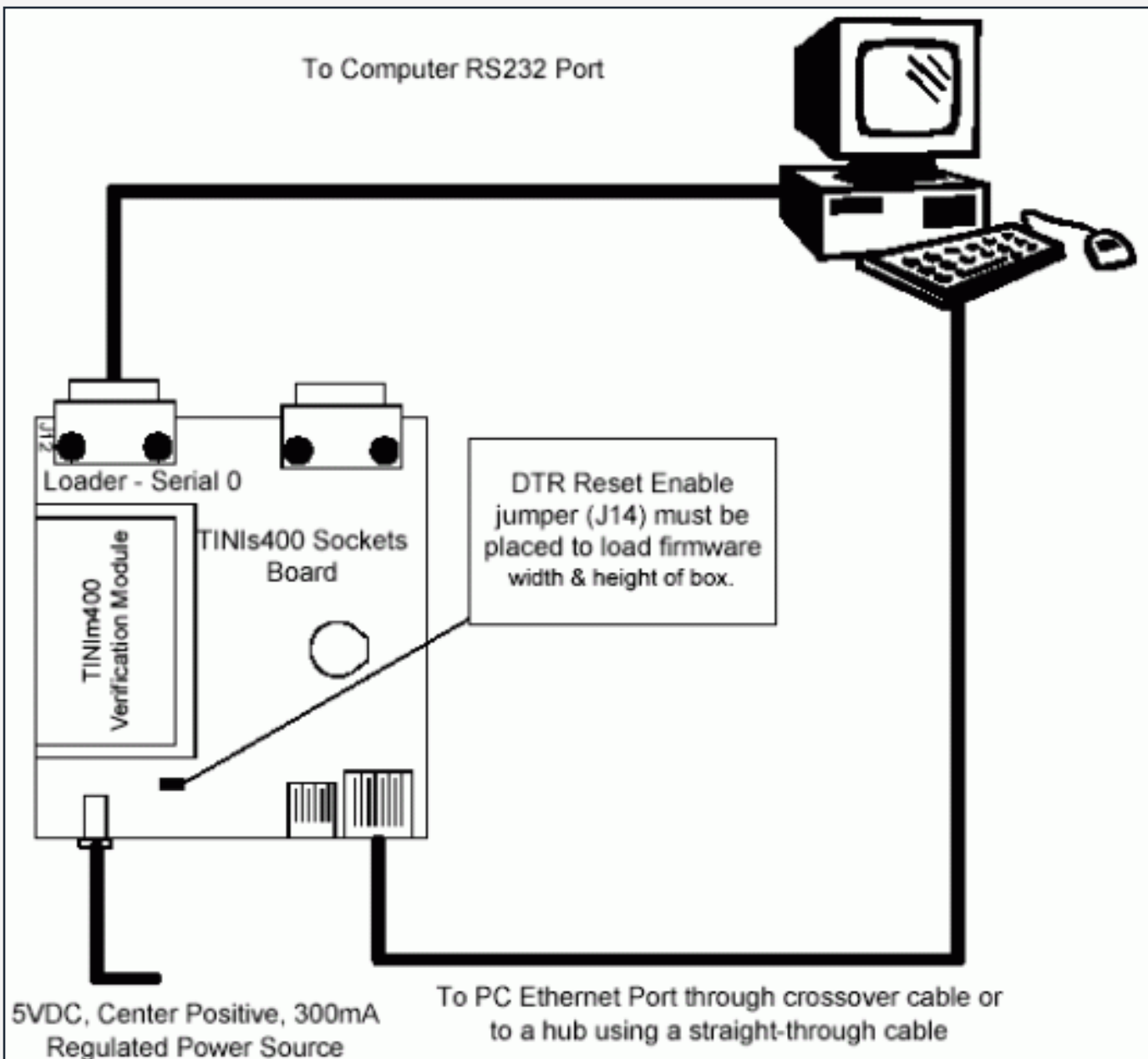


Figure 1. Typical hardware setup.

Software Setup

First, download and install the Java Development Kit and Java Communications API from Sun Microsystems. Installation instructions and example applications are included in both packages. You should verify correct installation by running the `BlackBox` example provided with the Java Communications API.

Loading the TINI Runtime Environment

Download the [TINI Software Development Kit](#). The download file should be unzipped to a location of your choice on your hard drive (<TINI SDK Install Dir>).

To begin using the TINI400 Verification Module, the TINI runtime environment must first be loaded into the flash of the verification module. Loading the runtime environment requires executing a program called `JavaKit`. Application binaries are loaded through the serial port of the TINI verification module.

Running JavaKit

1. To run JavaKit, open a command shell or DOS prompt on your PC. Change to the <TINI SDK Install Dir> \bin directory.
2. Next, type `java -classpath tini.jar JavaKit -400 -flash 40` at the command prompt and press return to run the JavaKit application. If you have the Java Development Kit and the Java Communications API properly installed, JavaKit should appear on the screen.
3. Select the port name where the TINI Verification Module is attached. Press the Open Port button. The default baud rate of 115200 should be used on all platforms where it is supported.
4. After opening the port, press the Reset button. The loader prompt should be displayed on the JavaKit screen (see **Figure 2**).

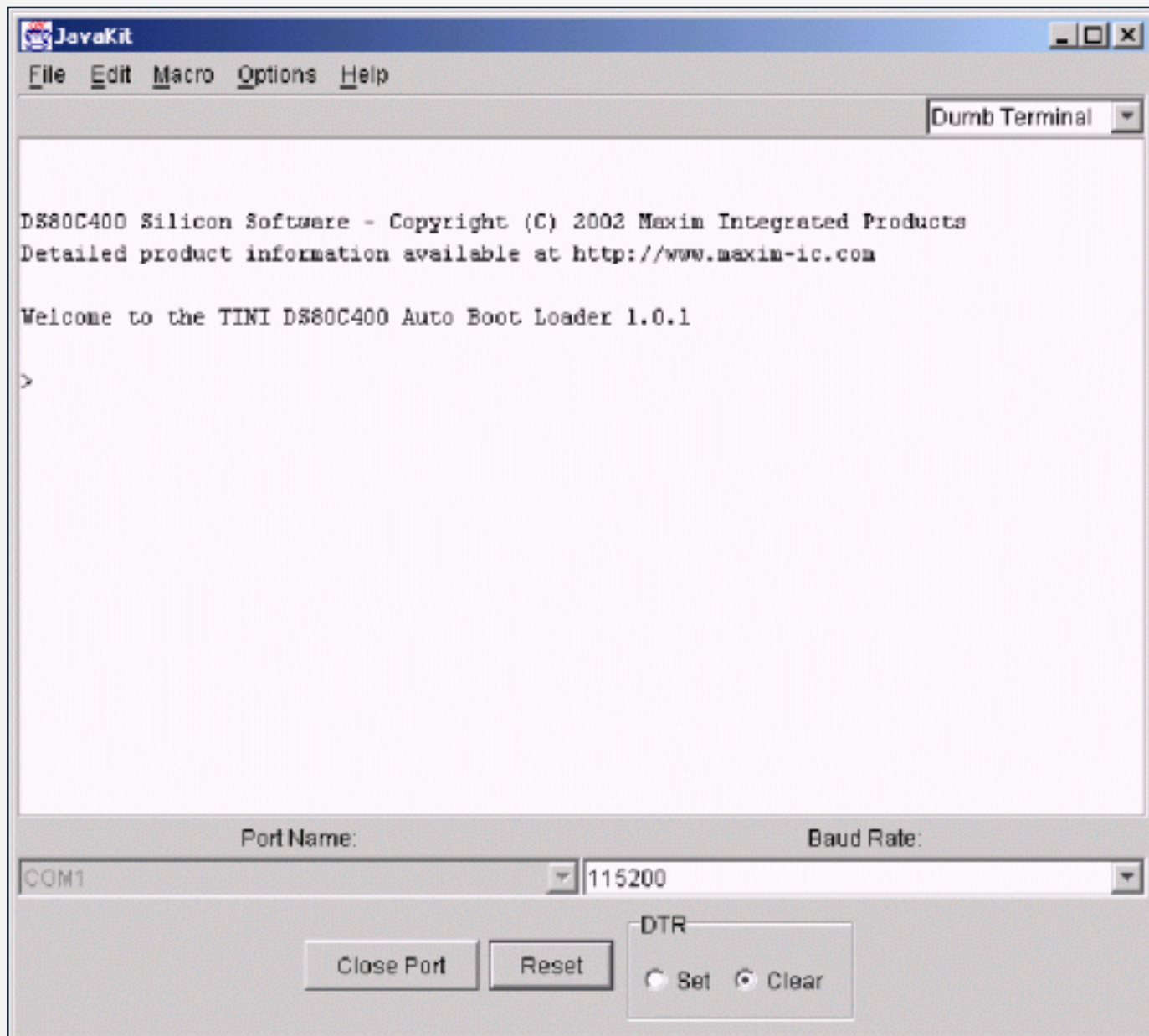


Figure 2. JavaKit loader prompt.

5. If the message is not displayed, verify the cable connections and check that the correct port name is selected. If everything appears to be attached correctly, read the `Running_JavaKit.txt` file contained in the TINI SDK for more detailed information.
6. At the JavaKit prompt, type `B0` and press return followed by `F0` and return. This clears the RAM of the TINI Verification Module. Any time a new runtime environment is loaded the memory should be cleared.
7. Now click on the File menu in JavaKit and select Load File. A file dialog will appear on the screen. Browse to the <TINI SDK Install Dir> \bin directory and select the `tini_400.tbin` file and click the Open button.

Load Complete will be displayed on the screen when the firmware is loaded, this may take up to 30 seconds.

- To load the command shell, click on the File menu and select Load File again. Select the slush_400.tbin file and click on the Open button. When Load Complete appears, slush has been successfully loaded into the flash of the TINIm400 Verification Module.
- Press the reset button and the JavaKit loader prompt shown in Figure 2 will be displayed.
- At the JavaKit prompt, type E and hit return. Slush will begin booting and text similar to that shown in Figure 3 will print on the JavaKit screen.

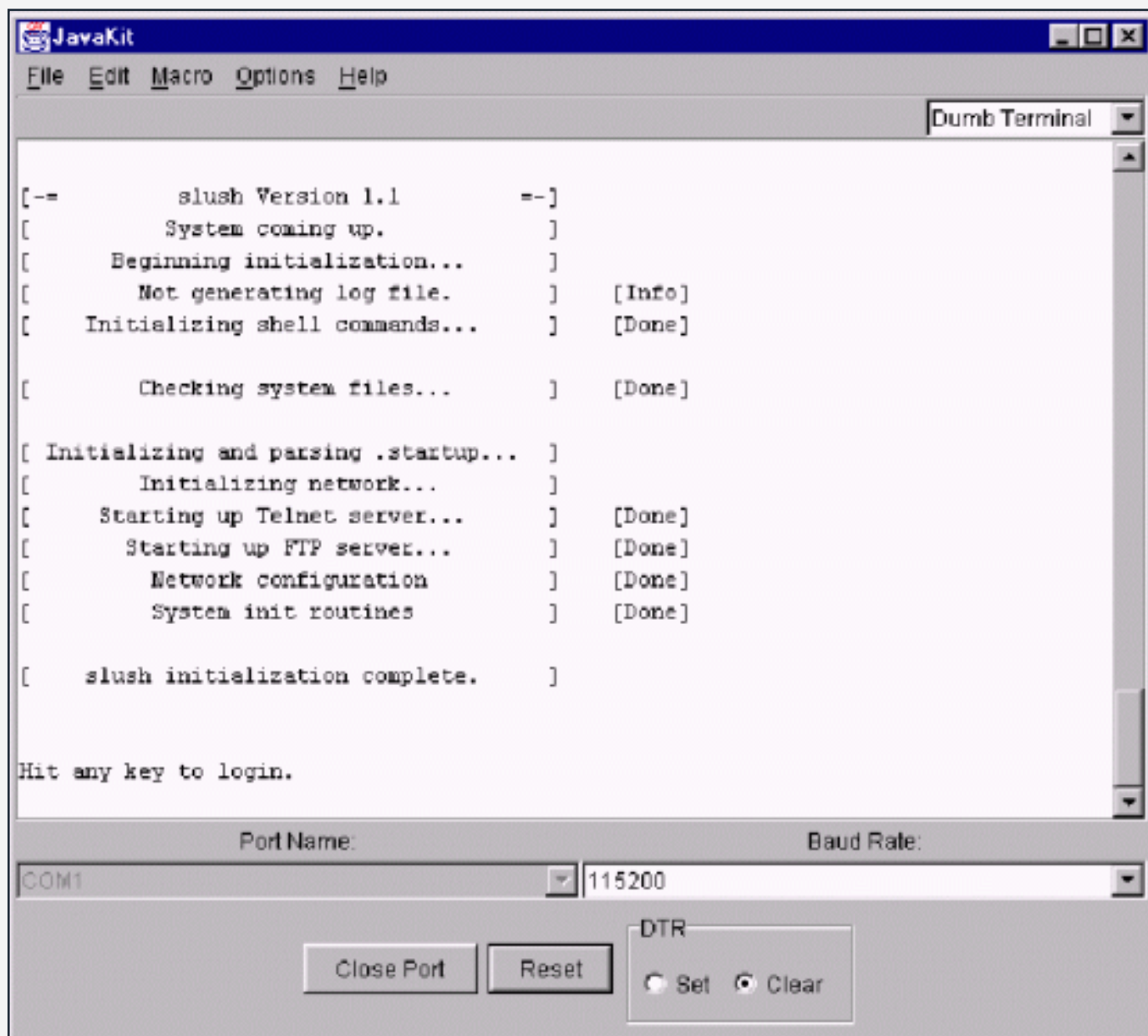


Figure 3. Slush login prompt.

- Press return and a Slush login prompt will be displayed. Type in the username root and press return. The initial password is tini.
- You should now be logged into Slush. To view a list of valid shell commands, type help at the command prompt. Detailed information on specific commands, such as dir, can be obtained by typing help followed by the command.

Network Setup

One of the primary features of the TINI Platform is its ability to access the network. The slush command to configure network access is ipconfig². Typing ipconfig -h at the command prompt will display detailed

configuration options. The -h option may be used with any slush command to view options.

The simplest method of connecting to the network is with DHCP, assuming it is supported on your network. To enable DHCP, type `ipconfig -d` and press enter. Once the TINI Verification Module has successfully leased a network address, a message is displayed on the slush prompt indicating success as in **Figure 4**.

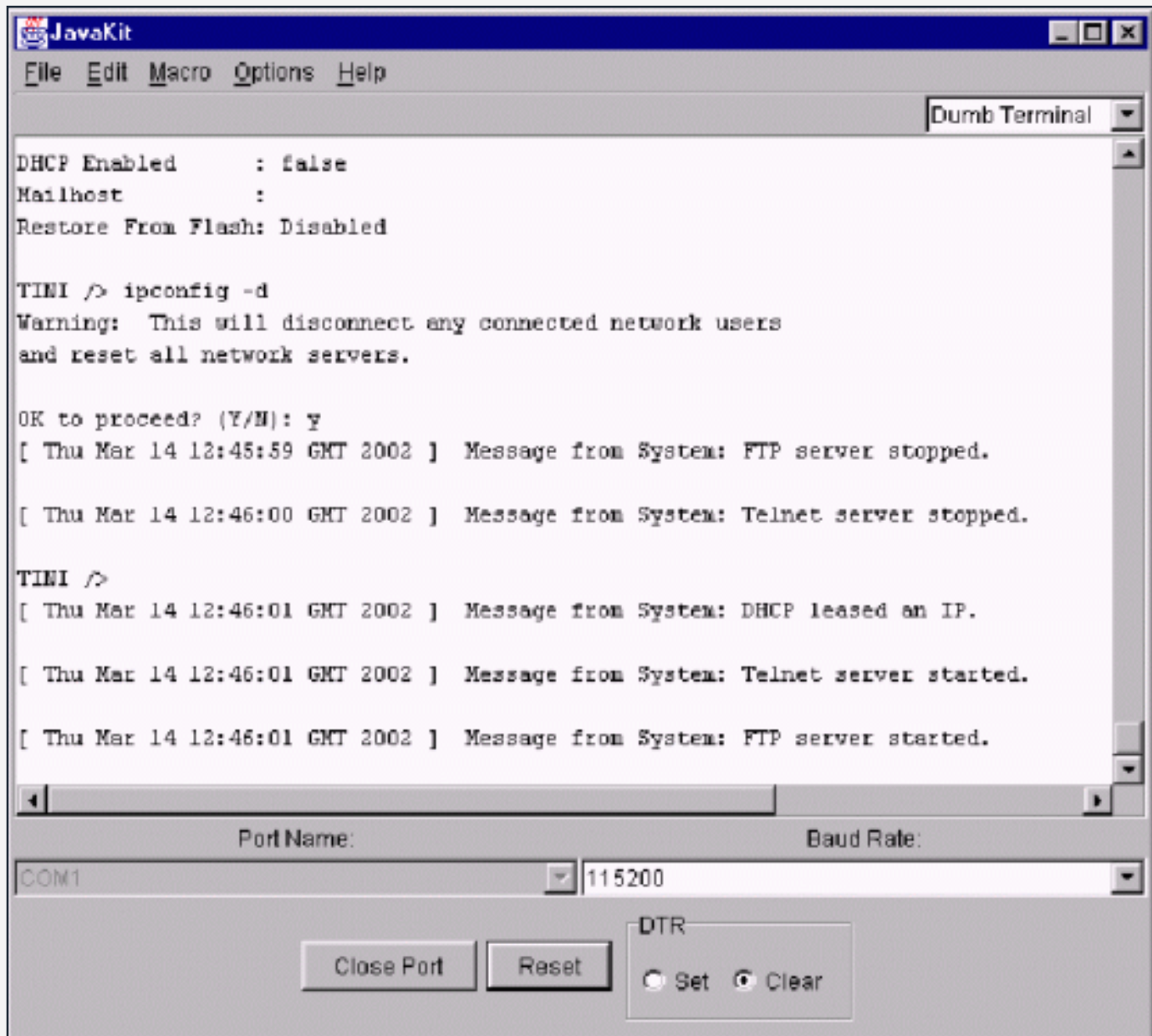


Figure 4. DHCP Network Configuration.

If your network does not support DHCP, then it is necessary to use a static IP address. First, you must get a valid network address and subnet mask from the network administrator. To set the IP address to a static value, use `ipconfig -a x.x.x.x -m y.y.y.y` where the IP address is denoted by x.x.x.x and the subnet is y.y.y.y. For example, to set your IP address to 192.168.0.1 with a subnet of 255.255.255.0, type `ipconfig -a 192.168.0.1 -m 255.255.255.0` at the slush command prompt and press enter.

A good way to test your network settings is to use the `ping` command. Either attempt to ping the TINI Verification Module from the host computer or ping the host computer from the TINI Verification Module using the slush `ping` command.

Running Your First Application

The TINI SDK contains many example applications. Each example application has batch files for compiling and

converting it to a binary file for execution on the TINI Runtime Environment. A very simple application included with the TINI SDK is `Blinky`. This application blinks an LED (DS1) on the TINI400. To transfer the application to your TINI400, open a command prompt on your host computer and change to the `<TINI SDK Install Dir>\examples\Blinky\bin` directory. This directory contains a file called `Blinky.tini`. From the command prompt on the host computer type `ftp`. The commands below will connect to the TINI and transfer the `Blinky.tini` file to the TINI400's file system. Replace the IP address used below with the IP address you set earlier in Slush. The default password for `ftp` and `telnet` on the TINI Runtime Environment is `tini`.

```
ftp> open 192.168.0.1
Connected to 192.168.0.1.
220 Welcome to slush. (Version 1.1) Ready for user login.
User (192.168.0.1:(none)): root
331 Password Required for root
Password:
230 User root logged in.
ftp> bin
200 Type set to Binary
ftp> put Blinky.tini
200 Command successful.
150 BINARY connection open, putting Blinky.tini
226 Closing data connection.
ftp: 514 bytes sent in 0.00Seconds 514000.00Kbytes/sec.
ftp> bye
221 Goodbye
```

The file now exists in the TINI file system. Next, connect to your TINI400 using `telnet` on your PC. Verify the existence of the `Blinky.tini` file using the Slush command `ls`. To run the `Blinky.tini` file, type `java Blinky.tini` followed by `enter`. The `Blinky` program executes and the LED should begin a steady blink.

More Information

Details of the TINI Platform are on the <http://www.maxim-ic.com> website. The "TINI Specification and Developer's Guide" is an invaluable resource when developing with the TINI platform and can be downloaded from the Maxim web site. Chapter 2 is dedicated to getting started with the TINI Platform and includes a detailed description of building and running several small example applications.

Additional Information and Links

- Java Development Kit - <http://java.sun.com/j2se>
- Java Communications API (Java COMM)
 - Windows/Solaris - <http://java.sun.com/products>
 - Linux - <http://www.rxtx.org>

Notes

¹The TINI Software Development Kit has been extensively tested with JDK1.3 from Sun Microsystems. Later versions of the JDK should work in the same manner. Check the documentation for your version of the JDK to verify the installation procedure.

²The DS80C400 ROM supports IPv6 for networking. By default, it is always on and configured automatically.

1-Wire is a registered trademark of Maxim Integrated Products, Inc.

Internet Explorer is a trademark of Microsoft Corp.

JDK is a trademark of Sun Microsystems, Inc.

TINI is a registered trademark of Maxim Integrated Products, Inc.

Application note 612: www.maxim-ic.com/an612

More Information

For technical support: www.maxim-ic.com/support

For samples: www.maxim-ic.com/samples

Other questions and comments: www.maxim-ic.com/contact

Automatic Updates

Would you like to be automatically notified when new application notes are published in your areas of interest? [Sign up for EE-Mail™](#).

Related Parts

DS80C400: [QuickView](#) -- [Full \(PDF\) Data Sheet](#) -- [Free Samples](#)

DS80C400-KIT: [QuickView](#) -- [Full \(PDF\) Data Sheet](#)

DSTINI400: [QuickView](#) -- [Full \(PDF\) Data Sheet](#)

DSTINIs400: [QuickView](#) -- [Full \(PDF\) Data Sheet](#)

AN612, AN 612, APP612, Appnote612, Appnote 612

Copyright © by Maxim Integrated Products

Additional legal notices: www.maxim-ic.com/legal