



# *NET+Works with GNU Tools Tutorial*





# *NET+Works with GNU Tools Tutorial*

---

**Operating system/version:** 6.2

**Part number/version:** 90000697\_B

**Release date:** April 2005

[www.netsilicon.com](http://www.netsilicon.com)

©2001-2005 NetSilicon, Inc. Printed in the United States of America. All rights reserved.

NetSilicon, NET+Works, and NET+OS are trademarks of NetSilicon, Inc. ARM is a registered trademark of ARM Limited. NET+ARM is a registered trademark of ARM Limited and is exclusively sublicensed to NetSilicon. Digi and Digi International are trademarks or registered trademarks of Digi International Inc. in the United States and other countries. All other trademarks are the property of their respective owners.

NetSilicon makes no representations or warranties regarding the contents of this document. Information in this document is subject to change without notice and does not represent a commitment on the part of NetSilicon. This document is protected by United States copyright law, and may not be copied, reproduced, transmitted, or distributed in whole or in part, without the express prior written permission of NetSilicon. No title to or ownership of the products described in this document or any of its parts, including patents, copyrights, and trade secrets, is transferred to customers. NetSilicon reserves the right to make changes to products without notice, and advises its customers to obtain the latest version of relevant information to verify, before placing orders, that the information being relied on is current.

**NETSILICON PRODUCTS ARE NOT DESIGNED, INTENDED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT APPLICATIONS, DEVICES, OR SYSTEMS, OR OTHER CRITICAL APPLICATIONS.**

NetSilicon assumes no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein. Nor does NetSilicon warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of NetSilicon covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used.

## Using This Guide

---

**R**eview this section for basic information about this guide, as well as for general support contact information.

### **About this guide**

This guide describes NET+OS with GNU Tools and how to use it as part of your development cycle. Part of the NET+Works integrated product family, NET+OS is a network software suite optimized for the NET+ARM.

### **Software release**

This guide supports NET+OS 6.2. By default, this software is installed in the `C:\netos62_gnu` directory.

## Who should read this guide

This guide is for software engineers and others who use NETWorks for NET+OS.

To complete the tasks described in this guide, you must:

- Be familiar with installing and configuring software.
- Have sufficient user privileges to do these tasks.
- Be familiar with network software and development board systems.

## Conventions used in this guide

This table describes the typographic conventions used in this guide:

This convention	Is used for
<i>italic type</i>	Emphasis, new terms, variables, and document titles.
<b>bold, sans serif type</b>	Menu commands, dialog box components, and other items that appear on-screen.
<b>menu name</b> → <b>option</b>	Menu commands. The first word is the menu name; the words that follow are menu selections.
monospaced type	Filenames, pathnames, and code examples.

## Related documentation

- *NET+Works Quick Installation Guide* describes how to install the hardware.
- *NET+Works with GNU Tools BSP Porting Guide* describes how to port the board support package (BSP) to a new hardware application with GNU tools.
- *NET+Work swith GNU Tools Programmer's Guide* describes how to use NET+OS to develop programs for your application and hardware.
- NET+Works online help describes the application programmer interfaces (APIs) that are provided with NET+OS.

For information about third-party products and other components, review the documentation CD-ROM that came with your development kit. The online help is located in `C:\netos62_gnu\Documentation`.

For information about the processor you are using, see your NET+Works hardware documentation.

## Documentation updates

NetSilicon occasionally provides documentation updates on the Web site.

Be aware that if you see differences between the documentation you received in your NET+Works package and the documentation on the Web site, the Web site content is the latest version.

## Customer support

To get help with a question or technical problem with this product, or to make comments and recommendations about our products or documentation, use the contact information listed in the next tables.

### *NetSilicon support*

For	Contact information
Technical support	Telephone: 1 800 243-2333 / 1 781 647-1234 Fax: 1 781 893-1388 E-mail: <a href="mailto:tech_support@netsilicon.com">tech_support@netsilicon.com</a>
Documentation	<a href="mailto:techpubs@netsilicon.com">techpubs@netsilicon.com</a>
NetSilicon home page	<a href="http://www.netsilicon.com">www.netsilicon.com</a>
Online problem reporting	<a href="http://www.netsilicon.com/support/problemreporting.jsp">www.netsilicon.com/support/problemreporting.jsp</a>



## *Digi support*

<b>For</b>	<b>Contact information</b>
Technical support	Telephone: 1 877 912-3444 / 1 952 912-3456 Fax: 1 952 912-4960
Documentation	<a href="mailto:techpubs@digi.com">techpubs@digi.com</a>
Digi home page	<a href="http://www.digi.com">www.digi.com</a>
Online problem reporting	<a href="http://www.digi.com/support/eservice/eservicelogin.jsp">www.digi.com/support/eservice/eservicelogin.jsp</a>



# Welcome to the NET+Works with GNU Tools Tutorial

---

In this hands-on exercise, you will:

- Set up a MAJIC or Raven debugger.
- Build and download a sample application.
- Run and debug the sample application.
- Specify configuration options for the development board.



**Note** The tutorial is brief – approximately 20 minutes – so you will be able to complete it in one sitting.

## Task 1: Preparing to do the tutorial

.....  
This section describes what you need to do before you get started.

### Make sure the hardware and software are installed

Verify that the hardware and the NET+Works 6.2 software are installed.

If you need to do the installation, see the *Quick Installation Guide*. The software installation uses a wizard to guide you through the process.

The instructions in this guide are based on the assumption that NET+Works is installed in the default installation directory:

```
C:/netos62_gnu
```

### Gather information you'll need (MAJIC only)

See your network administrator for the information in the list shown next. You'll use this information when you set up the MAJIC's IP address and when you configure the development board.

You may find it helpful to write down the information in the spaces provided.

▫ IP address for the board:

\_\_\_\_\_

▫ IP address for the MAJIC:

\_\_\_\_\_

▫ Subnet mask:

\_\_\_\_\_

▫ Default gateway:

\_\_\_\_\_

▫ MAC address:

\_\_\_\_\_

### What's next?

Go right on to the next section; you can begin the tutorial.

## Task 2: Setting up your debugger

In this section, if you're using a MAJIC probe, you'll make software configuration settings. The EPI MAJIC Setup Wizard guides you through this process, in which you'll make selections and provide information.

If you're using a Raven debugger, you'll install it.

### Configuring the MAJIC probe

#### ► To configure the MAJIC probe:

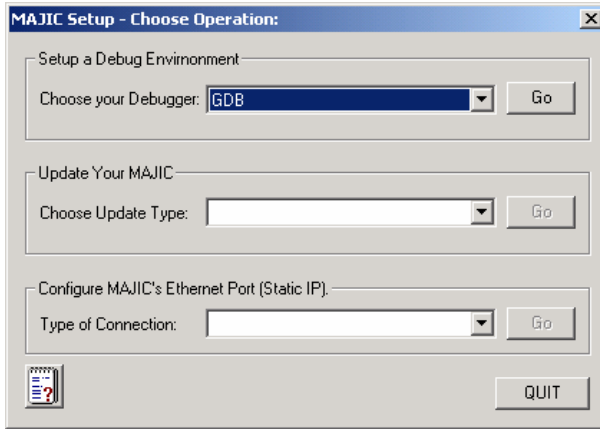
- 1 Set up an MDI server icon - the interface between the gdb and the MAJIC - by doing one of these steps:
  - Windows XP systems. Select Start → All Programs → EPI Tools-EDTA 2.2a → MAJIC Setup Wizard.
  - Windows 2000 systems. Select Start → Programs → EPI Tools-EDTA 2.2a → MAJIC Setup Wizard.

The EPI MAJIC Setup Wizard Introduction window opens:



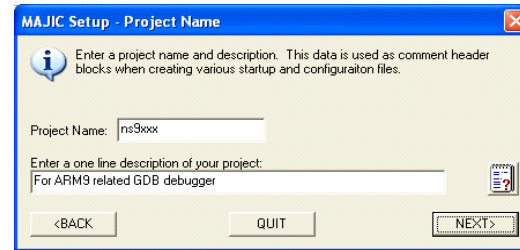
- 2 After you review the introduction, click NEXT.

The Choose Operation window opens:



- 3 From the Choose Your Debugger pulldown menu, select GDB and click Go.

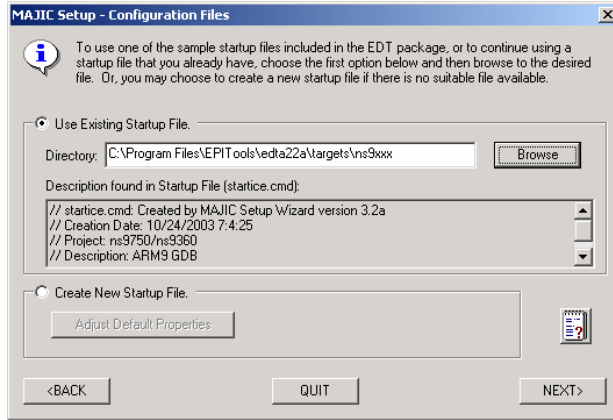
The Project Name window opens:



- 4 Create a new project by entering a project name and a brief description. Then click NEXT.



The Configuration Files window opens:



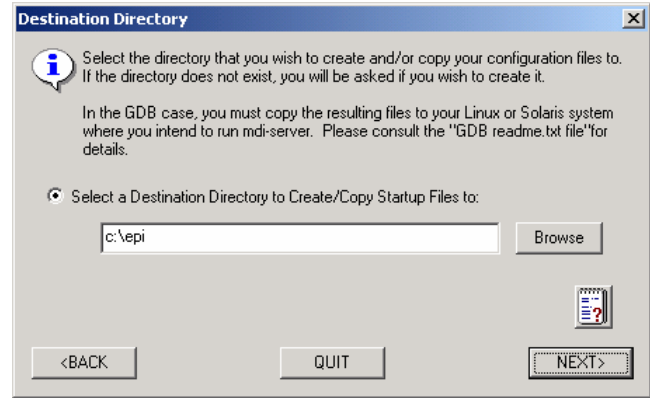
**7** Click Use Existing Startup File.

Then do these steps:

- a Click Browse.
- b Navigate to the Program Files → EPI Tools → edta22a → Targets → ns9xxx directory, click the startice.cmd file, and click Open.

and then click NEXT.

The Destination Directory window opens:

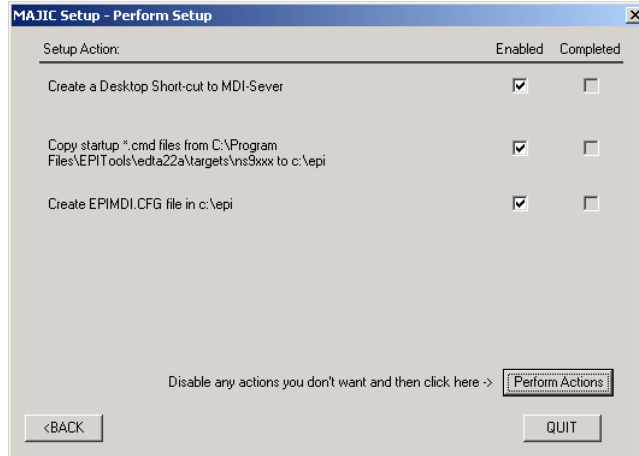


**8** Click Select a Destination Directory to Create/Copy Startup Files to.

Then browse to the directory you want to use for files that are created or copied during the MAJIC setup, and click NEXT.



The **Perform Setup** window opens with a summary of your selections:



**9** Check **Enabled** for each item, then click **Perform Actions**.

This step creates MAJIC setup files in the directory you specified in step 8. The setup program also creates a shortcut to the MDI server on your desktop.

**10** To exit from the wizard, click **Done**.

## Installing the Raven debugger

► To install the Raven:

- 1 Get a cable with a 20-pin connector for the Raven. (The Raven has both a 14- and a 20-pin cable installed.)
- 2 Connect the parallel cable from the Raven debugger to the parallel port on your PC.

### What's next?

You're ready to go on to the next section to set up the IP address of your MAJIC probe.

If you're using a Raven debugger, you can skip Task 3 and go right to Task 4.

### Task 3: Setting up the IP address of the MAJIC probe

This section describes how to set up the IP address of the MAJIC probe. In this procedure, you'll use the information you got from your network administrator.

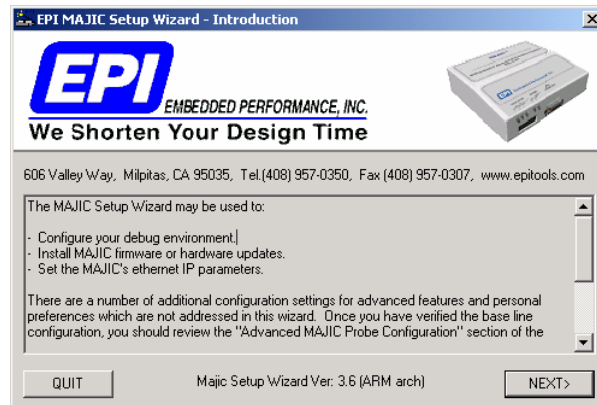
You'll use the EPI MAJIC Setup Wizard for this procedure.

If you're using a Raven debugger, skip this section and go right to "Task 4: Start a terminal session."

#### ► To set up the MAJIC's IP address:

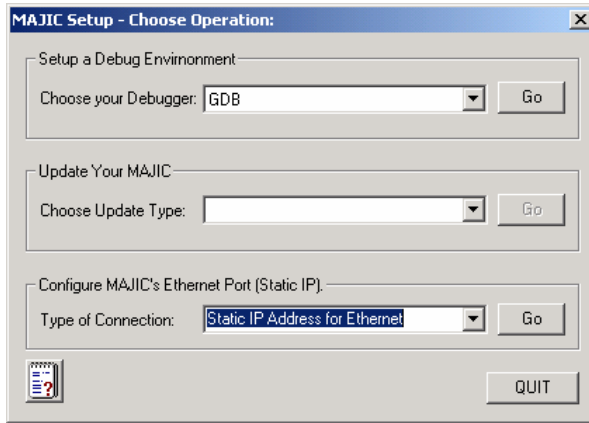
- 1 Do one of these steps:
  - Windows XP systems. Select Start → All Programs → EPI Tools-EDTA 2.2a → MAJIC Setup Wizard.
  - Windows 2000 systems. Select Start → Programs → EPI Tools-EDTA 2.2a → MAJIC Setup Wizard.

The EPI MAJIC Setup Wizard Introduction window opens:



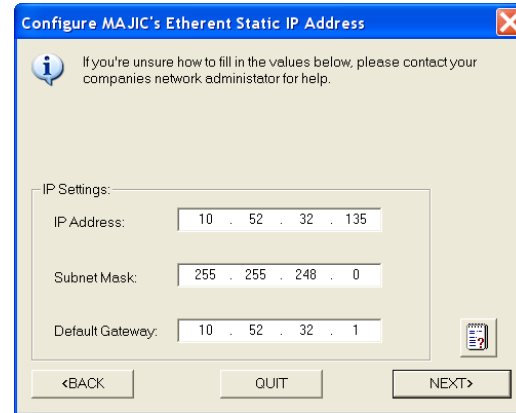
- 2 Click NEXT.

The Choose Operations window opens:



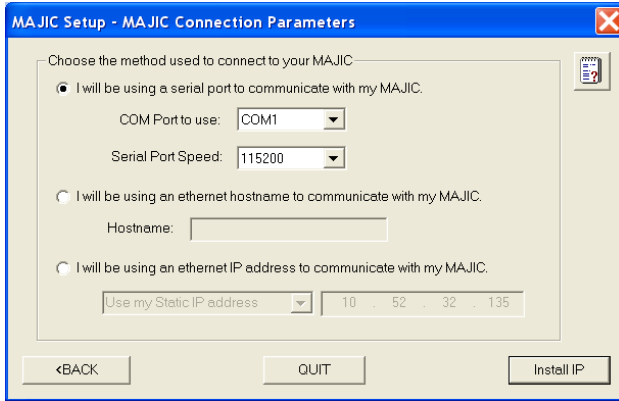
- 3 From the Type of Connection pulldown menu, select Static IP Address for Internet, and then click Go (as indicated by the arrow).

The Configure MAJIC's Ethernet Static IP Address window opens:



- 4 Enter the information you got from you network administrator:
  - IP address
  - Subnet mask
  - Default gatewayand then click NEXT.

The MAJIC Connection Parameters window opens:

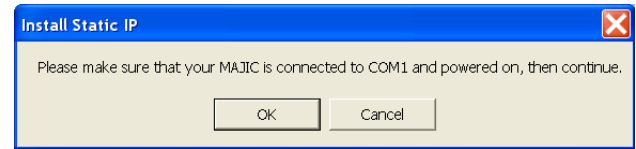


**5** Do these steps:

- a Click **I will be using a serial port to communicate with my MAJIC.**
- b Connect the MAJIC serial cable between the MAJIC JTAG interface controller's serial port and an available COM port on your PC.

- c From the COM port to use pulldown menu, select the serial port number. Make sure no other programs are using the COM port you select.
- d Click **Install IP.**

The Install Static IP dialog box opens:



- 6 Verify that the serial cable is attached from your PC to the MAJIC probe, and then click **OK.**

A dialog box and a DOS window open.

- 7 In the **Check Your Installation Result** dialog box, confirm that the IP address information in the DOS window is correct by clicking **OK.**

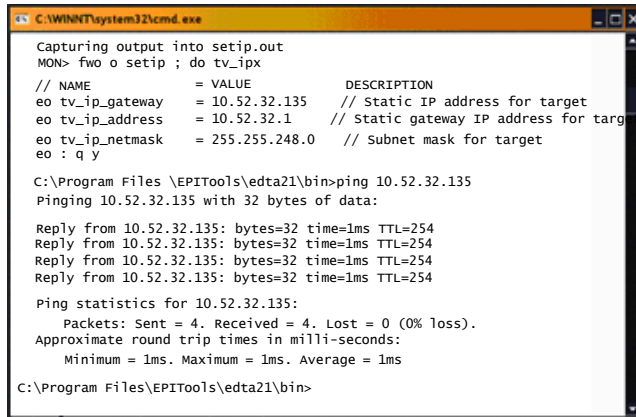
If there is a problem, correct the problem and go back to step 1 of this section.

- 8 Connect an Ethernet cable from your MAJIC probe to your LAN hub or switch.
- 9 After the MAJIC probe turns off, power cycle the MAJIC.
- 10 In the DOS window, ping the IP address by entering:

```
ping IP_ADDR
```

where *IP\_ADDR* is the IP address of the MAJIC.

This is what you see in the DOS window:



```
C:\WINNT\System32\cmd.exe
Capturing output into setip.out
MON> fwo o setip ; do tv_ipx
// NAME           = VALUE           DESCRIPTION
eo tv_ip_gateway  = 10.52.32.135    // Static IP address for target
eo tv_ip_address  = 10.52.32.1      // Static gateway IP address for target
eo tv_ip_netmask  = 255.255.248.0  // subnet mask for target
eo : q y

C:\Program Files\EPITools\edta21\bin>ping 10.52.32.135
Pinging 10.52.32.135 with 32 bytes of data:

Reply from 10.52.32.135: bytes=32 time=1ms TTL=254
Reply from 10.52.32.135: bytes=32 time=1ms TTL=254
Reply from 10.52.32.135: bytes=32 time=1ms TTL=254
Reply from 10.52.32.135: bytes=32 time=1ms TTL=254

Ping statistics for 10.52.32.135:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 1ms, Average = 1ms

C:\Program Files\EPITools\edta21\bin>
```

Note that the colors of the text and background are reversed in this document for easier reading.

If the ping succeeds, the IP address is installed.

- 11 Close the DOS window.

- 12 In the Check Your Install Results dialog box, click OK.

## What's next?

In the next section, you'll start a terminal session.

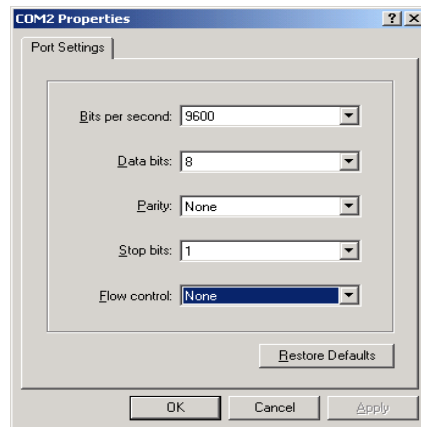
## Task 4: Start a terminal session

From your PC, open a terminal connection to the development board so you can view outputs from the board. Although several applications are available for this purpose, this tutorial uses HyperTerminal.

As you go through the tutorial, keep the HyperTerminal window open for quick access.

### ► To open a HyperTerminal window:

- 1 Do one of these steps:
  - Windows XP systems. Select **Start** → **All Programs** → **Accessories** → **Communications** → **HyperTerminal**.
  - Windows 2000 systems. Select **Start** → **Programs** → **Accessories** → **Communications** → **HyperTerminal**.
- 2 Use the settings shown here:



This example uses COM2; use the available port on your system.

- 3 When you finish reviewing or changing settings, click **OK**.

## What's next?

You're ready to build the BSP and the sample application, and then download the application.



## Building the template application

### ► To build the template application:

- 1 Change to the directory in which the template resides:

```
cd c:/netos62_gnu/src/apps/template/32b
```

and press Enter.

- 2 If you are using a MAJIC, enter this command:

```
make PLATFORM="my platform" gdbinit
```

and press Enter.

This step creates a `.gdbinit` file, which is the GNU debugger initialization file. This is the only time you'll use this command.

- 3 If you're using a Raven debugger, do one of these steps:

- **ARM9.** Copy `c:/netos62_gnu/debugger_files/gdbns9xxx.raven` to the 32b folder for your application, and rename it `.gdbinit`.
- **ARM7.** Copy `c:/netos62_gnu/debugger_files/gdbyourplatform.raven` to the 32b folder for your application, and rename it `.gdbinit`.

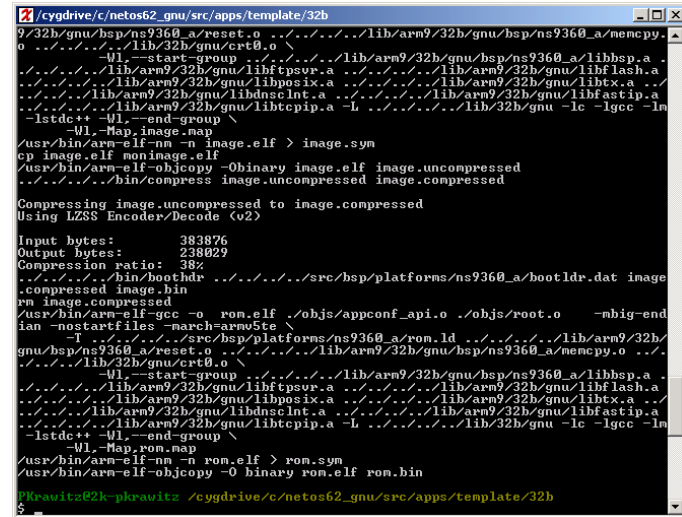
For example, your platform could be `connectsp`.

- 4 To build the application, enter this command:

```
make PLATFORM="my platform"
```

and press Enter.

This command builds the `image.elf` file, which is the image you'll download to the debugger, and `rom.bin`, which is the corresponding ROM image. This process, which you can see in the console, takes a few seconds.



```
C:\cygdrive/c/netos62_gnu/src/apps/template/32b
9/32b/gnu/bsp/ns9360_a/reset.o ../../lib/arm9/32b/gnu/bsp/ns9360_a/memcpy.o
9/32b/gnu/bsp/ns9360_a/reset.o ../../lib/arm9/32b/gnu/crt0.o
-ld -start-group ../../lib/arm9/32b/gnu/bsp/ns9360_a/libbsp.a
../../lib/arm9/32b/gnu/libftpsur.a ../../lib/arm9/32b/gnu/libflash.a
../../lib/arm9/32b/gnu/libposix.a ../../lib/arm9/32b/gnu/libtx.a
../../lib/arm9/32b/gnu/libnscint.a ../../lib/arm9/32b/gnu/libfastip.a
../../lib/arm9/32b/gnu/libtcpip.a -L ../../lib/32b/gnu -lc -lgcc -ln
-lstdc++ -Wl,--end-group \
-Wl,--Map,image.map
/usr/bin/arm-elf-ld -n image.elf > image.syn
cp image.elf nonimage.elf
cp /usr/bin/arm-elf-objcopy -O binary image.elf image.uncompressed
../../bin/compress image.uncompressed image.compressed
Compressing image.uncompressed to image.compressed
Using LZSS Encoder/Decoder (v2)
Input bytes: 383876
Output bytes: 238029
Compression ratio: 38%
../../bin/boothdr ../../src/bsp/platforms/ns9360_a/boothdr.dat image
.compressed image.bin
rm image.compressed
/usr/bin/arm-elf-gcc -o rom.elf ./objs/appconf_api.o ./objs/root.o -mbig-end
ian -nostartfiles -march=armv5te \
-I ../../src/bsp/platforms/ns9360_a/rom.ld ../../lib/arm9/32b/
gnu/bsp/ns9360_a/reset.o ../../lib/arm9/32b/gnu/bsp/ns9360_a/memcpy.o
../../lib/32b/gnu/crt0.o \
-Wl,--start-group ../../lib/arm9/32b/gnu/bsp/ns9360_a/libbsp.a
../../lib/arm9/32b/gnu/libftpsur.a ../../lib/arm9/32b/gnu/libflash.a
../../lib/arm9/32b/gnu/libposix.a ../../lib/arm9/32b/gnu/libtx.a
../../lib/arm9/32b/gnu/libnscint.a ../../lib/arm9/32b/gnu/libfastip.a
../../lib/arm9/32b/gnu/libtcpip.a -L ../../lib/32b/gnu -lc -lgcc -ln
-lstdc++ -Wl,--end-group \
-Wl,--Map,rom.map
/usr/bin/arm-elf-ld -n rom.elf > rom.syn
/usr/bin/arm-elf-objcopy -O binary rom.elf rom.bin
PKrauita@2k-pkrauitz /cygdrive/c/netos62_gnu/src/apps/template/32b
$
```



## Downloading the template application

### ► To download Hello World:

- 1 Power cycle the development board and the MAJIC or Raven.
- 2 Do one of these:
  - If you are using a Raven, click either the `ocdremote ARM7` or `ARM9` icon.  
If you get Error 43, check whether your board has power and the Raven is connected properly, and then try again.  
(If the window closes immediately, or is blank, power-cycle your board and try again.)
  - If you are using a MAJIC, click the MDI server icon on your desktop.
- 3 Make sure you are in an `arm -elf` shell. If you are not, enter this command:  

```
xtools arm -elf
```

and press Enter.

- 4 To start up `gdb` - the debug environment - enter this command:

```
gdbtk -se image.elf
```

and press Enter.

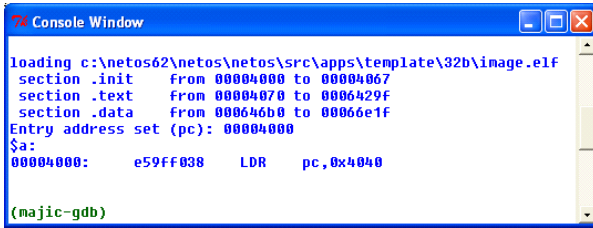
If you are using a Raven, when you see a pop-up window with this message:

The target may not be able to correctly handle a memory-write-packet-size of 1024 bytes. Change the packet size?

Click **Yes**.

The debugger **Console** window opens, and the download begins.

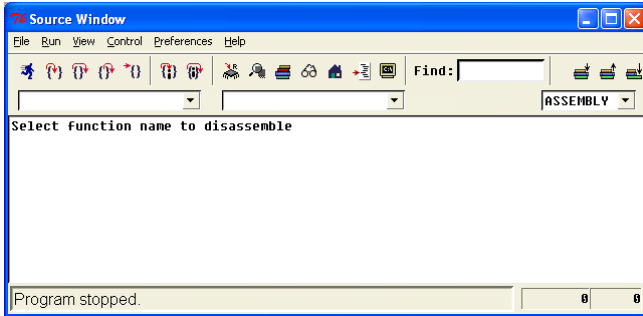
If you are using a Raven debugger, and you see errors in the Console window, power-cycle your board, and do this step again.



```
Console Window
loading c:\netos62\netos\netos\src\apps\template\32b\image.elf
section .init      from 00004000 to 00004067
section .text      from 00004070 to 0000429F
section .data      from 000046b0 to 00006e1F
Entry address set (pc): 00004000
$a:
00004000:  e59ff038   LDR   pc,0x4040
(majic-gdb)
```

This process takes about a minute to complete.

Then the debugger Source window opens:



## What's next?

Now you're ready to run and debug the template application. Go on to the next section.

## Task 6: Running and debugging the template application

You run and debug your application in the debugger window. The instructions in this section are for both the Raven and the MAJIC.

### Running the application

► To run the template application, do one of these steps:

- *Either* select **View** → **Console**.

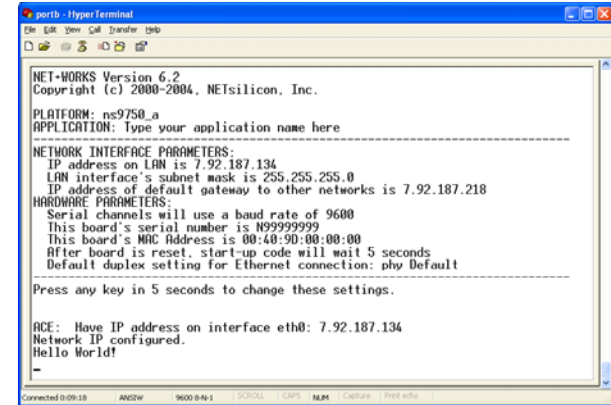
A GNU Console window opens.

In the **Console** window, enter **c** and press **Enter**.

- *Or* in the **Source** window, click this icon:

→{ }

The application appears in the terminal window:



```
portb - HyperTerminal
File Edit View Call Transfer Help

NET-WORKS Version 6.2
Copyright (c) 2000-2004, NETsilicon, Inc.

PLATFORM: ns9750_a
APPLICATION: Type your application name here

-----
NETWORK INTERFACE PARAMETERS:
IP address on LAN is 7.92.187.134
LAN interface's subnet mask is 255.255.255.0
IP address of default gateway to other networks is 7.92.187.218
HARDWARE PARAMETERS:
Serial channels will use a baud rate of 9600
This board's serial number is N99999999
This board's MAC Address is 00:40:90:00:00:00
After board is reset, start-up code will wait 5 seconds
Default duplex setting for Ethernet connection: obey Default
-----

Press any key in 5 seconds to change these settings.

ACE: Have IP address on interface eth0: 7.92.187.134
Network IP configured.
Hello World!
-
```

Note that **Hello World** appears in the last line of the window.

## Debugging the application

### ► To debug the template application:

- 1 In the **Source** window, click **Stop**.

When you see this warning, which is not a fatal error, click **OK**:



- 2 Enter this command in the **Console** window:

```
b function name
```

where you replace *function name* with the function at which you want to insert the break point and then press Enter.

For this example, enter:

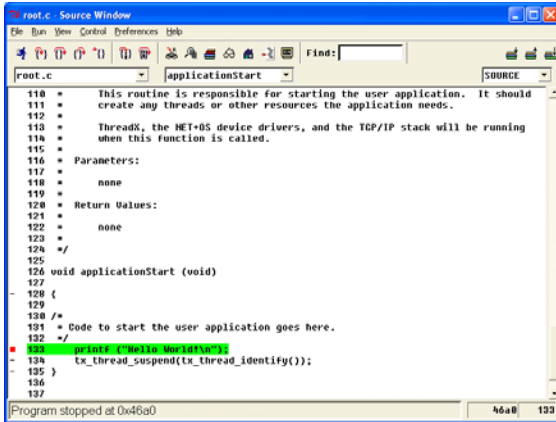
```
b applicationStart
```

- 3 Close the debugger by closing the **Source** window.
- 4 To start up gdb (the debug environment), enter this command in the X-Tools window:  

```
gdbtk -se image.elf
```

and press Enter.
- 5 To continue to run the program, in the **Console** window, enter **C** and then press Enter.

You see this window:



```
root.c - Source Window
File Edit View Control Preferences Help
Find:
root.c applicationsStart SOURCE
110 * This routine is responsible for starting the user application. It should
111 * create any threads or other resources the application needs.
112 *
113 * ThreadX, the NET-BS device drivers, and the TCP/IP stack will be running
114 * when this function is called.
115 *
116 * Parameters:
117 *
118 * none
119 *
120 * Return Values:
121 *
122 * none
123 *
124 */
125
126 void applicationStart (void)
127 {
128 {
129
130 /*
131 * Code to start the user application goes here.
132 */
133 printf ("Hello World!\n");
134 tx_thread_suspend(tx_thread_identify());
135 }
136
137
Program stopped at 0x48a0 46a0 133
```

## What's next?

Go on to the next section to configure the development board.

## Task 7: Configuring the development board

During this procedure, be prepared to move quickly to your HyperTerminal at step 10, because you have only a few seconds to respond to the prompt.

### ► To configure the development board:

- 1 Reboot the development board by disconnecting it from the power source and then reconnecting it.

- 2 Change to this directory:

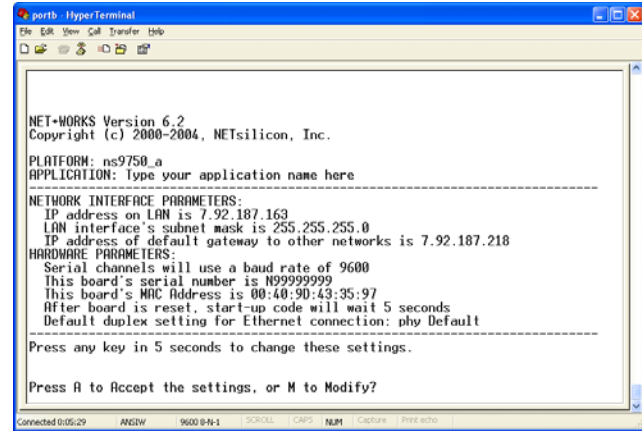
```
cd c:/netos62_gnu/src/apps/template
```

- 3 In the `appconf.h` file, verify that the `USE_NVRAM` define is set to 1.

*If you need to change this setting, you must rebuild your application.*

- 4 Run the application again as you did in Task 5, in the section “Downloading the template application.”
- 5 Go to the PC that’s running HyperTerminal, and press Enter.

You see this information in the HyperTerminal window:



```
portb HyperTerminal
File Edit View Call Transfer Help

NET+WORKS Version 6.2
Copyright (c) 2000-2004, NETsilicon, Inc.

PLATFORM: ns9750_a
APPLICATION: type your application name here
-----
NETWORK INTERFACE PARAMETERS:
IP address on LAN is 7.92.187.163
LAN interface's subnet mask is 255.255.255.0
IP address of default gateway to other networks is 7.92.187.218
HARDWARE PARAMETERS:
Serial channels will use a baud rate of 9600
This board's serial number is N999999999
This board's MAC Address is 00:40:90:43:35:97
After board is reset, start-up code will wait 5 seconds
Default duplex setting for Ethernet connection: phy Default
-----
Press any key in 5 seconds to change these settings.

Press A to Accept the settings, or M to Modify?

Connected 0:05:29 ANEW 9600 8-N-1 SERIAL CAPS NUM | Capture Print etc.
```

- 6 To change the configuration, press M and then press Enter.  
You are prompted for a password.
- 7 Enter Netsilicon, the default password, and press Enter.  
The first of a series of configuration prompts appears.

- 8 At each prompt, do one of these steps:
- To accept the current value, press Enter.
  - To change a setting, enter a value and press Enter.

After you scroll through the settings, a prompt indicates that you must press a key within five seconds if you want to change additional settings.

When you finish this procedure, you will have completed the tutorial.

### *Tips and suggestions*

Now that you've completed the exercise, here are some tips for when you start using NET+Works with GNU Tools with your own projects.

### **Where should I put my code, and why?**

Add your code as an example off of the `netos62_gnu/src/examples` directory. The software calls the `applicationStart` function in the `root.c` file.

Start by duplicating some other example and modify the makefiles. (For information about modifying makefiles, see the *NET+OS with GNU Tools BSP Porting Guide*).

A good choice is `naftpapp`, which is the FTP server example. It may make sense to add your application to this example. It will allow you reload new code after it's running in flash. Modify `root.c` to load your specific application, and modify `appconf.h` for your board settings.

### **What should my next step be?**

The next step should be running your application from flash. The flash code is broken up into 2 parts: the bootloader and your application. The bootloader (`rom.bin`) is located in the `netos62_gnu/src/bsp/platforms/your_platform` directory. Your application (`image.bin`) is located in the `netos62_gnu/src/examples/your_example/32b` directory.







