XAPP947 (v1.1) April 3, 2008

Reference System: VxWorks 6.x on the ML403 Embedded Development Platform

Author: Richard Griffin, Brian Hill

Abstract

This application note discusses the use of Wind River VxWorks Real-Time Operating System (RTOS) on a Xilinx ML403 board.

Included Systems

Included with this application note is one reference system:

www.xilinx.com/support/documentation/application_notes/xapp947.zip

Introduction

The use of the Wind River VxWorks Real-Time Operating System (RTOS) on Virtex™-4 embedded PowerPC™ processors continues to be a popular choice for high performance FPGA designs. The introduction of the Wind River Workbench design environment has enabled a new and easier way for designers to control the configuration of the VxWorks kernel. This guide shows the steps required to build and configure a ML403 Embedded Development Platform to boot and run the VxWorks RTOS. A VxWorks bootloader is created, programmed into Flash, and used to boot the design. The concepts presented here can be scaled to any PowerPC enabled development platform.

Hardware and Software Requirements

The hardware and software requirements are:

- Xilinx ML403 Development Board
- Xilinx Platform USB Cable or Parallel IV Cable
- RS232 Cable
- Ethernet Cable
- Serial Communications Utility Program (such as, HyperTerminal)
- Xilinx Platform Studio 10.1i
- Xilinx ISE™ 10.1i
- Wind River Workbench 2.5
- Wind River Probe / ICE
- Adaptor from Wind River Dual Header to the Xilinx Parallel IV cable ribbon connector. This adaptor must be purchased separately.
- A license for Wind River multi-core debugging for the Probe/ICE

Reference System Specifics

The included ML403 system was created with Base System Builder. The system includes a PowerPC 405, Multi-Port Memory Controller (MPMC), XPS GPIO connected to on-board LEDs and Push Buttons, XPS Interrupt Controller, XPS LL TEMAC Ethernet controller, XPS UART 16550, XPS Multi-Channel External Memory controller (MCH EMC) connected to on-board flash, and 16k bytes of XPS BRAM. The system address map is shown in Table 1.

Address Map

Table 1: **Reference System Address Map**

Instance	Peripheral	Base Address	High Address
DDR_SDRAM	mpmc	0x00000000	0x03FFFFFF
LEDs_Positions	xps_gpio	0x81400000	0x8140FFFF
Push_Buttons_Position	xps_gpio	0x81420000	0x8142FFFF
LEDs_4Bit	xps_gpio	0x81440000	0x8144FFFF
xps_intc_0	xps_intc	0x81800000	0x8180FFFF
TriMode_MAC_GMII	xps_ll_temac	0x81C00000	0x81C0FFFF
RS232_Uart	xps_uart16550	0x83E00000	0x83E0FFFF
DDR_SDRAM (SDMA)	mpmc	0x84600000	0x8460FFFF
FLASH	xps_mch_emc	0xFF000000	0xFF7FFFFF
xps_bram_if_cntlr_1	xps_bram_if_cntlr	0xFFFFC000	0xFFFFFFFF

Executing the Reference System

The bitstreams for the system are available in the `ready_for_download/` directory under the project root directory.

Executing the Reference System using the Pre-Built Bitstream and the Compiled Software Applications

To execute the system using files inside the `ready_for_download/` directory in the project root directory, follow these steps:

1. Change directories to the `ready_for_download` directory.
2. Connect the Xilinx Download cable to the board.
3. Use iMPACT to download the bitstream by using the following:

   ```
   impact -batch xapp947.cmd
   ```

4. Invoke XMD and connect to the processor by the following command:

   ```
   xmd -opt xapp947.opt
   ```

5. Download the executables by the following command:

   ```
   dow <executable>
   ```

Executing the Reference System from XPS

To execute the system using EDK, follow these steps:

1. Open `system.xmp` inside EDK.
2. Use **Hardware** → **Generate Bitstream** to generate a bitstream for the system.
3. Download the bitstream to the board with **Device Configuration** → **Download Bitstream**.
4. Launch XMD with **Debug** → **Launch XMD..**.
5. Download the executables by the following command:

   ```
   dow <executable>
   ```

Console Connection

Connect a serial cable to the RS232 port on the ML403. The terminal application, such as HyperTerminal, is configured as **9600** BPS, **8** Data Bits, No Parity, **1** Stop Bit, and No Flow Control.

Create and Modify the BSP

XPS is used to create a VxWorks Board Support Package (BSP) specific to the design provided with this application note with the following steps:

1. In XPS, choose **Software** → **Software Platform Settings**:
   a. OS selection of `ppc405_0` is changed in "Software Platform" to **vxworks6_3** version 1.01a.
   b. Change the STDIN and STDOUT settings in "OS and Libraries" to **RS232_Uart.**
   c. Add the RS232_UART and TriMode_MAC_GMII to the "connected peripherals" list in the VxWorks settings.
2. The VxWorks Board Support Package (BSP) is generated using **Software** → **Generate Libraries and BSPs**.
3. Create a directory in the EDK project called `modified_vxworks_bsp`.
4. Copy the contents of the `<edk_project_dir>\ppc405_0\bsp_ppc405_0` directory to the `modified_vxworks_bsp` directory.
5. Edit the `modified_vxworks_bsp\config.h` file to reflect the addresses applicable to a given system. The changes shown in **bold** below reflect the system supplied with this application note:

Note: The value of ROM_BASE_ADRS matches the base address of the `xps_mch_emc` peripheral chosen by Base System Builder for the system included with this application note. Enter the value appropriate to th particular system.

```
/* memory configuration */
#define LOCAL_MEM_LOCAL_ADRS 0
#undef  LOCAL_MEM_AUTOSIZE                    /* no run-time memory sizing */
#define LOCAL_MEM_SIZE       0x04000000        /* TODO: fixed memory size */
#define USER_RESERVED_MEM    0                 /* see sysMemTop() */

#define ROM_BASE_ADRS    0xff000000          /* base address of ROM */
#define ROM_TEXT_ADRS    (ROM_BASE_ADRS+0x100)     /* with PC & SP */
#define ROM_WARM_ADRS    (ROM_TEXT_ADRS+8)   /* warm reboot entry */
#define ROM_SIZE         0x00800000          /* 8MB ROM space */
#define RAM_HIGH_ADRS    0x00C00000          /* RAM address for bootrom */
#define RAM_LOW_ADRS     0x00010000          /* RAM address for vxWorks */
...
#define CONSOLE_BAUD_RATE       9600     /* console baud rate */
```

6. Edit the `modified_vxworks_bsp\Makefile` file to reflect the changes highlighted in bold red type below:

```
ROM_TEXT_ADRS= ff000100 # ROM cold entry address
ROM_WARM_ADRS= ff000108 # ROM warm entry address
ROM_SIZE= 00800000 # number of bytes of ROM space
RAM_LOW_ADRS= 00010000 # RAM text/data address for vxWorks
RAM_HIGH_ADRS= 00C00000 # RAM text/data address for bootrom
```

Figure 1 shows how ROM_BASE_ADRS, RAM_HIGH_ADRS, RAM_LOW_ADRS, etc., are used in VxWorks bootloading process.

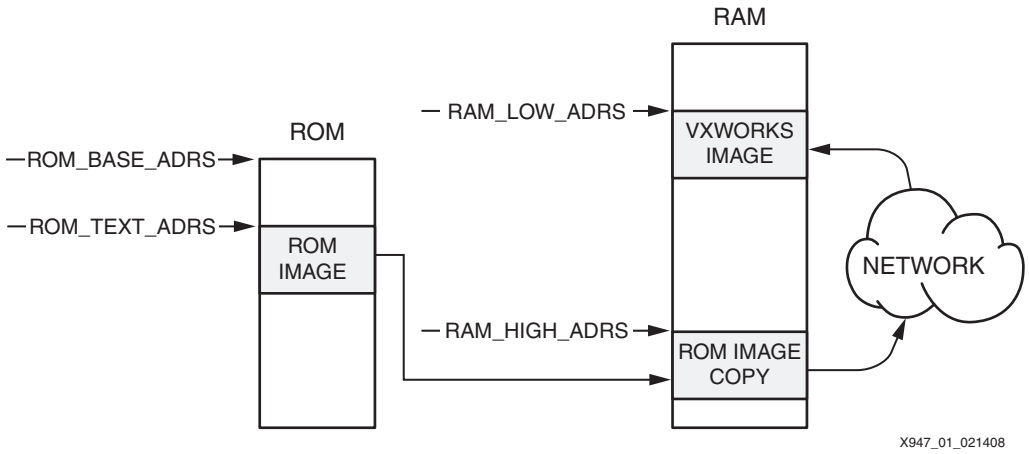

Figure 1: **Bootloading Process**

7. Edit the `modified_vxworks_bsp\sysNet.c` file to reflect the MAC address of the target board. The Ethernet MAC address on the ML403 is printed on a sticker under the System ACE CompactFlash socket. An example for the Ethernet MAC address `AB-CD-12-34-FF-E0` is shown:

```
static char XLlTemacMacAddr0[6] = { 0xAB, 0xCD, 0x12, 0x34, 0xFF, 0xE0 };
```

8. (optional) Edit the `sysLib.c` file to display a Virtex-4 message when VxWorks boots up. See example below.

```
char * sysModel (void)
{
/* return ("ppc405_0 VirtexII Pro PPC405"); */
    return ("Xilinx Virtex-4 FX PPC405");
}
```

Configure and Build a VxWorks Kernel Image

The modified VxWorks BSP is used with Wind River Workbench to create a VxWorks kernel image using the following instructions:

1. The Wind River Workbench tool is run. If prompted, choose **Start**, then **Workbench** at the main screen. Create a location for the workspace in the `<edk_project>\windriver_workspace` directory.
2. In the Project Navigator pane on the left of the screen, right-click and choose **New → VxWorks Image Project.**
   - Assign the project name `ml403_vxworks_image`, and choose to create the project within the workspace.
   - Choose to base the project upon a **board support package** and then browse to the `modified_vxworks_bsp` directory.
   - Set the tool chain to either **sfgnu** or **sfdiab**.
   - Click **Finish**. A project appears in the left hand Project Navigator pane, and the BSP will be analyzed by the tool.
3. Expand the project tree in the Project Navigator pane and double-click the **Kernel Configuration** submenu item. A hierarchical tree appears in the center window. The items in bold signify that they are included in the kernel, and the ones in standard text can be added by the user. To add and remove items from the kernel, right-click the item in question

and select the **Include (quick include)** or **Exclude (quick exclude)** options. Configure the kernel as follows:

- Include **development tool components** → **kernel shell components**
- Include **development tool components** → **WDB agent components**
- Include **development tool components** → **WDB agent components** → **select WDB mode** → **WDB system debugging**

4. Close the component configuration window, and save the changes.
5. In the Project Navigator pane, right-click on the **ml403_vxworks_image** project and select **Build Project**. The kernel is built and **"Build Finished……"** is shown at the bottom of the screen in the Build Console pane.
6. Ensure that the board is powered, then configure the board using the **Device Configuration** → **Download Bitstream** menu option in XPS. Check that the DONE LED on the target is illuminated.
7. Unplug the Xilinx Download cable from the target board, and connect the Wind River Probe/ICE to the same socket using the 14-pin adapter.
8. From the Wind River Workbench tool, configure a target server to connect to the board via the Wind River Probe / ICE debug cable. The ML403 board has a combined JTAG chain, therefore a BRD file must be specified to enable the Wind River tools to correctly connect to the board. Select the **Board File** radio button, then browse to the BRD file representing the ML403 board. This file can be found in the Wind River installation in the `<windriver_installation>\workbench-2.5\dfw\0145b\host\boardfiles\PowerPC\4xx\Xilinx` directory.
   - Right-click in the Target Manager window and select **New** → **Connection**, or click on the **New Connection** icon.

Figure 2: **New OCD Connection**

- Choose the type of cable that is available, then click **Next**.

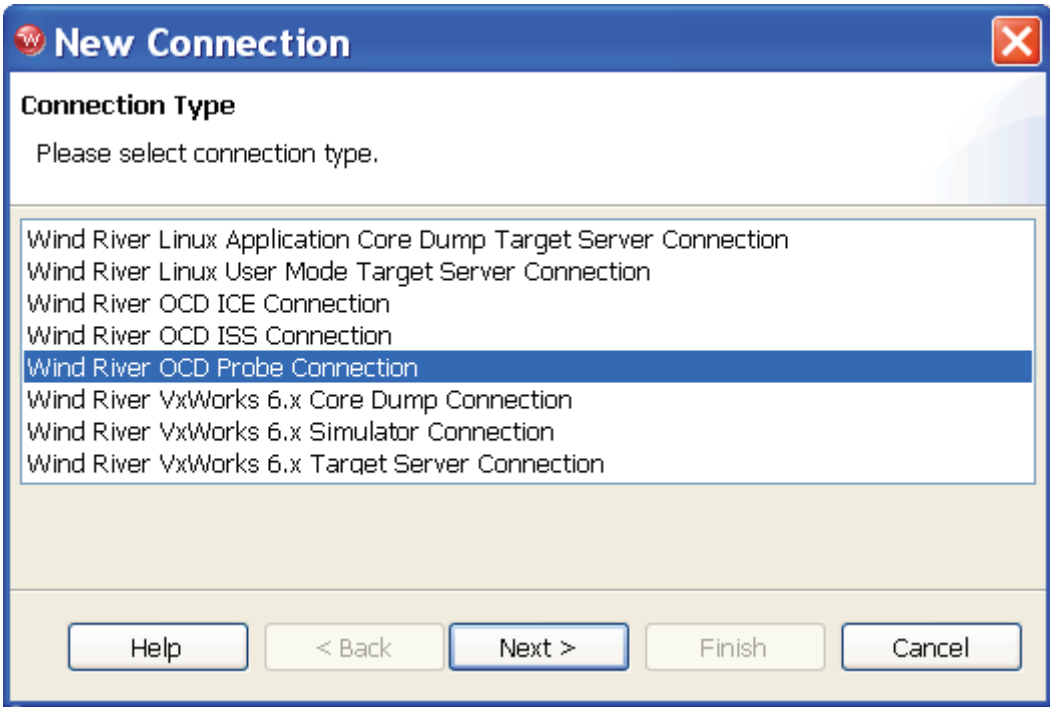

Figure 3: **OCD Probe Selection**

- Select the **Board File** radio button and then browse to the BRD file representing the ML403 board. This file, `ml403.brd`, is in the Wind River installation in the `<windriver_installation>\workbench-2.5\dfw\0145b\host\boardfiles\PowerPC\4xx\Xilinx` directory

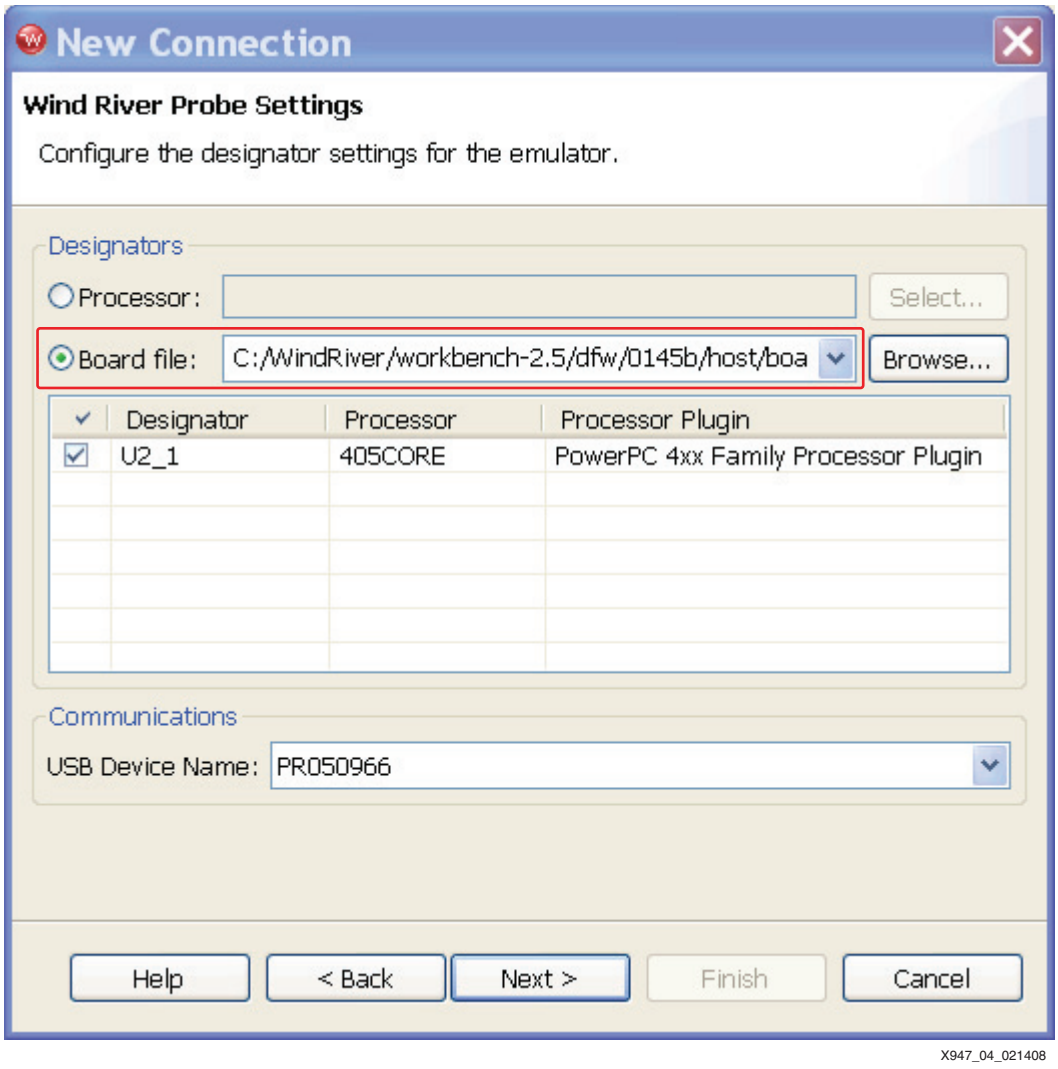

Figure 4: **New Connection Configuration**

- Click **Next** or **Finish**. The tools connect to the board. The status of **Connected** is shown next to the target connection in the Target Manager pane. If the connection is not automatically invoked, right-click the connection, then select **Connect**.

9. Download the VxWorks kernel image to the board.
   - In the hierarchical tree next to the target connection, a processor instance is shown. Right-click on this line, then choose **Reset and Download.**
   - In the Reset tab, ensure that the **Reset** option checked.
   - In the Download tab, click **Add Files**, then browse to the `<edk_project>\windriver_workspace\ml403_vxworks_image\default\VxWorks` file within the project. This is the kernel image that was previously created.

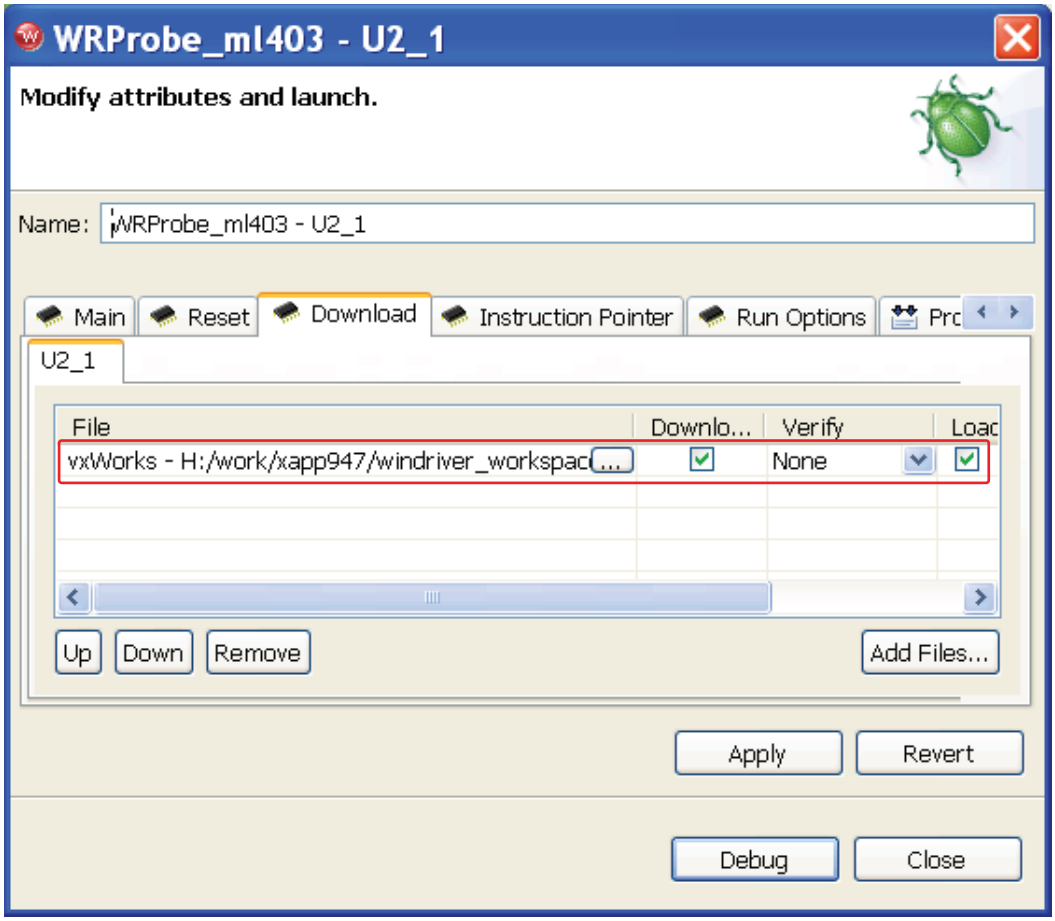

Figure 5: **Download File Selection**

- In the Run Options tab, select the **Run to End of Program**, then click **Apply**.

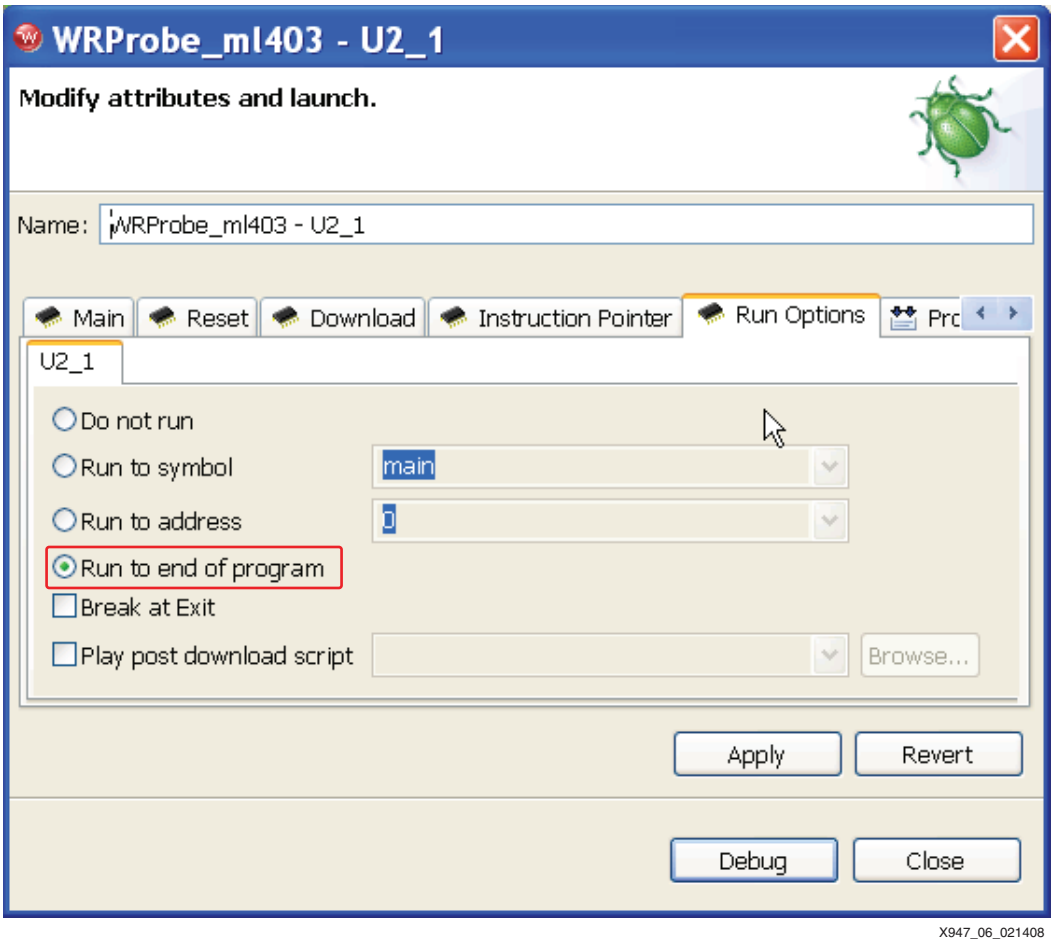

Figure 6:

- Click the **Debug** button. The VxWorks image is downloaded to the target, then run on the processor. The VxWorks banner appears on the UART terminal. A working VxWorks image that is suitable for use on the ML403 board has been created.

10. Right click on the connection entry in the Target Manager pane, then click **Disconnect** to terminate the Wind River debug connection.

Create a VxWorks Boot Loader

A VxWorks bootloader is a special build of VxWorks used to load the general VxWorks kernel at boot time over the network. Use the the sequence below to generate a bootloader image.

1. In the Wind River Workbench Project Navigator pane on the left of the screen, right click on **New** → **VxWorks Boot loader Project**.
   - Assign the project name, for example, `ml403_vxworks_bootloader`, then click **Next**.
   - Choose the `modified_vxworks_bsp` Board Support Package which was created earlier.
   - Choose the image style of **Uncompressed** from the drop-down list.
   - Choose the format of **Bin** from the drop-down list.
   - Click **Finish.**
2. Right-click the newly created project, then choose **Build Project** from the menu options.

The boot loader is built and a “Build Finished……” is displayed the bottom of the screen in the Build Console pane.

Program the Boot Loader into the Flash

The bootloader is most useful when programmed into flash. Use the following sequence to place the bootloader into the on board flash:

1. Wind River Workbench includes utilities for programming flash devices. To access this functionality, reconnect to the target board using the Wind River debug cable.
2. Right-click on the processor instance and choose the **Attach to Core** option.
3. From the Workbench menus, click **Window** → **Open Perspective** → **On-Chip Debug**.
4. From the Workbench menus, click **Window** → **Show View** → **Flash Programmer**.
5. At the bottom of the screen, you will see the Flash Programmer pane. Click the **Configuration View** tab, and choose the **Intel** → **28F320Jx** → **2048 x 16** → **2 Devices** option. This represents the part number, quantity, and configuration of the flash devices on the ML403 board.

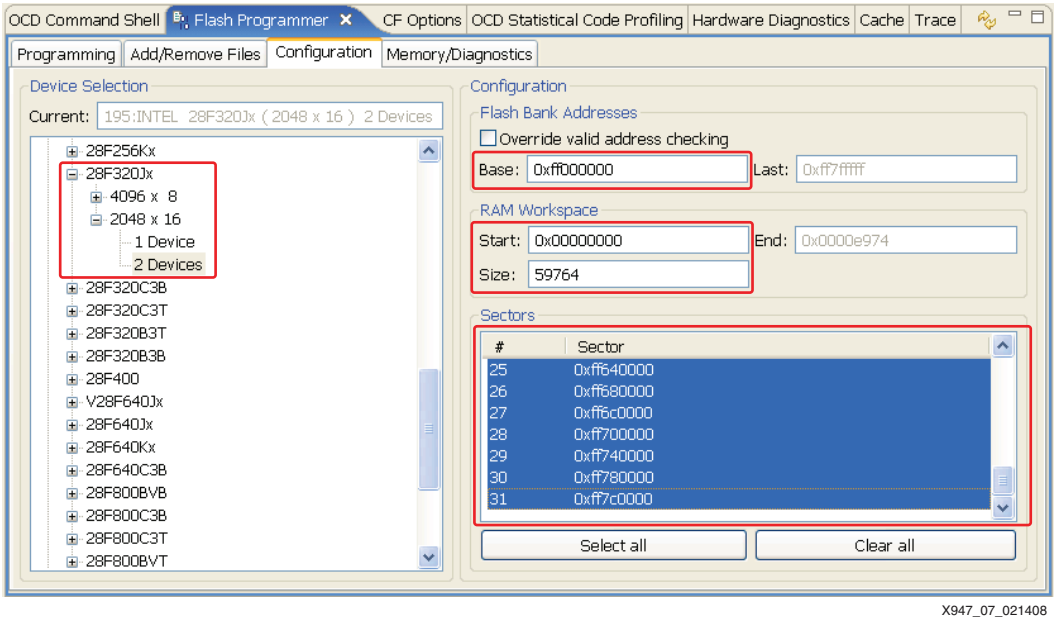

Figure 7: **Flash Configuration**

6. Enter the base address of `0xFF000000` in the Flash Bank Address box. The high address should be automatically completed by the tool (`0xFF7FFFFF`).
7. In the RAM Workspace box, ensure that the value is set to the base address of the DDR SDRAM (`0x00000000`).
8. Set the RAM Workspace Size to **60000**. The tool will automatically alter this figure slightly to place it onto a 2^n boundary.
9. Check that the 0-31 sectors of the flash in the Sectors window are accessible, then click the **Select All** button.
10. Select the Add/Remove Files tab. On the right of the window, click the **Add File** button, then browse to the `ml403_vxworks_bootloader\bootrom_uncmp.bin` file within the `windriver_workspace` area. After choosing this file, check the box for **This is a raw binary file**, then Click **OK.**

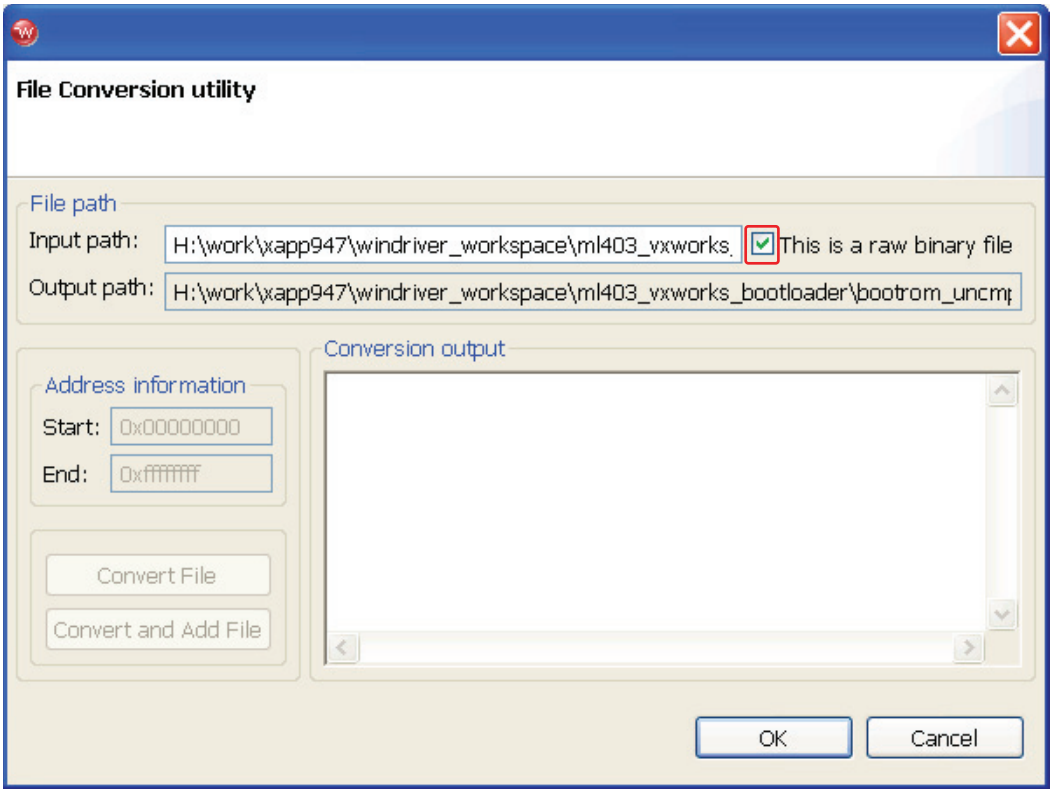

Figure 8: **File Conversion Utility**

11. Click on the start address field for the file that has just been added to the list. The start address value should be at the base address of the flash plus `0x100` bytes. In this example, the value is `0xFF000100`.

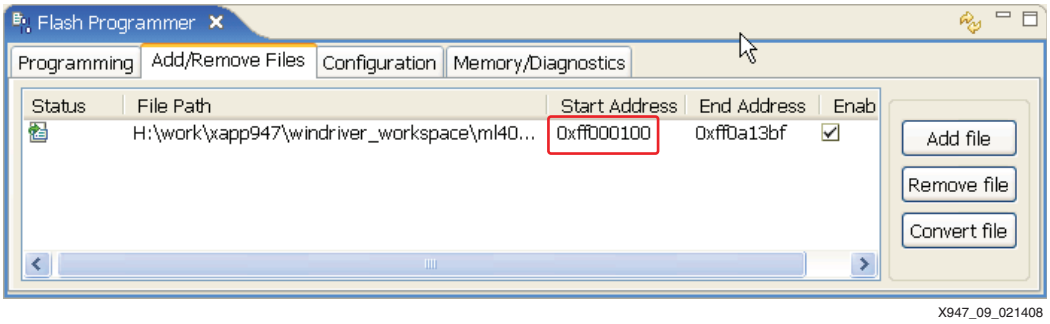

Figure 9: **Location in the flash to program the PROM file**

12. Note of the End address generated by the tool; this indicates the size of the file to be programmed into the flash.
13. Select the Programming View tab. Check that the Erase Sectors boxes indicate that the flash will be erased at least the size noted in the previous step.
14. In the Flash Programming section of this tab, check the **Send "IN" before each operation** box.
15. Select the Programming tab, then select **Erase/Program**. The flash is now programmed with the `bootrom_uncmp.bin` image.

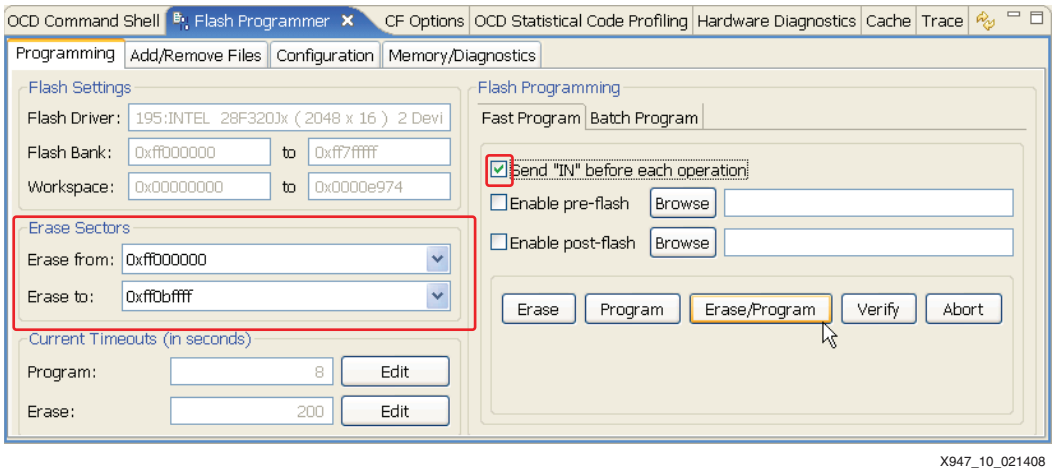

Figure 10: **Program the Flash**

16. Right click on the active debug connection in the Target Manager pane, then select **Disconnect** to terminate the connection to the board.
17. Switch off the power to the ML403 board, then disconnect the Wind River debug cable.

Configure the PPC Block RAM Application to Access the Flash

Included with this application note is the **VxWorks_Start** application. This application is built with the Xilinx standalone library BSP. A linker script is used to place the application within BRAM, which is the memory from which the processor will boot in the included system. VxWorks_Start performs one simple function - it will execute at boot time and branch to the location of the bootloader in flash.

Note that the `#define flash_address 0xFF000100` line in the source code reflects the address where the boot loader image was programmed into the flash. In the provided example, this is `0xFF000100`.

The reference system included with this application note provides a prepared file `download-vxworks_start.bit` in the `ready_for_download` directory. To generate a new file, follow these steps:

1. In XPS, choose **Software** → **Software Platform Settings**:
   a. Change the OS selection, **ppc405_0**, changed to **standalone** in **Software Platform**.
   b. Change the STDIN and STDOUT settings in OS and Libraries to **RS232_Uart**.
2. Select **Software** → **Generate Libraries and BSPs** to generate the standalone Board Support Package.
3. Right-click on the **VxWorks_Start** project and choose **Mark to Initialize BRAMs**
4. Only VxWorks_Start may be seleced to initialize BRAMs. Righ-click on the **ppc405_0_bootloop** project and de-select **Mark to Initialize BRAMs**
5. Generate a new `download.bit` file with **Device Configuration** → **Update Bitstream**. This is created in the `implementation` directory.

Test the Boot Loader

To test the boot loader, use the steps below.

1. Re-connect the Xilinx cable and power on the board.
2. Download the new bitstream which includes VxWorks_Start in BRAM to the ML403 by using **Device Configuration** → **Download Bitstream** from the Xilinx Platform Studio tool.
3. A status line is printed in the UART terminal window to indicate that the VxWorks_Start project is running, followed by a pause of a few seconds. The Wind River boot loader application then runs and begins an automated countdown.
4. Press any key in the UART terminal window to abort the countdown. The "`[VxWorks Boot]:`" prompt is seen.

Note: If the auto-boot is not terminated, the system will display the error message, *Error loading file:*.

Configure the Boot Loader

The boot loader is designed to fetch the main VxWorks kernel image across the network from an FTP server. As part of the BSP, the target board is configured with the IP address of 192.168.0.2. The host is configured with an appropriate IP address, such as 192.168.0.1. An FTP server is configured on the host, and the boot loader application is directed to access the desired VxWorks image.

1. Configure the IP address of the host PC as 192.168.0.1.
2. On the host PC, choose **Start** → **All Programs** → **Wind River** → **WorkBench 2.5** → **FTP Server** from the Start menu.
3. In the FTP Server application, select **Security** → **Users/Rights** from the menu.
4. Click **New User** and enter the name, **`my_ftp_user`**.
5. Enter **`pass`** in both of the password boxes.
6. In the Home Directory box, enter the path to the VxWorks kernel image which will be at `<edk_project_dir>\windriver_workspace\ml403_vxworks_image\default`, then click **Done**.
7. In the FTP Server application, select **Logging** → **Log options** from the menu. Check all of the boxes except **Windsock calls**, then click **OK**.
8. In the UART terminal, type **`p`** and press the Enter key to display the current boot loader settings. The settings are similar to those shown below.

```
[VxWorks Boot]: p

boot device             : lltemac          NAME OF ETHERNET MAC
unit number             : 0                ID OF ETHERNET MAC
processor number        : 0
host name               : host             NOT USED
file name               : VxWorks          NAME OF VXWORKS IMAGE
inet on ethernet (e)    : 192.168.0.2      IP ADDRESS OF TARGET BOARD
host inet (h)           : 192.168.0.1      IP ADDRESS OF HOST PC
user (u)                : xemhost          FTP USER NAME
flags (f)               : 0x0              NOT USED

[VxWorks Boot]:
```

Some of these parameters must be adjusted. Type **`c`** and press **Enter** in the UART terminal window; each option is presented individually. To leave the setting at the default value, press **Enter** in the UART terminal window. To change the setting, type in the new value.

9. Change the settings to match the ones shown in italics in the text below (only the user name and password must be changed.

```
boot device             : lltemac0
processor number        : 0
```

```
host name               : host
file name               : vxWorks
inet on ethernet (e)    : 192.168.0.2
inet on backplane (b)   :
host inet (h)           : 192.168.0.1
gateway inet (g)        :
user (u)                : my_ftp_user
ftp password (pw)       : pass
flags (f)               : 0x0
target name (tn)        :
startup script (s)      :
other (o)               :
```

Note: These settings will not be saved. VxWorks does not supply a software driver for the flash devices on the ML403 board.

Boot Load the Design

1. Type **@** at the `[VxWorks Boot]` prompt to boot the design.
2. The boot loader connects to the board, downloads the image via FTP, then runs the VxWorks kernel. Monitor the UART terminal window and the FTP server log during this time. The bootload process can be observed at each stage.
3. Check that the VxWorks kernel is alive by typing **i** at the prompt. The currently running tasks are listed. Type **version** at the prompt to display the version information.

```
boot device          : lltemac
unit number          : 0
processor number     : 0
host name            : host
file name            : vxWorks
inet on ethernet (e) : 192.168.0.2
host inet (h)        : 192.168.0.1
user (u)             : my_ftp_user
ftp password (pw)    : pass
flags (f)            : 0x0

Attaching interface lo0... done
lltemac: warning - MII clock is defaulted to 1000 Mbps
lltemac: Buffer information:
        0x00018810 bytes allocated for all buffers
         1564 byte cluster size
           60 Rx buffers, 1st buffer located at 0x00D00A20
            4 Tx buffers, 1st buffer located at 0x00D179A0
        0x00001030 bytes allocated for all BDs
           32 RxBDs, BD space begins at 0x00D1BD40
           32 TxBDs, BD space begins at 0x00D1C540
Attached IPv4 interface to lltemac unit 0
Loading... 1259648
Starting at 0x10000...

Attaching interface lo0... done
lltemac: warning - MII clock is defaulted to 1000 Mbps
lltemac: Buffer information:
        0x00018808 bytes allocated for all buffers
         1564 byte cluster size
           60 Rx buffers, 1st buffer located at 0x005BC5C0
            4 Tx buffers, 1st buffer located at 0x005D3540
        0x00001008 bytes allocated for all BDs
           32 RxBDs, BD space begins at 0x005D7900
           32 TxBDs, BD space begins at 0x005D8100
Attached IPv4 interface to lltemac unit 0

Adding 4439 symbols for standalone.


 ]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]
```

```
 ]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]
 ]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]
      ]]]]]]]]]]]  ]]]]     ]]]]]]]]]]       ]]              ]]]]         (R)
 ]     ]]]]]]]]]  ]]]]]]     ]]]]]]]]       ]]               ]]]]
 ]]     ]]]]]]]  ]]]]]]]]     ]]]]]] ]     ]]                ]]]]
 ]]]     ]]]]] ]    ]]]  ]     ]]]] ]]]   ]]]]]]]]]  ]]]] ]] ]]]]  ]]   ]]]]]
 ]]]]     ]]]  ]]    ]  ]]]     ]] ]]]]] ]]]]]]   ]] ]]]]]]] ]]]] ]]   ]]]]
 ]]]]]     ]  ]]]]     ]]]]]      ]]]]]]]] ]]]]   ]] ]]]]    ]]]]]]]    ]]]]
 ]]]]]]      ]]]]]     ]]]]]]    ]  ]]]]]  ]]]]   ]] ]]]]    ]]]]]]]]    ]]]]
 ]]]]]]]    ]]]]]  ]    ]]]]]]  ]    ]]]   ]]]]   ]] ]]]]    ]]]] ]]]]    ]]]]
 ]]]]]]]]  ]]]]]  ]]]    ]]]]]]]      ]     ]]]]]]]  ]]]]    ]]]]  ]]]] ]]]]]
 ]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]
 ]]]]]]]]]]]]]]]]]]]]]]]]]]]]]       Development System
 ]]]]]]]]]]]]]]]]]]]]]]]]]]]]
 ]]]]]]]]]]]]]]]]]]]]]]]]]]]       VxWorks 6.3
 ]]]]]]]]]]]]]]]]]]]]]]]]]]       KERNEL: WIND version 2.9
 ]]]]]]]]]]]]]]]]]]]]]]]]]       Copyright Wind River Systems, Inc., 1984-2006

 CPU: Xilinx Virtex-4 FX PPC405.  Processor #0.
 Memory Size: 0x4000000.  BSP version 1.2/0.
 Created: Jan 21 2008, 13:36:47
 ED&R Policy Mode: Deployed
 WDB Comm Type: WDB_COMM_END
 WDB: Ready.

-> i

  NAME         ENTRY       TID    PRI   STATUS      PC       SP     ERRNO  DELAY
---------- ------------ -------- --- ---------- -------- -------- ------- -----
tJobTask    98860         5352f0   0 PEND          e7850   5351c0       0     0
tExcTask    97a34         140c80   0 PEND          e7850   142f00       0     0
tLogTask    logTask       538598   0 PEND          e59a8   538470       0     0
tNbioLog    99ecc         53be20   0 PEND          e7850   53bd10       0     0
tShell0     shellTask     5f04e8   1 READY         eee0c   5f01d0       0     0
tWdbTask    wdbTask       5dbec8   3 PEND          e7850   5dbdc0       0     0
tNetTask    netTask       540d60  50 READY         e7850   540c80       0     0
value = 0 = 0x0
->
```

Make the Board Stand-alone Bootable

Throughout this application note, the FPGA has been configured manually. For stand-alone functionality, the Xilinx Flash PROM on the board is programmed with the bitstream, which contains not only the hardware configuration, but also the VxWorks_Start software (which is stored in the block RAMs). To do this, the Xilinx iMPACT tool is used to program the PROM.

1. Run the Xilinx iMPACT tool.
2. Choose to prepare a PROM file by double clicking PROM File Formatter.
3. The location for the PROM files to be generated and a file name are chosen. `<edk_project_dir>\VxWorks_Start.mcs` is used. Click **Next**.

Note: A previously generated file `<edk_project_dir>\ready_for_download\VxWorks_Start.mcs` is provided.

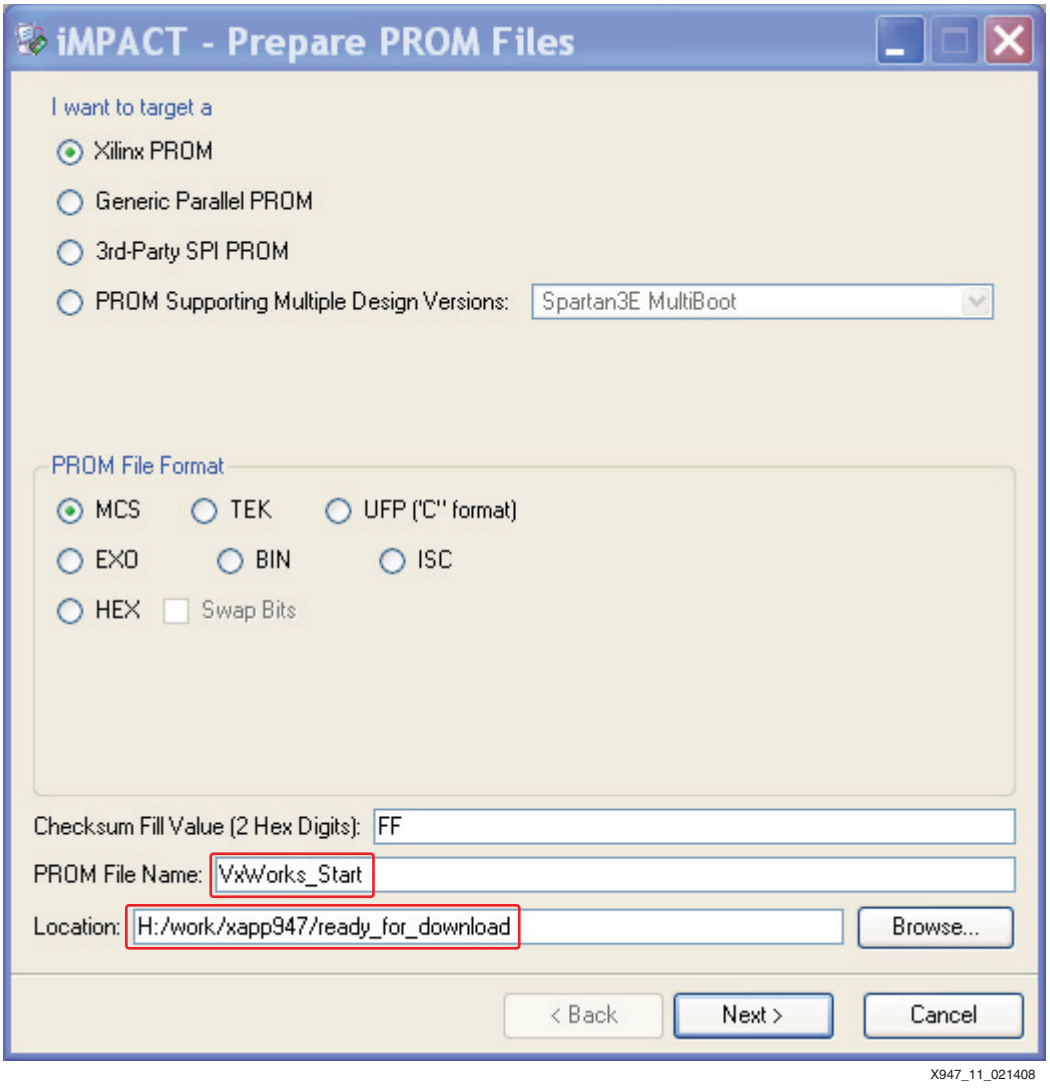

Figure 11: **Prepare PROM Files**

4. Select **I am using a Xilinx PROM in Serial Mode**, as shown in Figure 12, then click **Next**.

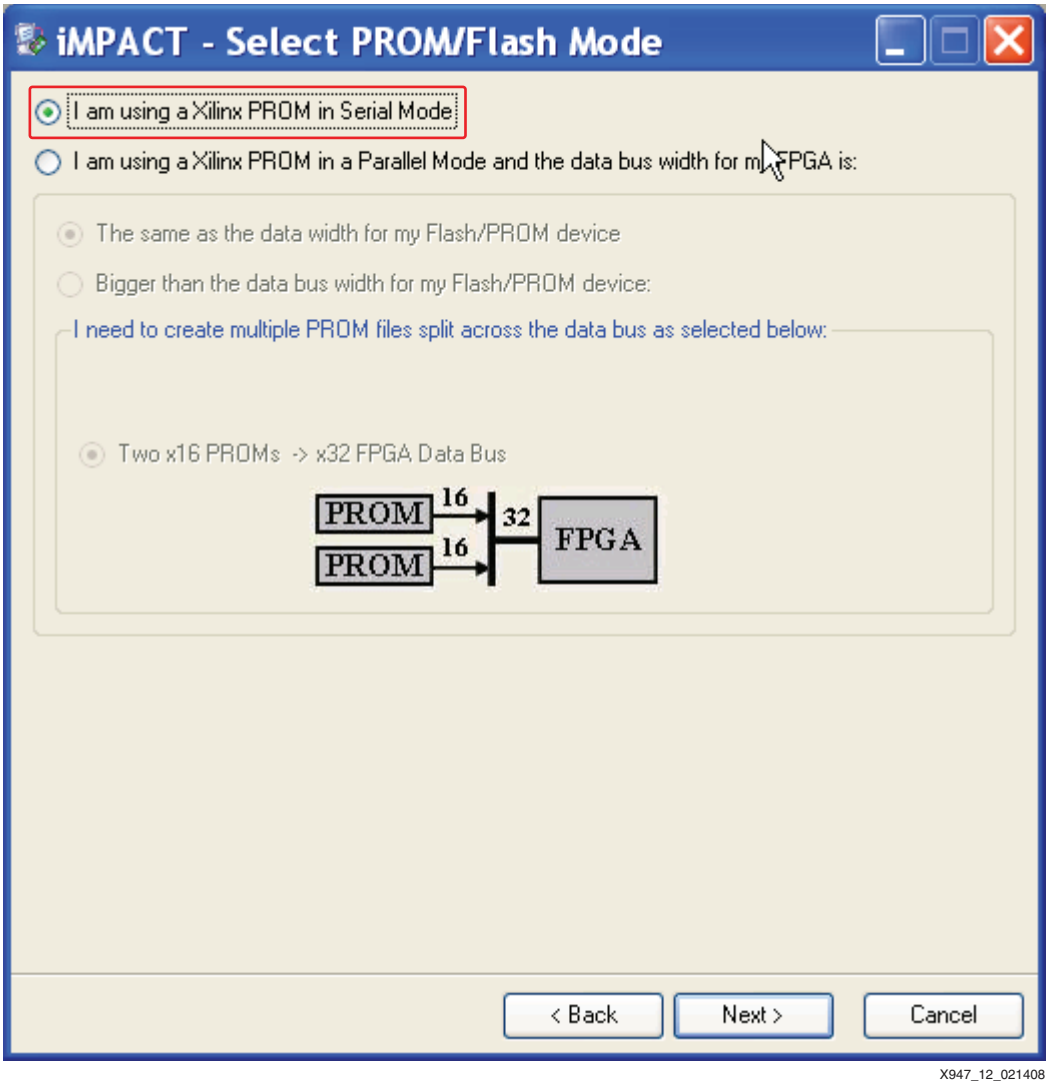

Figure 12: **Select PROM Flash Mode**

5. Choose the **XCF32p** PROM from the list, then click **Add**. The PROM will appear in the list as shown in Figure 13. Click **Next**.

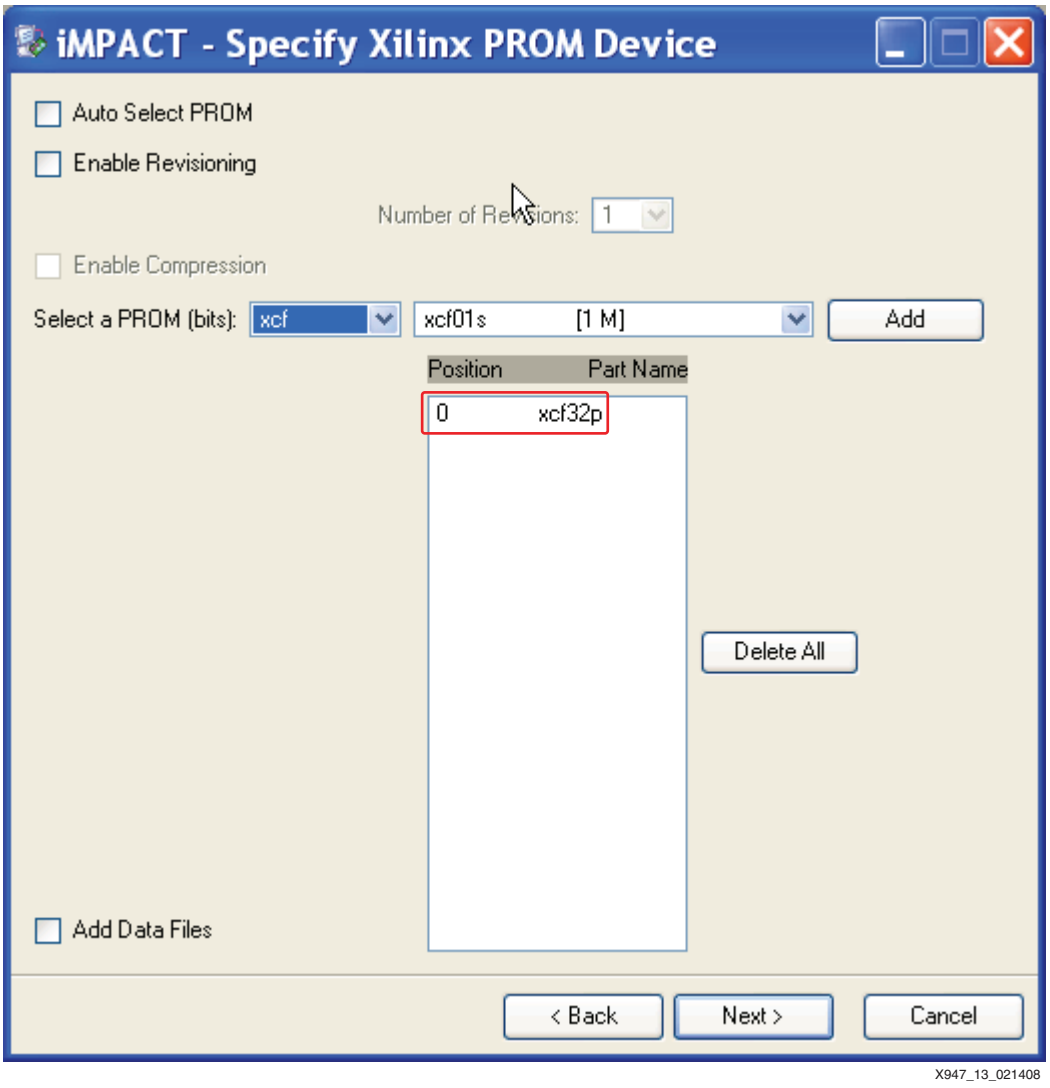

Figure 13: **Specify Xilinx PROM Device**

6. Impact will prompt for the file to use. If the user has generated a `download.bit` file as previously instructed, `<edk_project_dir>\implementation\download.bit` is used. A previously generated bitstream file, `<edk_project_dir>\ready_for_download\download-vxworks_start.bit` is provided as an alternative.
7. If a warning about the setting of the CCLK parameter is displayed, click **OK**.
8. Choose **No** to specify that no other design files are to be added to the PROM.
9. Double-click **Generate File** to generate the PROM file.

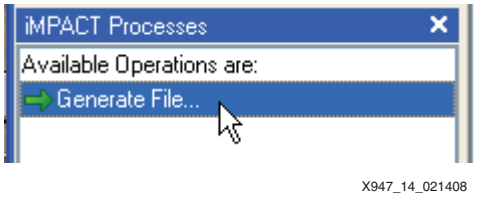

Figure 14: **Generate the .mcs PROM file**

10. Double-click the **Boundary Scan** icon. Right-click on **Initialize Chain** as shown in Figure 15. The chain of JTAG devices is automatically detected.

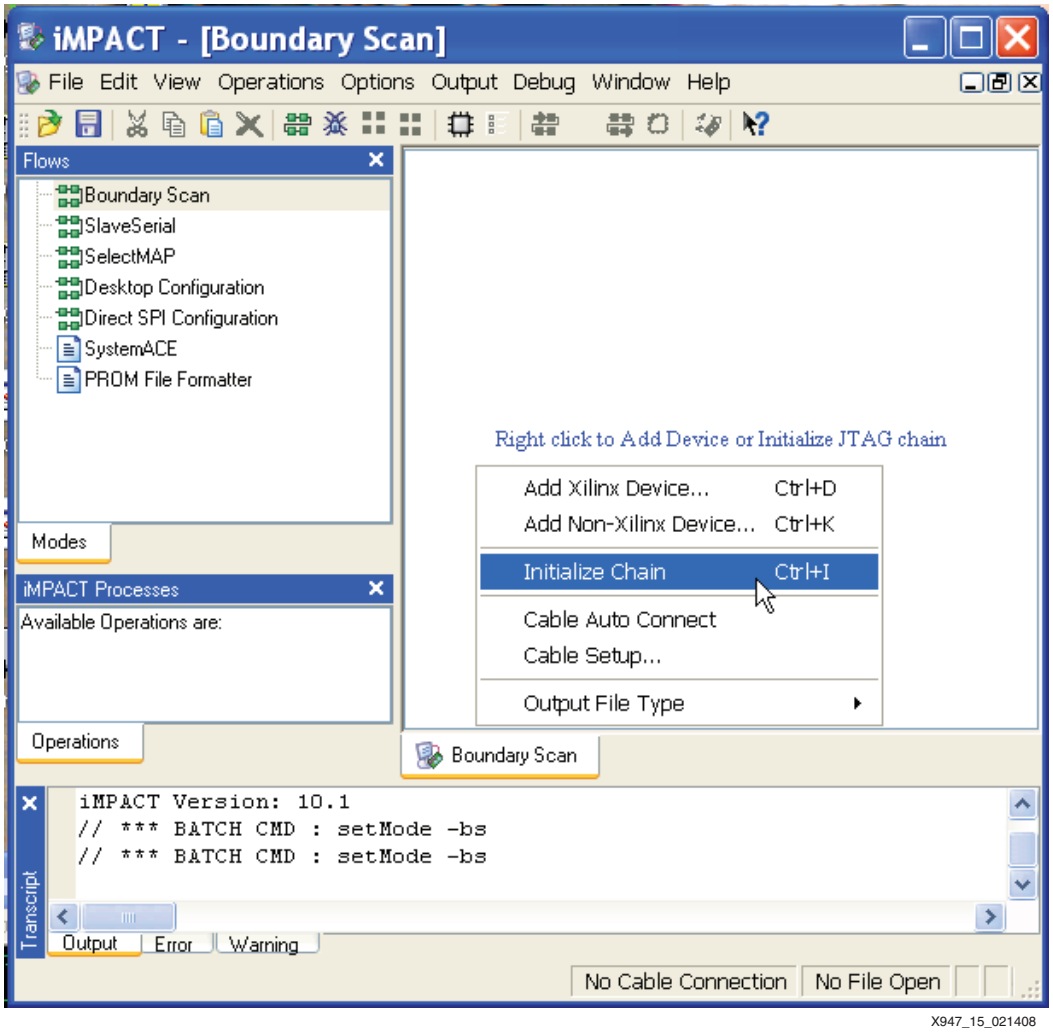

Figure 15: **Initialize JTAG Chain**

11. The **Add New Configuration** File dialog is now displayed for each device on the JTAG chain. The present device icon will appear green. Select **BYPASS** for all devices except the XCF32P.
12. The file previously prepared by the user `<edk_project_dir>\VxWorks_Start.mcs` is chosen for the **XCF32P PROM**.

Note: A previously generated file `<edk_project_dir>\ready_for_download\VxWorks_Start.mcs` is also available for use.

13. The Device Programming Properties dialog is displayed. As shown in Figure 16, select **Verify**, **Erase Before Programming**, and **Load FPGA** for the XCD32P device.

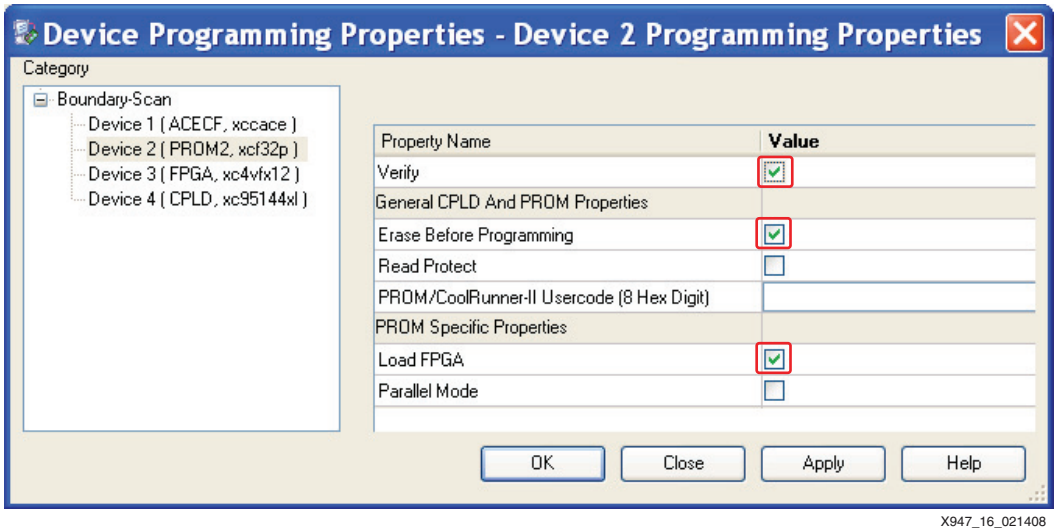

Figure 16: **Device Programming Properties**

14. The **XCF32P PROM** icon is selected. As shown in Figure 17, use **Operations** → **Program** in the iMPACT window to program the device.

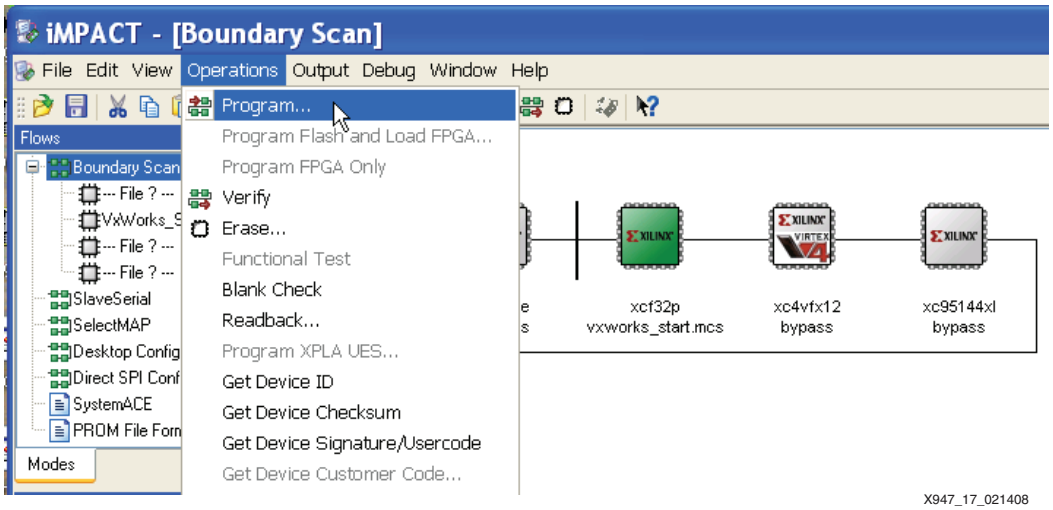

Figure 17: **Program the XCF32P PROM**

15. Close iMPACT.
16. Remove power from the ML403 board.
17. Disconnect the JTAG cable from the board.
18. Move the configuration switch SW12 at the edge of the board to the **Plat Flash** setting.
19. Power up the board, and verify that it auto-boots to the Wind River boot loader screen.

Configure a Target Server

The board has been configured to provide Ethernet connectivity. This presents the opportunity to debug the kernel, and any applications running on it, via the network. This functionality is especially useful for use in the software development team, because it removes the necessity for a Wind River probe to be attached to the host development machine. All software applications that need to be run and debugged on the target can be transferred via Ethernet, meaning that the target board need not be geographically local to the software developers.

1. From the Wind River Workbench "Target Manager", right-click in the pane, then select **New** → **Connection**.
2. Select **Wind River VxWorks 6.x Target Server Connection** from the list, then click **Next**.
3. Check that the setting in the Backend field is configured for **wdbrpc**.
4. Enter the IP address of the board in the Target name / IP Address field as shown in Figure 18. In the provided example, this is 192.168.0.2
5. In the **Kernel Image** section, select **File** and then browse to the `<edk_project>/windriver_workspace\ml403_vxworks_image\default\vxWorks` file. Click **Next**.

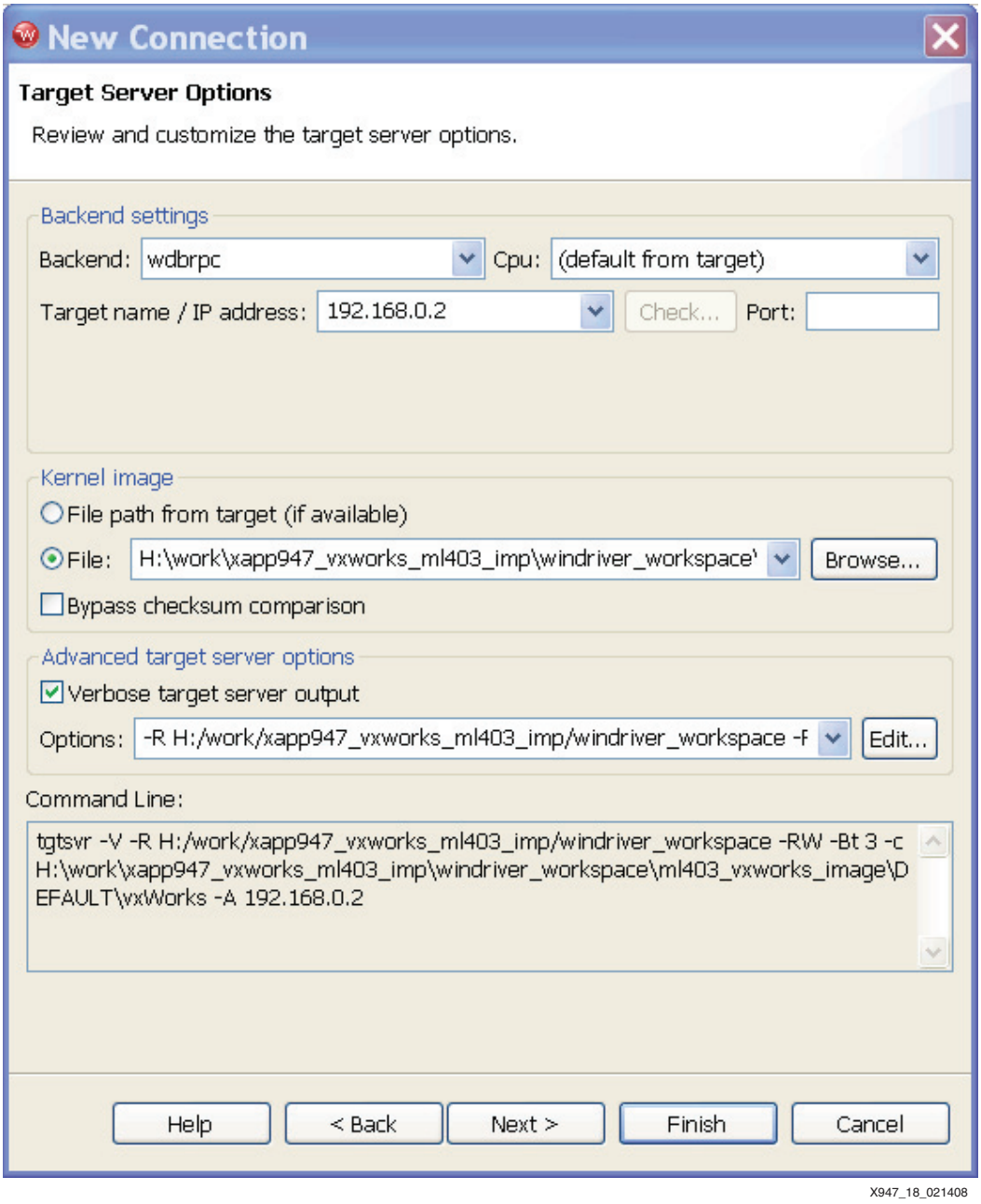

Figure 18: **New Target Server Connection**

6. Ensure that the **Load module symbols…..** box is checked, then click **Next**.
7. Click **Finish** to accept the settings. The connection is automatically started via Ethernet and a `tgt_192.168.0.2` entry appears in the Target Manager window.
8. In the Target Manager window, expand the **Kernel Tasks** tree to view the tasks that are currently running on the kernel.

Kernel Tasks

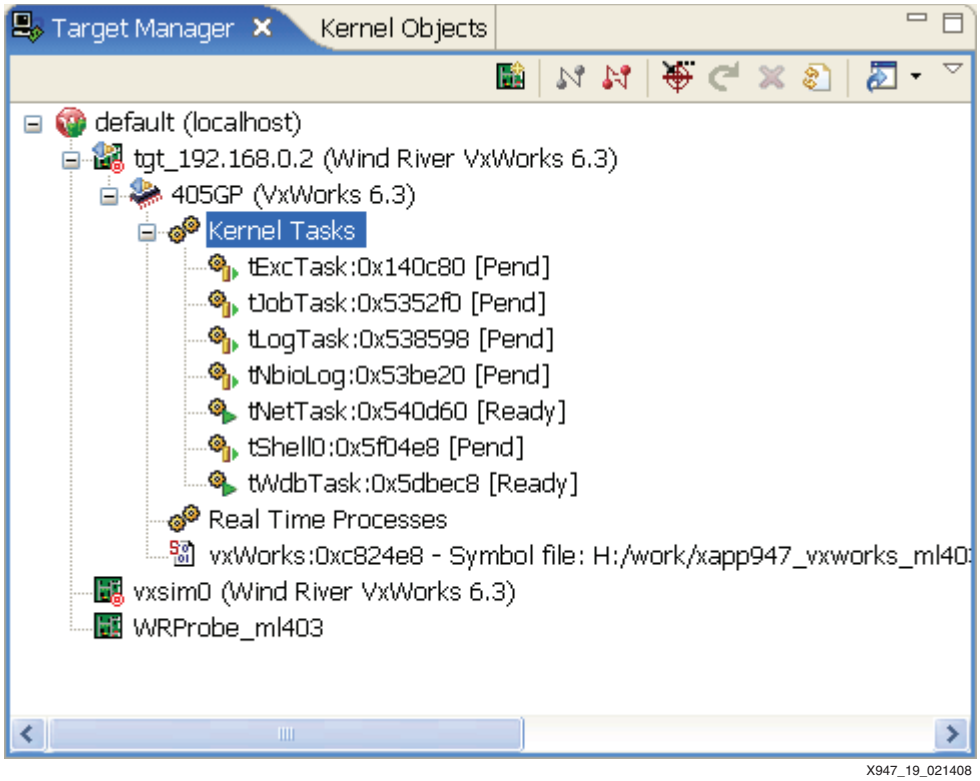

Figure 19: **Kernel Tasks**

References

1. UG080 *ML401/ML402/ML403 Evaluation Platform*
2. XAPP548 *Getting Started With EDK and Wind River VxWorks*

Revision History

The following table shows the revision history for this document:

Date	Version	Description of Revisions
10/05/06	1.0	Initial Xilinx release.
4/3/08	1.1	Updated to EDK 10.1

Notice of Disclaimer