ML40x Getting Started Tutorial

For ML401/ML402/ML403/ML405 Evaluation Platforms

UG083 (v5.0) June 30, 2006

XILINX®

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
11/22/04	1.0	Initial Xilinx release.
03/04/05	2.0	Renamed title from *ML401 Getting Started Tutorial* user guide to *ML40x Getting Started Tutorial* user guide. Expanded document from ML401-specific to include ML401, ML402, and ML403 evaluation platforms. Added the following sections: "ChipScope Pro Tools (ML403)" "DSP48 (ML403)" "QNX (ML403/ML405)" "Web Server (Using Hard Embedded Tri-Mode Ethernet MAC - ML403/ML405)" Minor edits to text and figures.
09/26/05	3.0	Updated the tutorial to be compatible with EDK 7. Minor edits to text for clarity.
02/24/06	4.0	Updated the tutorial to be compatible with EDK 8.1. Revised text and figures in"My Own Platform Flash Image (ML401/ML403/ML405)," page 34.
06/30/06	5.0	Expanded to include ML405 evaluation platform.

Table of Contents

About This Guide

The *ML40x Getting Started Tutorial* provides step-by-step instructions for setting up and using the ML40x evaluation platform (the board). The ML40x board comes with a number of pre-installed demonstrations. This tutorial guides you through these demonstrations and provides instructions to run them on the ML401, ML402, ML403, and ML405 evaluation platforms.

Guide Contents

This guide contains one chapter:

Additional Resources

To search the database of silicon and software questions and answers, or to create a technical support case in WebCase, see the Xilinx website at: http://www.xilinx.com/support.

Conventions

This document uses the following conventions. An example illustrates each convention.

Typographical

The following typographical conventions are used in this document:

Convention	Meaning or Use	Example
`Courier font`	Messages, prompts, and program files that the system displays	`speed grade: - 100`
`Courier bold`	Literal commands that you enter in a syntactical statement	**`ngdbuild`** *design_name*
Helvetica bold	Commands that you select from a menu	**File → Open**
	Keyboard shortcuts	**Ctrl+C**

Convention	Meaning or Use	Example
Italic font	Variables in a syntax statement for which you must supply values	**ngdbuild** *design_name*
	References to other manuals	See the *Development System Reference Guide* for more information.
	Emphasis in text	If a wire is drawn so that it overlaps the pin of a symbol, the two nets are *not* connected.
Square brackets [ ]	An optional entry or parameter. However, in bus specifications, such as **bus[7:0]**, they are required.	**ngdbuild** [*option_name*] *design_name*
Braces { }	A list of items from which you must choose one or more	**lowpwr ={on|off}**
Vertical bar |	Separates items in a list of choices	**lowpwr ={on|off}**
Vertical ellipsis . . .	Repetitive material that has been omitted	IOB #1: Name = QOUT' IOB #2: Name = CLKIN' . . .
Horizontal ellipsis ...	Repetitive material that has been omitted	**allow block** *block_name loc1 loc2 ... locn;*

Online Document

The following conventions are used in this document:

Convention	Meaning or Use	Example
Blue text	Cross-reference link to a location in the current document	See the section "Additional Resources" for details. Refer to "Title Formats" in Chapter 1 for details.
Red text	Cross-reference link to a location in another document	See Figure 2-5 in the *Virtex-4 User Guide.*
Blue, underlined text	Hyperlink to a website (URL)	Go to http://www.xilinx.com for the latest speed files.

ML40x Getting Started Tutorial

Overview

The ML40*x* evaluation platform (the board) comes with a number of pre-installed demonstration programs. This tutorial guides you through these demonstrations and provides instructions to run them on the ML401, ML402, ML403, and ML405 evaluation platforms. Differences between these boards are noted when necessary.

Some demonstrations interact with a PC or an external device. For these demonstrations, use a computer installed with:

- ISE software version 8.1i
- ChipScope™ Pro software version 8.1i

The following additional equipment is also recommended:

- VGA monitor and cable
- PC speaker with audio cable
- Ethernet port and an RJ-45 Ethernet cable
- USB keyboard (without a built-in USB hub)
- Null modem serial cable
- CompactFlash (CF) reader/writer for the PC

For current information about ML40*x* evaluation platforms, visit the corresponding Web page:

- ML401: http://www.xilinx.com/ml401
- ML402: http://www.xilinx.com/ml402
- ML403: http://www.xilinx.com/ml403
- ML405: http://www.xilinx.com/ml405

Board Setup

1. Position the ML40*x* board so the Virtex™-4 and Xilinx logos are oriented near the top edge of the board.
2. Make sure the power switch, located in the upper right corner of the board, is in the OFF position.
3. Locate the CF card slot (on the back side of the ML40*x* board), and carefully insert the System ACE™ CF card with its front label facing away from the board. Figure 1 shows the back side of the board with the CF card properly inserted.

   Note: The CF card provided with your board might differ.

   Caution! Be careful when inserting or removing the CF card from the slot. Do *not* force it.

Figure 1: **ML40*x* Evaluation Platform with CF Card**

Note: The ML405 board is similar but not identical to the example shown in Figure 1.

4. Connect the AC power cord to the power supply brick. Plug the power supply adapter cable into the ML40*x* board. Plug in the power supply to AC power.
5. Set the following switches:
   - Configuration address and mode DIP switch (6-position DIP switch) to `000111`
   - Configuration source selector switch (3-position slide switch) to SYS ACE

6. Connect a null modem serial cable between your PC and the ML40*x* board, and open a serial terminal program:
   - Select **Start** → **Programs** → **Accessories** → **Communications** → **HyperTerminal**
   - In the Connection Description window, type **9600** in the Name box, then click **OK**
   - In the Connect To window, click **Cancel**
   - In the 9600-HyperTerminal window, select **File** → **Properties**
     - Select the **Connect To** tab
     - Select **COM1** in the Connect using box (see Figure 2)
     - Click **Configure...**

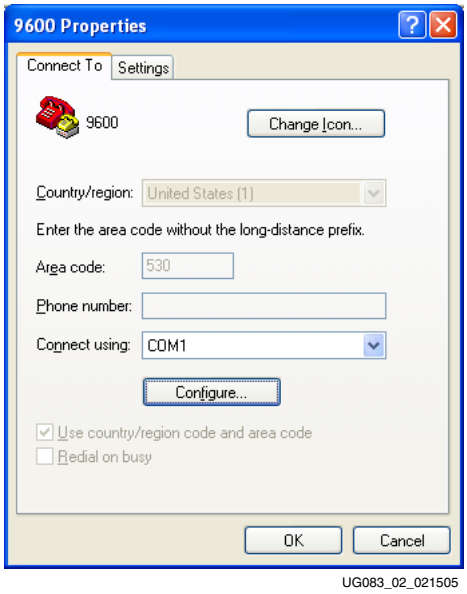

Figure 2: **HyperTerminal Setup and Properties**

Use the pull-down menu to set the COM1 properties (see Figure 3) to the following:

- Bits per second = 9600
- Data bits = 8
- Parity = None
- Stop bits = 1
- Flow control = None
- Click **OK** → **OK** to accept settings

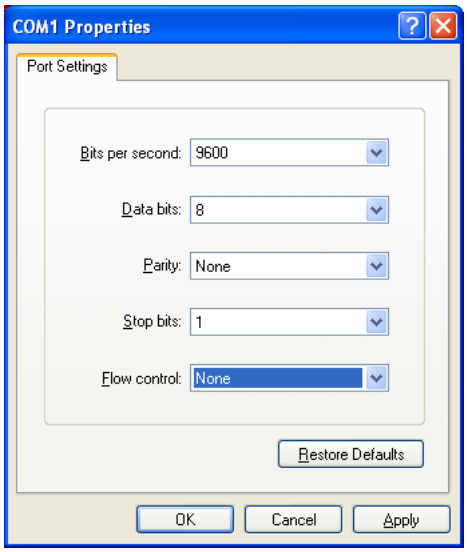

Figure 3: **COM1 Properties Setup**

7. Connect the VGA monitor to the board, if available.
8. Turn on the ML40*x* board's main power switch, and press the System ACE **RST** button. After the FPGA has been programmed, the LEDs in the lower left corner should be:
   - Bus Error 1 and 2 = off
   - FPGA INIT = green
   - FPGA DONE = green
   - System ACE "Err" = off
   - System ACE "Stat" = green

   Note: When the CF card is ejected or not installed, the System ACE "Err" LED blinks.
9. Extract the associated training lab files to your local PC.

   Unzip the training lab files to a working directory, name the directory, and make note of the directory's name. This directory with the extracted files is referred to as <*LAB_DIR*> in this tutorial.

ML40*x* Demonstrations in System ACE CF

Bootloader

Location

System ACE configuration address 0.

Description

The ML40*x* Bootloader demonstration displays a menu of demonstration designs that can be loaded by using the System ACE controller's reconfiguration feature. The menu is displayed on the serial terminal, LCD, and VGA.

To choose a demonstration, use the North-East-South-West navigation buttons (GPIO_SW_N, GPIO_SW_E…) on the board, then press the center button (GPIO_SW_C) to start the demonstration. Alternatively, you can select a demonstration by entering its number into the serial terminal. The demonstrations are:

- "Virtex-4 Slide Show," page 14
- "Web Server (Using Soft Ethernet MAC - ML401/ML402)," page 15
- "Web Server (Using Hard Embedded Tri-Mode Ethernet MAC - ML403/ML405)," page 18
- "DSP48 (ML401/ML402)," page 21
- "Linux (ML403/ML405)," page 22
- "ChipScope Pro Tools (ML401/ML402)," page 23
- "QNX (ML403/ML405)," page 24
- "USB," page 25
- "My Own ACE File," page 26
- "Restore CPLD/Flash Images," page 27

To return to the ML40*x* Bootloader at anytime, press the System ACE **RST** button. The bootloader runs only if the leftmost configuration address DIP switches are set to `000` and the configuration source selector switch (3-position slide switch) is set to SYS ACE.

Virtex-4 Slide Show

Location

System ACE configuration address 1.

From the Bootloader menu, select option **1** to start the Virtex-4 Slide Show demonstration.

Description

This demonstration displays a sequence of picture files stored on the CF card accompanied by audio playback of a music file stored on the CF card. Pressing the East/West (GPIO_SW_E/GPIO_SW_W) buttons on the board manually switches to the previous/next slide. The North/South (GPIO_SW_N/GPIO_SW_S) buttons on the board change the volume. The center button toggles between pausing and continuing the slide show.

Note: This demonstration requires a VGA monitor connected to the VGA port and a headphone or external speaker connected to the audio jacks.

In this program, the processor reads the CF file system through the System ACE MPU port and loads the audio/video data into DDR SDRAM. The processor then controls the flow of data to the VGA controller and audio controller connected to the internal CoreConnect bus.

How to Change or Customize the Slide Show

Setup

To change the side show, follow these instructions:

1. Place the picture files in the root directory.
2. Name the picture files `image<`*XX*`>.bmp` where <*XX*> is a numerical sequence starting from 01 and counting up.

   The program reads the picture files through the System ACE MPU interface starting from `image01.bmp` then counts upward. A maximum of 16 images can be read. The BMP files must be sized as 640x480 pixels with 24-bit color.
3. Give the sound file the name `sound.wav` and encode it as a 44.1 KHz, 16-bit stereo wave file (CD format). The sound file cannot be greater than 32 MB in size.

Note: When adding additional images or larger sound files, it might be necessary to use a higher capacity System ACE CF card than the one shipped with the ML40*x* board.

Try to add your own slides and music. For example, in Microsoft PowerPoint, you can export a presentation to HTML for a 640x480 screen. You can then convert the JPG or GIF slides to BMP format using Microsoft Photo Editor that is installed on many PCs. Rename the BMP files to `image<`*XX*`>.bmp` and copy to the System ACE CF card. Now you can run your own customized slide show. For audio, try to extract a song from a CD into a WAV file. Copy the WAV file into the System ACE CF card and name it `sound.wav`.

Web Server (Using Soft Ethernet MAC - ML401/ML402)

Note: For ML403 and ML405 boards, see "Web Server (Using Hard Embedded Tri-Mode Ethernet MAC - ML403/ML405)," page 18.

Location

System ACE configuration address 2.

From the Bootloader menu, select option **2** to start the Web Server demonstration.

Description

In this demonstration, an Ethernet-controlled GPIO interface application uses the ML40x board as a Web server. A remote host, such as a PC running a Web browser, can communicate with the ML40*x* board using the Ethernet to read the value of the ML40*x* board's DIP switches or to set the LEDs on the board. Refreshing or reloading the remote PC's Web browser causes the background color to change and the current DIP switch values to be re-read. By default, the IP address of the ML40*x* board is 1.2.3.4, but it can be changed by recompiling the software.

Setup

1. Connect an Ethernet cable (straight or crossover) from your host PC to the ML40*x* board.

   Note: The Ethernet PHY chip on the ML40*x* board has an auto-crossover feature.

2. Configure the remote PC host's IP address to 1.2.3.9 (Subnet mask can be 255.0.0.0).

   Note: Record the previous network settings so they will be easier to restore after the lab. The screen shots and icon names might be slightly different depending on your PC's operating system version.

   - Right click **My Network Places** on your PC, and select **Properties**
   - Right click **Local Area Connection,** and select **Properties**

- Select **Internet Protocol (TCP/IP),** and click **Properties** (see Figure 4)

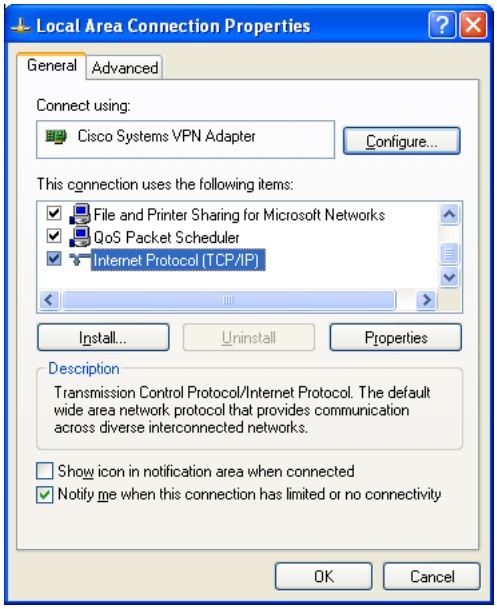

Figure 4: **Local Area Connection Properties Setup (ML401/ML402)**

- Select **Use the following IP address:** (see Figure 5)
- Enter this information: IP address = **1.2.3.9** and Subnet mask = **255.0.0.0**
- Click **OK** → **OK** to accept settings

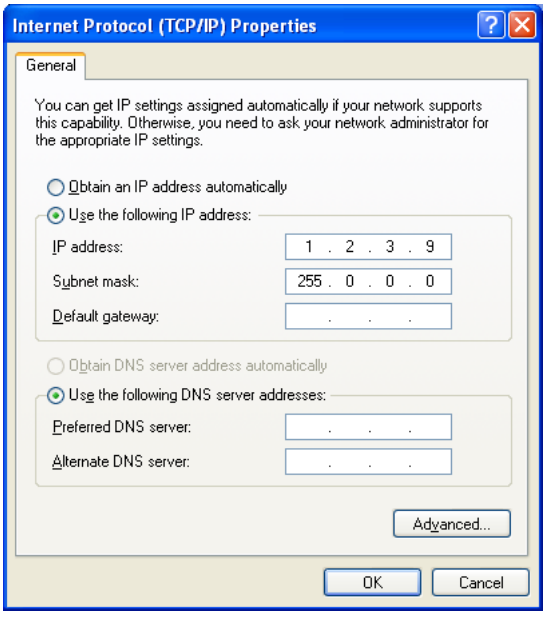

Figure 5: **IP Settings (ML401/ML402)**

3. Make sure the connection is running at 10 or 100 Mb/s and the ML40*x* board's link lights are on (the lights are located in the upper left corner of the board). The link LEDs labeled 10, 100, and 1000 indicate the link is established at that speed.
   - You might need to force your PC to link in 10 or 100 Mb/s (duplex) mode. If so, then:

     Right-click **Local Area Connection** → **Properties** → **Configure** → **Advanced tab** → **Speed**
4. On the remote PC host, open a Web browser connection to **http://1.2.3.4:8080**, and follow the instructions on the loaded Web page.
   - You might need to turn off your browser's proxy (use direct Internet connection mode) especially if you have multiple networking devices on your PC.
   - On the remote PC host, you can ping **1.2.3.4** to confirm that the network connection is alive.
5. Restore your PC's network settings when finished.

Web Server (Using Hard Embedded Tri-Mode Ethernet MAC - ML403/ML405)

Note: For the ML401 and ML402 boards, see "Web Server (Using Soft Ethernet MAC - ML401/ML402)," page 15.

Location

System ACE configuration address 2.

From the Bootloader menu, select option **2** to start the Web Server demonstration.

Description

This demonstration uses the embedded Virtex-4 Tri-Mode Ethernet MAC (TEMAC) to implement a Web server running Treck protocol on the PowerPC™ 405 processor. A remote host, such as a PC running a Web browser, can communicate with the ML40*x* board over Ethernet. The IP settings of the board can be obtained via DHCP or manually set by the user.

Setup

1. Connect an Ethernet straight or crossover cable from the ML40*x* board to a network jack or directly to your PC.

   Note: The Ethernet PHY chip on the ML40*x* board has an auto-crossover feature. This web server demonstration only supports full-duplex Ethernet connections.

2. When the Web server is started, it provides the option of a static IP address or a DHCP-obtained IP address. If your network supports DHCP, select DHCP (GPIO_SW_E button), and continue to Step 3. If you are using a static IP address for the board, select Static IP (GPIO_SW_W button), and follow the instructions below.

   The default static IP address is set to 192.168.0.101. To use this address, select *accept* (GPIO_SW_E button), otherwise select *change* (GPIO_SW_W button), and follow the instructions on the serial port to enter the new network settings.

   Note: If connecting the ML40*x* board directly to a stand-alone PC, it is recommend that the default static IP address be used.

3. After the IP address is determined, the web server STARTS UP. The IP address of the board is displayed on the LCD.

4. If necessary, configure your PC's network settings to be able to access the board over Ethernet. For example, if you connect the ML40*x* board directly to a stand-alone PC and have the board set to the default static IP address of 192.168.0.101, you should configure the PC to have the IP address 192.168.0.1 so it is on the same subnet.

   Note: You might want to record your previous network settings so they can be easily restored after the lab. The screen shots and icon names might be slightly different depending on your PC's operating system version.

   - Right click **My Network Places** on your PC, and select **Properties**
   - Right click **Local Area Connection**, and select **Properties**
   - Select **Internet Protocol (TCP/IP)**, and click **Properties** (Figure 6, page 19)

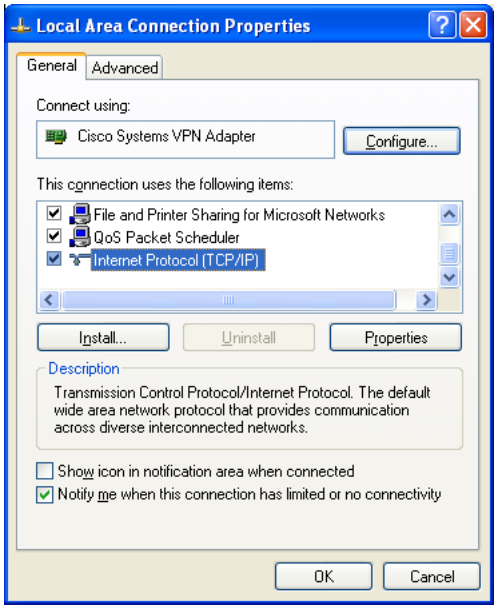

Figure 6: **Local Area Connection Properties Setup**

- Select **Use the following IP address**: (see Figure 7)
- Enter this information: IP address = **192.168.0.1** and Subnet mask = **255.0.0.0**
- Click **OK** → **OK** to accept settings

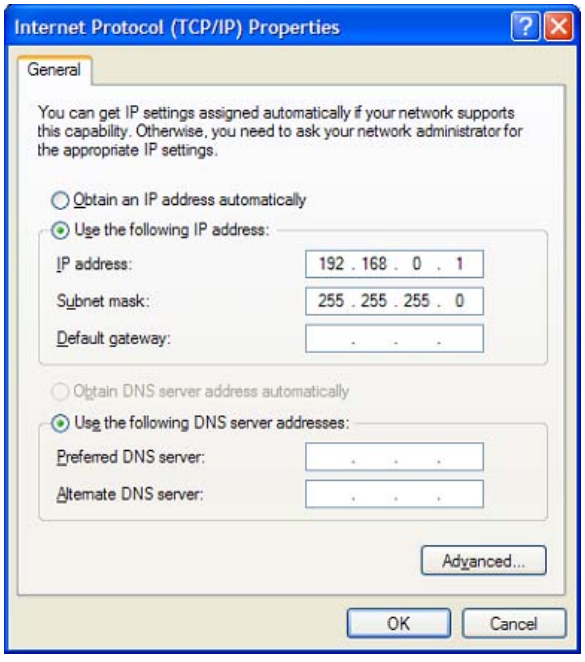

Figure 7: **IP Settings**

5. On the remote PC host, open a Web browser connection to **http://192.168.0.101**, and follow the instructions on the loaded Web page.
   - You might need to turn off your browser's proxy (use direct Internet connection mode) especially if you have multiple networking devices on your PC.
   - On the remote PC host, you can ping **192.168.0.101** to confirm that the network connection is alive.
6. Restore your PC's network settings if necessary when finished.

DSP48 (ML401/ML402)

Location

System ACE configuration address 3 (ML401/ML402).

From the Bootloader menu, select option **3** to start the DSP48 demonstration.

Note: For ML403 boards: Platform Flash configuration address 0. See "ML40x Demonstrations in Platform Flash," page 30.

Description

This demonstration highlights the use of the XtremeDSP™ slice (DSP48) in image processing. The design illustrates the two commonly used image processing functions: color space conversion and two-dimensional (2D) filtering. The demonstration divides the 640x480 pixel screen into 20 tiles, each of which has different parameters for both color conversion and 2D filtering. The filters shown are: pass-through, blur, edge enhancing, and edge detecting. The demonstration runs at 100 MHz with a 25-MHz pixel clock. This allows the DSP48s to be used in a 4:1 over-sampled mode to reduce the size of the design by 4:1. Both the color space converter and 2D filter are designed to meet timing and run at 450 MHz to meet HDTV's 75- to 112-MHz pixel clock requirements with a 4:1 over-sampled clock.

Color Space Conversion

Each output color is computed as $C = r*R_C + g*G_C + b*B_C + O_C$; where {r g b} are 8-bit input pixels and $\{R_C\ G_C\ B_C\ O_C\}$ are 18-bit color coefficients. The design uses three DSP48 blocks for color space conversion with each DSP48 block used as a multiply-accumulator to implement the above equation.

2D Filter

The 2D filter consists of an 8-tap vertical filter followed by an 8-tap horizontal filter. The color channels are processed independently using three DSP48s to implement each filter. Each of the three vertical filters uses four block RAMs for line storage.

In addition to the color space conversion and 2D filtering, the demonstration includes memory for the test patterns (24 block RAMs), a test pattern generator, a VGA timing generator, and circuitry to switch between coefficients dynamically. The entire design uses 21 DSP48s, 36 block RAMs, and 1,258 4-input LUTs.

Linux (ML403/ML405)

Location

ML403/ML405: System ACE configuration address 3.

From the Bootloader menu, select option **3** to start the Linux demonstration.

Description

This demonstration shows MontaVista Linux running on the PowerPC 405 processor. Linux includes support for the peripherals, such as PS/2 mouse and keyboard, VGA (X Window), 10/100 Ethernet, UART, and file system mounted on the CF card using the System ACE CF controller. Linux console input/out is available using the UART. On the VGA output, a web browser is displayed. The PS/2 mouse and keyboard can be used to interact with the X Window display. The user names and passwords are:

ML403:

username = root, password = 403ml (root account)

username = linux, password = ml403 (user account)

ML405:

username = root, password = 405ml (root account)

username = linux, password = ml405 (user account)

Note: Remember to logoff and shutdown using the GUI **shutdown** button or the **shutdown –h now** Linux command before turning off or resetting the board. This ensures the Linux file system is shut down correctly. The message `system halted` indicates the shutdown process has completed.

ChipScope Pro Tools (ML401/ML402)

Location

System ACE configuration address 4 (ML401/ML402).

From the Bootloader menu, select option **4** to start the ChipScope Pro tools demonstration.

Note: For ML403 boards: Platform Flash configuration address 1. See "ML40x Demonstrations in Platform Flash," page 30.

Description

This demonstration contains a loadable 32-bit binary counter that can be read and controlled using the ChipScope Pro Virtual I/O (VIO) feature. The LEDs show the upper bits of the counter. The inputs and outputs of the counter are also shown in the ChipScope Pro logic analyzer mode.

Setup

1. Connect the PC4 Cable from the PC to the ML40*x* board.
2. Open the ChipScope Analyzer.
3. Open the `ML40x_chipscope_demo.cpj` file using **File** → **Open Project**.
4. Establish a JTAG connection with the ChipScope Pro ILA core running on the ML40*x* board. Click the leftmost icon in the ChipScope Pro Analyzer icon bar: 
5. Look at the Virtual I/O (VIO) console. Select **Window** → **Console - DEV2: My…(VIO)**

   You should see 32_bit_counter_count_val changing as the 32-bit counter runs at 100 MHz. Click the **counter_load_enable** button (box to the right of the signal name) to force the counter to 0xEEE1234 where it resumes counting.
6. Look at the waveform window. Select **Window** → **Waveform - DEV:2 My…(ILA)**
7. Click the trigger immediate (T!) button. 
8. Use the magnifying glass icon  to zoom in to see the individual 32-bit counter values.

QNX (ML403/ML405)

Location

ML403/ML405: System ACE configuration address 4.

From the Bootloader menu, select option **4** to start the QNX demonstration.

Description

This demonstration shows the QNX operating system running on the PowerPC 405 processor. QNX includes support for the peripherals, such as PS/2 mouse, VGA (X Window), UART, and 10/100 Ethernet. QNX console input/out is available using the UART. The Othello game that opens and appears in the X Window display can be played by using the PS/2 mouse.

Note: Remember to shutdown using right-click → **Shutdown** before turning off or resetting the board. This ensures the file system is shut down correctly.

USB

Location

System ACE configuration address 5.

From the Bootloader menu, select option **5** to start the USB demonstration.

Description

This demonstration uses the processor and the USB controller chip on the ML40*x* board to communicate with a USB keyboard.

The program functions by first reading the file `demo.bin` from the CompactFlash card. This file contains the software for the internal microprocessor inside the USB controller (Cypress CY7C76300). The FPGA's processor reads this file and writes the data to the memory inside the USB controller through its HPI port. The USB controller's internal processor then starts and can begin implementing low-level USB commands to communicate with the USB keyboard. Data from the USB keyboard is transferred to the FPGA's processor using mailbox registers over the HPI port.

Setup

Connect a standard USB keyboard to the ML40*x* board. Keys typed on the USB keyboard are then displayed on the character LCD and serial port.

Note: This demonstration requires a USB keyboard without a built-in hub.

My Own ACE File

Location

System ACE configuration address 6.

From the Bootloader menu, select option **6** to start the demonstration of the *My own ACE file* program.

Description

This program is a placeholder design to be replaced by a user design.

Setup

Take a bitstream and make your own ACE file:

1. Open a DOS command shell. Click **Start Menu** → **Run**, then enter **`cmd`** as the program to run, and click **OK**.
2. Change directory to your lab directory. Type **`cd <LAB_DIR>`**.

   In this directory, there is a bitstream called `ml40x_char_lcd_hw.bit`. This program displays the text ML40*x* on the character LCD.
3. Convert this bitstream to an ACE file.

   Type:

   ```
   ml40x_bit2ace ml40x_char_lcd_hw.bit ml40x_char_lcd_hw.ace
   ```

   This creates the `ml40x_char_lcd_hw.ace` file.
4. Carefully remove the System ACE CF card from the ML40*x* board (preferably with the power off). Open the CF card on your PC. This requires either a PC card adapter or a USB CompactFlash reader (not included with ML40*x* board, but available at PC stores).
5. Copy `ml40x_char_lcd_hw.ace` to the CF card into the **ml40x/myace** directory.
6. In the **ml40x/myace** directory on the CF card, rename `system_my_ace.ace` to `system_my_ace.bak`. This ensures that there is only one ACE file in this directory.

   Note: On ML402, there might be insufficient space on the CF card to keep the old ACE file. It might need to be backed up outside the CF card.
7. Eject the CF card from your PC (right-click on the CF card in Windows Explorer and click **Eject**). This shuts down the CF card to prevent data corruption. Carefully reinsert the CF card into the ML40*x* board (preferably with the power off).
8. Turn the power back on, if necessary, and press the System ACE RST button to restart the Bootloader. Select option 6 to start the *My own ACE file* program. You should now see `ML40x` displayed on the character LCD.

Restore CPLD/Flash Images

Location

System ACE configuration address 7.

From the Bootloader menu, select option **7** to start the demonstration of the *Restore CPLD/Flash Images* program.

Description

This program reads up to eight bitstreams from the CF card using the System ACE MPU port and then programs linear flash. This allows the CPLD to read the linear flash and program the FPGA with the bitstream specified by the configuration address DIP switches.

Setup

The directory and name of the bitstreams to be loaded are specified by the `flash.txt` file in the XILINX directory of the CF card.

Note: You can edit `XILINX\flash.txt` on the CF card to specify different directory locations or file names for the bitstreams that are to be loaded into the linear flash.

While running, this program uses the serial terminal and LCD to display status information about its progress in programming the linear flash. This lab continues in "ML40x Demonstrations in Linear Flash," page 28.

ML40*x* Demonstrations in Linear Flash

Description

This demonstration shows the FPGA being configured by an external linear flash device and a CPLD. This method of download is used in some embedded processor systems where it is necessary to keep software and bitstream data in one non-volatile device.

Setup

1. After the bitstreams are programmed into linear flash, shown in "Restore CPLD/Flash Images," page 27, you can slide the configuration source selector switch (3-position slide switch) to CPLD Flash.

   Optional: You can change the three leftmost configuration address DIP switches to different binary values between 0 and 7 to load in a different bitstream from the linear flash memory.

   Note: The ML402 board supports only four bitstreams in the linear flash memory.
2. Press the **Prog** button to load bitstream 0 into the FPGA. The serial terminal displays a message describing how the bitstream was loaded. The LCD also displays a message indicating that the design was loaded from linear flash.
3. Now try to program a different bitstream into linear flash. Copy the `hello_char_lcd_hw.bit` bitstream from <*LAB_DIR*> to the CF card into the XILINX\flash directory. Rename `system0.bit` to `system0.bak` as backup. Then rename the `hello_char_lcd_hw.bit` file to `system0.bit`.

   Note: On the ML402 board, there might be insufficient space on the CF card to keep the old ACE file. It might need to be backed up outside the CF card.
4. Set the configuration address and mode DIP switch (6-position DIP switch) back to `000111`.
5. Set the configuration source selector switch (3-position slide switch) back to SYS ACE and press the **System ACE RST** button.
6. From the Bootloader menu, select option **7** again to start the *Restore CPLD/Flash Images* program.
7. After the new bitstream is programmed into linear flash, slide the configuration source selector switch (3-position slide switch) to CPLD Flash again.
8. Press the **Prog** button to load bitstream 0 into the FPGA. The `hello` message displayed on the character LCD indicates the new bitstream is stored in flash and loaded through the CPLD.

Figure 8 shows step 1 of the linear flash configuration process when the bitstreams are uploaded. In this step, the CPU reads `*.bit` files from the CompactFlash and writes them into linear flash. Up to eight different configurations can be stored in the flash.

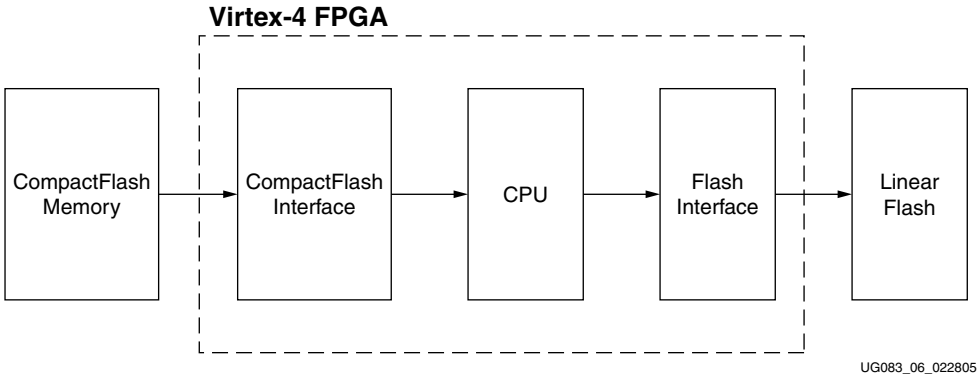

Figure 8: **Linear Flash Configuration Process: Step 1**

Figure 9 shows step 2 of the configuration process, where the FPGA is configured. In this step, the CPLD (95144XL) reads the flash and configures the FPGA. DIP switches on the ML40*x* board select which bitstream to download. The FPGA is configured in slave-serial mode.

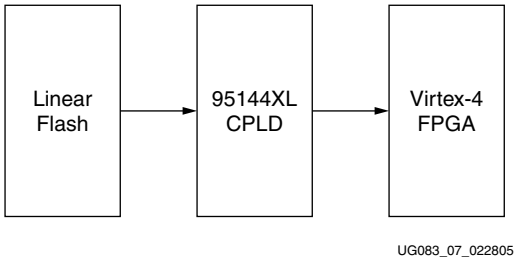

Figure 9: **Linear Flash Configuration Process: Step 2**

ML40*x* Demonstrations in Platform Flash

The Platform Flash PROM contains advanced features, such as revision control, and is a convenient and easy-to-use method of configuring FPGAs.

The Platform Flash on the ML40*x* board can hold multiple bitstreams:

Setup

1. Set the configuration address and mode DIP switch (6-position DIP switch) to `000111`.
2. Set the configuration source selector switch (3-position slide switch) to Plat Flash.

Menu of Contents (ML401/ML405)

Location

ML401/ML405: Platform Flash configuration address 0.

ML402/ML403: Not present.

Description

This program displays a menu listing the demonstrations stored in the Platform Flash. The information is displayed on the serial port, character LCD, and VGA display.

Setup

1. Set the three leftmost configuration address DIP switches to the binary value 001.
2. Press the **Prog** button to see a menu of demonstration programs stored on the Platform Flash and how to access them. The information is presented on the serial terminal, character LCD, and VGA screen.

DSP48 (ML403)

Location

ML403: Platform Flash configuration address 0.

ML401/ML402/ML405: Not present.

Description

The demonstration highlights the use of the XtremeDSP slice (also known as the DSP48) in image processing. See "DSP48 (ML401/ML402)," page 21 under "ML40x Demonstrations in System ACE CF," page 13, for more information.

Setup

1. Set the three leftmost configuration address DIP switches to the binary value `001`.
2. Press the **Prog** button to run the demonstration.

Game Demonstration (ML401/ML402/ML405)

Location

ML401/ML405: Platform Flash configuration address 1.

ML402: Platform Flash configuration address 0.

ML403: Not present.

Description

This demonstration implements the *Simon* game using the North-East-South-West-Center directional buttons (GPIO_SW_N, GPIO_SW_E…) and LEDs on the board. Messages are displayed on the character LCD.

Setup

1. Set the three leftmost configuration address DIP switches to the binary value `001`.
2. Press the **Prog** button to run the demonstration.

Instructions

At the beginning of the game, all the LEDs blink rapidly.

1. Press any button to start the game.
2. In each round, the LEDs blink in a given sequence. The player must press the buttons to repeat the sequence corresponding to the order in which the LEDs blinked.
3. Correctly repeating the sequence gives the player one point, and a new round is started. Each sequence becomes increasingly complex.
4. If a mistake is made, the score resets to 0 on the screen, and all the LEDs blink rapidly to signify a new game.

ChipScope Pro Tools (ML403)

Location

ML403: Platform Flash configuration address 1.

ML401/ML402/ML405: Not present.

Description

This demonstration contains a loadable 32-bit binary counter that can be read and controlled using the ChipScope Pro Virtual I/O (VIO) feature.

See "ChipScope Pro Tools (ML401/ML402)," page 23 under "ML40x Demonstrations in System ACE CF," page 13, for more information.

Setup

1. Set the three leftmost configuration address DIP switches to the binary value `001`.
2. Press the **Prog** button to run the demonstration.

My Own Platform Flash Image (ML401/ML403/ML405)

Location

ML401/ML403/ML405: Platform Flash configuration address 2.

ML402: Not present. Refer to the "Setup" section for instructions to load your own design into the Platform Flash PROM.

Description

This exercise is a placeholder design that prompts you, through the serial port, LCD, and VGA, to *put your own design here*.

Setup

1. Set the three leftmost configuration address DIP switches to the binary value `010`.
2. Press the **Prog** button to run the demonstration.

   Caution! This exercise might overwrite all the contents of the Platform Flash.

Preparing PROM Files

Load your own bitstreams into the Platform Flash on ML40*x* board:

1. Open iMPACT: **Start** → **Programs** → **Xilinx ISE** → **Accessories** → **iMPACT**.
   - At start up, iMPACT asks for an iMPACT Project file.
     - Select **Create a new project…** and enter a project name and directory.
2. Under Welcome to iMPACT, select **Prepare a PROM File**, then click **Next**.
3. Under Prepare PROM Files (see Figure 10, page 35):
   - Select the **Xilinx PROM** radio button
   - Under PROM File Format, select the **EXO** radio button
   - In the PROM File Name box, enter **hello_files**
   - In the Location box, browse to or enter your <*LAB_DIR*> directory
   - Click **Next**

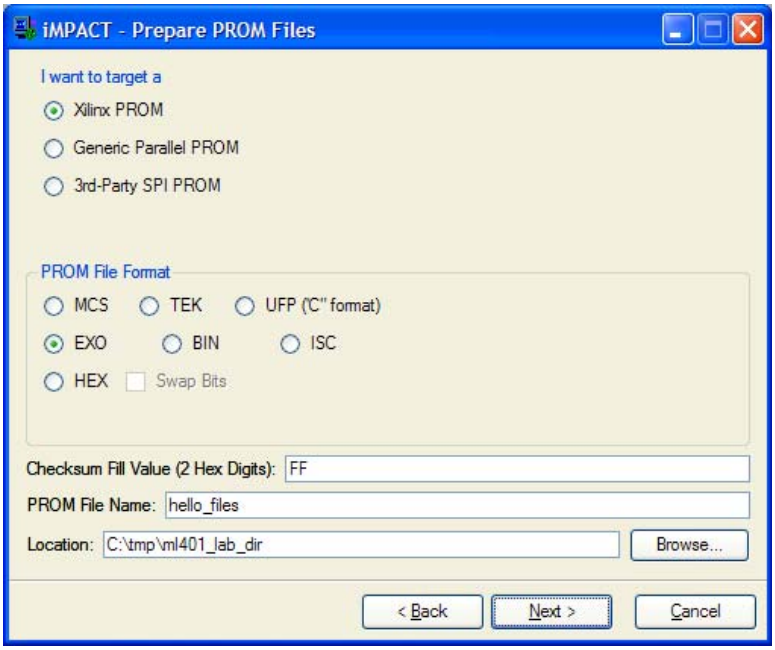

Figure 10: **Prepare PROM Files**

4. Under Specify Xilinx PROM Device (see "Specify Xilinx PROM Device"):
   - Check the **Enable Revisioning** box
   - From the Number of Revisions drop-down box, choose **4**
   - From the Select a PROM drop-down boxes, choose **xcfp** and **xcf32p**, then click **Add**

     Note: The ML402 board supports only two revisions due to the larger bitstream size.
   - Click **Next**

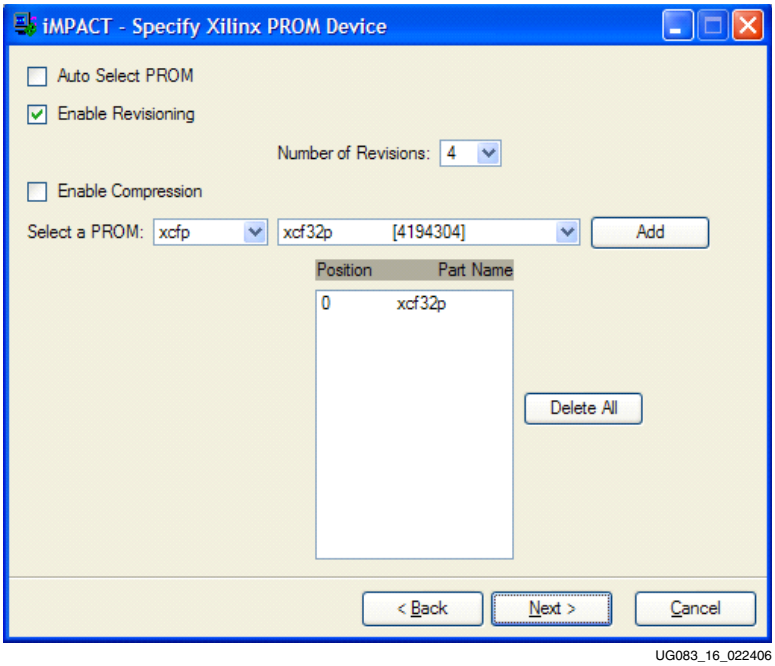

Figure 11: **Specify Xilinx PROM Device**

5. Under File Generation Summary, click **Finish**
6. Under **Add Device → Start adding device file to Revision: 0**, click **OK**
7. Under Add Device File, click **Add File**, then browse for `hello_char_lcd_hw0.bit` in your <*LAB_DIR*> directory, and click **Open**
8. Under **Add Device → Would you like to add…to Revision:0?**, click **No**
9. Under Add Device File, click **Add File**, then browse for `hello_char_lcd_hw1.bit` in your <*LAB_DIR*> directory, and click **Open**
10. Under **Add Device → Would you like to add…to Revision:1?**, click **No**

    Note: For the ML402 board, skip Steps 11-14 and go directly to Step 15.
11. Under Add Device File, click **Add File**, then browse for `hello_char_lcd_hw2.bit` in your <*LAB_DIR*> directory, and click **Open**
12. Under **Add Device → Would you like to add…to Revision:2?**, click **No**
13. Under Add Device File, click **Add File**, then browse for `hello_char_lcd_hw3.bit` in your <*LAB_DIR*> directory, and click **Open**
14. Under **Add Device → Would you like to add…to Revision:3?**, click **No**
15. Under **Add Device → You have completed … entry … Click 'OK' to continue**, click **OK**

16. Select Menu option: **Operations → Generate File…**
17. Under **Generate PROM File → Do you want to compress file?**, click **No**
18. After a pause, the `PROM File Generation Succeeded` message appears.

You have now created a PROM image file and are ready to program that image into the ML40*x* board.

Programming the PROM

For the following steps, make sure your ML40x board is powered on and a download cable is connected from your PC to the board.

1. Start iMPACT or **Select File → New** from the menu
   - At start up, iMPACT asks for an iMPACT Project file
     - Select **Create a new project…** and enter a project name and directory
2. Under Welcome to iMPACT, select **Configure devices using … JTAG**, then select **Automatically connect…chain**, then click **Finish**
3. Under Assign New Configuration File, click **Cancel All**
4. Double-click **xcf32p**
5. Under Assign New Configuration File, browse to your <*LAB_DIR*> directory
   - Change Files of type: to **EXO Files (*.exo)**
   - Select `hello_files.exo` by clicking it, then click **Open**
6. Right-click the **xcf32p** icon and select **Program…**
7. Click **Programming Properties** under the Category section. The default options should look like Figure 12:

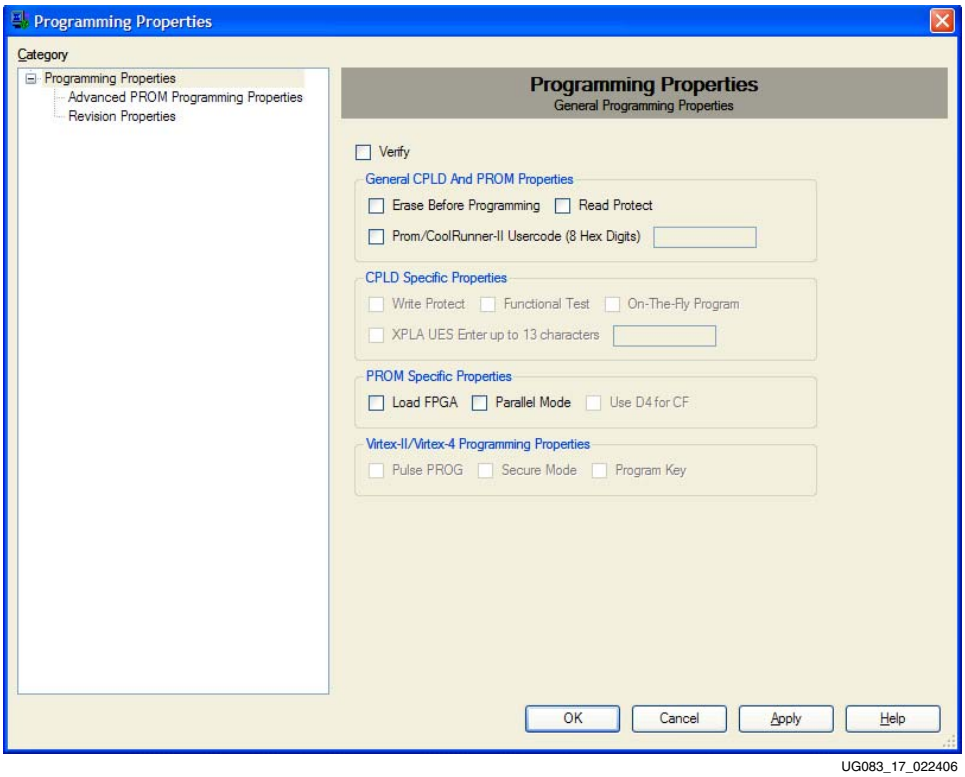

Figure 12: **Programming Properties**

8. Under **Programming Properties**, click **Advanced PROM Programming Properties** under the Category section
   - Under **During Configuration**, select **PROM is Configuration Master…**
     - Inside the sub box, select the **Internal Clock** radio button, and set the clock frequency to 40 MHz
     - Your options should look like Figure 13:

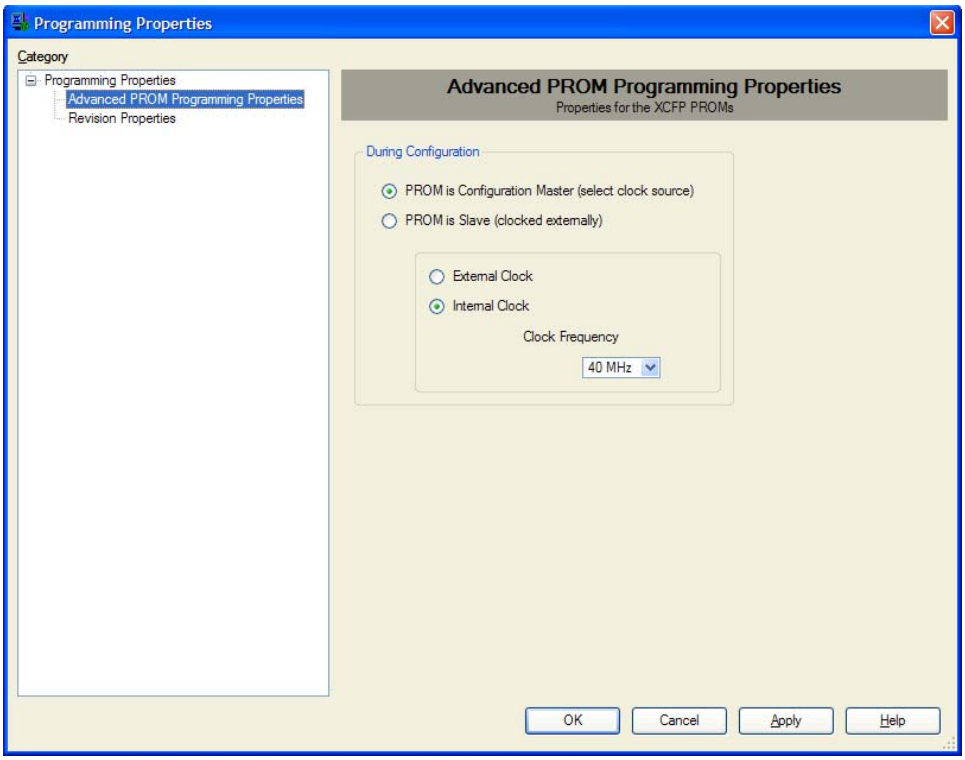

Figure 13: **Advanced PROM Programming Properties**

9. Under Programming Properties, click **Revision Properties** under the **Category** section
   - In the Design Revision column, select the **Rev 2** box

     Note: For the ML402 board, select the **Rev 0** box.
     - Check the corresponding box in the **Erase Column**
     - Your options should look like Figure 14:
10. Click **OK**

    This programs the Rev 2 bitstream (Rev 0 for the ML402 board) into the PROM using master serial mode for the Platform Flash, which is slave serial mode for the FPGA. The mode DIP switches on the board are set to 111 = Slave Serial.

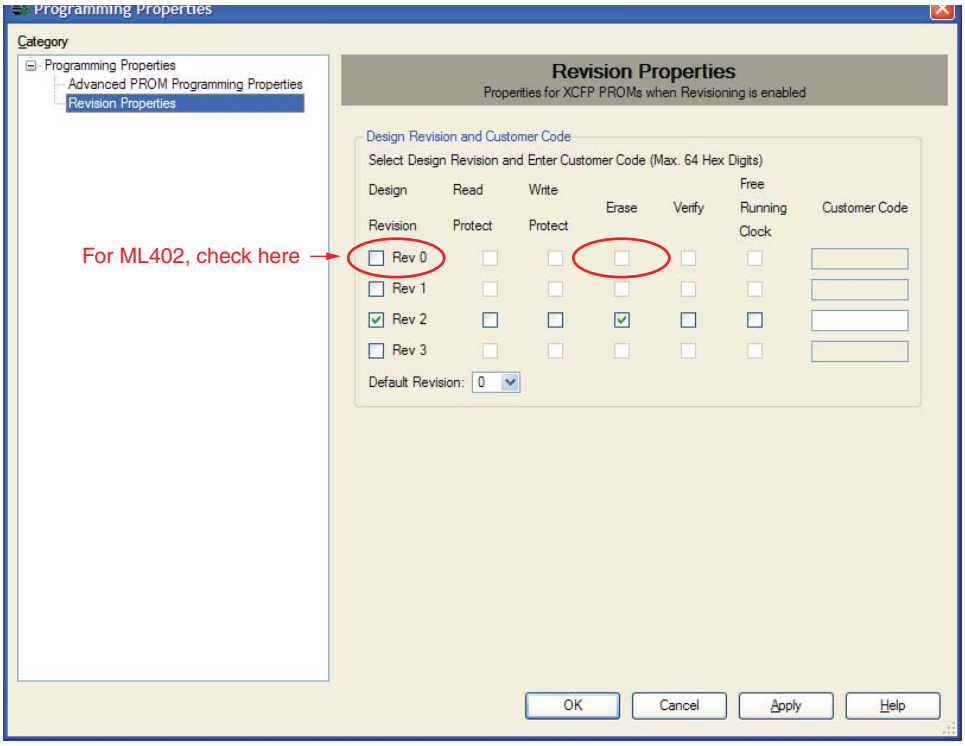

Figure 14: **Revision Properties**

11. When the PROM is finished programming, press the **Prog** button on the board. Revision 2 (Rev 0 on ML402) on the Platform Flash now loads a design that displays `Hello2` (`Hello0` on ML402) on the character LCD.

    Note: To restore the Platform Flash to its original contents, repeat these steps using the master image from the ML40*x* Web page:
    - ML401: http://www.xilinx.com/ml401
    - ML402: http://www.xilinx.com/ml402
    - ML403: http://www.xilinx.com/ml403
    - ML405: http://www.xilinx.com/ml405

    Select the **Reference Designs** link on the left-hand navigation menu.

XROM (ML401/ML402/ML403/ML405)

Location

ML401/ML403/ML405: Platform Flash configuration address 3.

ML402: Platform Flash configuration address 1.

Description

The XROM program presents a menu over the serial port offering various diagnostic tests of the ML40*x* board features.

Setup

1. Set the three leftmost configuration address DIP switches to the binary value `011`.

   Note: For the ML402 board, set the DIP switches to 001.

2. Press the **Prog** button to run this demonstration.
3. Select from the menu presented on the serial terminal to run various diagnostic tests.