XAPP957 (v1.1) October 8, 2008

Virtex-5 Embedded Tri-Mode Ethernet MAC Hardware Demonstration Platform

Summary

This application note describes a system using the Virtex®-5 Embedded Tri-Mode Ethernet MAC (Ethernet MAC) Wrapper core on a Xilinx Virtex-5 ML505 or ML507development board. The system provides an example of how to integrate the Virtex-5 Embedded Tri-Mode Ethernet MAC and the Virtex-5 Embedded Tri-Mode Ethernet MAC wrapper using a hardware design to target the development board and a PC-based Graphical User Interface (GUI) to control the demonstration platform.

Introduction

The hardware demonstration platform demonstrates the functionality of the following Xilinx LogiCORE™ IP and Embedded Ethernet products:

- Virtex-5 Embedded Tri-Mode Ethernet MAC
- Virtex-5 Embedded Tri-Mode Ethernet MAC Wrapper v1.5
- Ethernet Statistics v2.5 core

The design demonstrates how to integrate these cores into a system, generate the required clock resources, handle the Ethernet data flow using packet FIFOs and flow control, and connect to a physical interface. Interfacing the Ethernet core to a microprocessor is not provided in this release. Data is generated in the fabric logic; the processor is used to configure the Ethernet core and does not handle data in real time.

Xilinx provides a GUI which controls the platform through an RS-232 serial cable allowing the user to experiment with different configurations of the Embedded Ethernet MAC, monitor frame information, and observe the Ethernet MAC statistics.

Requirements

Development Board

The Xilinx ML505 and ML507 development boards are the target boards in this example; however, the design can be converted to any board with suitable hardware. The minimum hardware is a Virtex-5 LXT or FXT part and a suitable Ethernet interface (GMII is used on the ML505/ML507 but can be changed to SGMII or RGMII).

A second Ethernet port is available on the ML505 and ML507 boards (using an SFP optical transceiver). This interface can be used by the demonstration platform in place of the on-board copper PHY using GMII. As shipped by Xilinx, the SFP module is not included with the board.

PC

The demonstration platform requires a PC to run the standard Windows® XP application. A spare serial port is required to connect the GUI to the development board. If a serial port is not available, a USB-to-serial converter can be used. Installation of a USB-to-serial converter is not provided in this application note.

Demonstration Platform System Design

Overview

The hardware demonstration platform consists of the following components:

- A development board with a Xilinx FPGA loaded with the hardware design
- A PC to control the hardware design using a RS-232 serial cable
- A connection partner (optionally, the design can operate in loopback mode)

Ethernet MAC Hardware Design

Figure 1 illustrates the FPGA design on the ML505/ML507, which includes a Virtex-5 Embedded Ethernet MAC Wrapper core from the CORE Generator™. This wrapper includes a soft FIFO example, clocking logic, and the logic required to interface to the PHY. GMII and PCS/PMA interfaces are used in this example, but other physical interfaces can be selected using the CORE Generator. See the *Virtex-5 Embedded Tri-Mode Ethernet MAC Wrapper Getting Started Guide*, located in the CORE Generator directory in the .zip file provided with this application note. Also included in the design (and shaded in Figure 1) is the Ethernet Statistics core from the CORE Generator.

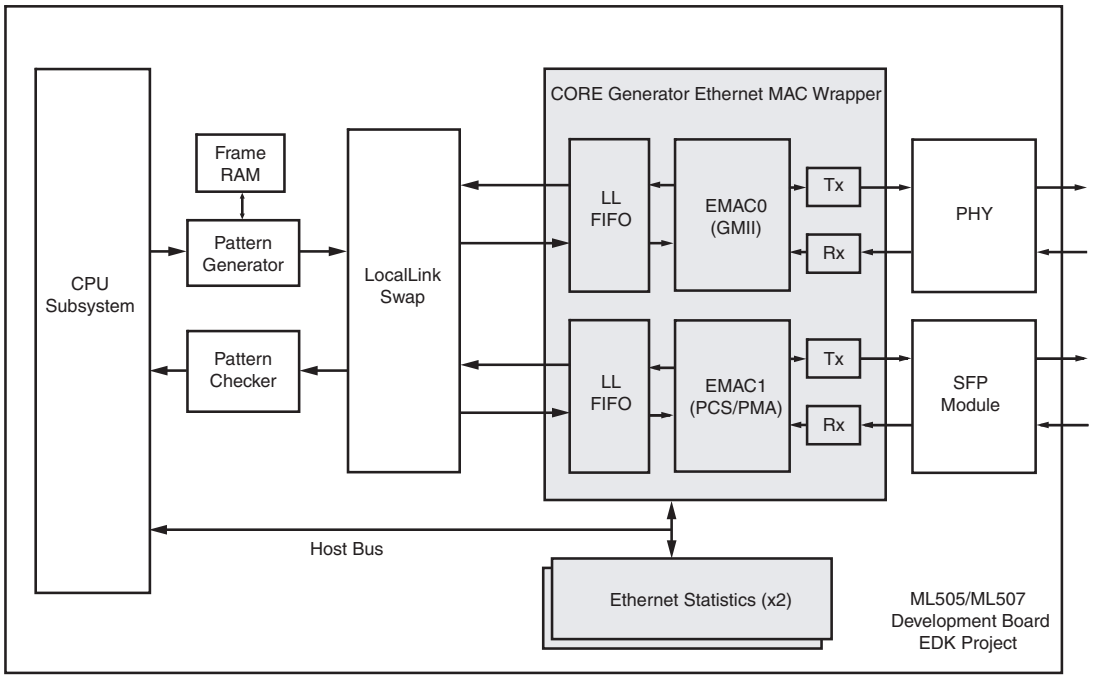

Figure 1: **Ethernet MAC FPGA Block Diagram**

Virtex-5 Embedded Tri-Mode Ethernet MAC Wrapper

The Ethernet MAC wrapper core generated by the CORE Generator is intended for use with the Virtex-5 Embedded Tri-Mode Ethernet MAC and includes the following components:

- Embedded Tri-Mode Ethernet MAC primitive.
- Soft FIFO with LocalLink interface (optional). The user can choose to use a direct interface to the Ethernet MAC in their own designs). The FIFO is based on the LocalLink FIFO design XAPP691.
- Clocking logic. See the *Virtex-5 Embedded Tri-Mode Ethernet MAC User Guide* for clock options (www.xilinx.com/support/documentation/user_guides/ug194.pdf).
- Logic required to interface to the PHY. In this example, GMII and PCS/PMA interfaces are used but other physical interfaces can be selected using the CORE Generator GUI. See the *Virtex-5 Embedded Tri-Mode Ethernet MAC Wrapper Getting Started Guide*, included in the CORE Generator directory in the .zip file provided.

The Ethernet MAC primitive consists of two independent EMACs (EMAC0 and EMAC1), both of which are used in the design to demonstrate connection to different physical interfaces. EMAC1 has a GMII interface and EMAC0 includes the optional PCS/PMA logic for direct connection to a RocketIO™ GTP transceiver, which is then connected to a SFP socket for an optical transceiver on the ML505/ML507. Pinout constraints on the board dictate the RocketIO and connection options for the PCS/PMA interface. See the *Virtex-5 Embedded Tri-Mode Ethernet MAC User Guide* for detailed information about the PCS/PMA logic.

The Ethernet MAC wrapper core was generated using the options in the XCO file included in the CORE Generator directory of the .zip file provided with this application note.

Statistics Counters

The Ethernet Statistics core, used to provide statistical counters (shaded in Figure 1) for the EMAC cores, are CORE Generator instances of the Ethernet Statistics core, provided with an example design to simplify connection to the Embedded Ethernet MACs. The Ethernet Statistics core is generated using the options in the XCO file included in the CORE Generator directory of the .zip file provided with this application note.

Physical Interfaces

As defined in "Virtex-5 Embedded Tri-Mode Ethernet MAC Wrapper" above, the Ethernet MAC wrapper, generated by the CORE Generator, provides the logic for the physical interfaces. The GMII interface of EMAC1 is connected to an external 10/100/1000Base-T PHY (Marvell Alaska 88E1111) included on the ML505/ML507. The PHY provides loopback capability to enable the demonstration platform to function without connected to a second unit.

The PCS/PMA interface of EMAC0 is connected internally to a RocketIO transceiver, which is connected to the SFP socket on the ML505/ML507. The SFP optical module is not provided with the board. The GMII interface can be used for all demonstration functions if a transceiver and link partner are not available.

Clock Management

Clock generation for the design is provided by the Ethernet MAC wrapper. A 125 MHz reference clock (available in the ML505/ML507 using the onboard PLL) is used as the input clock to the clock management logic and RocketIO transceiver. The clock management logic uses this 125 MHz reference clock to generate an output clock for the MAC transmitter logic at 125 MHz, 25 MHz, or 2.5 MHz, as defined by the Embedded MAC operating speed. The clocking management logic also provides phase-shifting abilities to correct the receiver clock from the PHY to adjust for delays across the receiver interface. When the physical interface is put into loopback, the clock management logic sets the receiver clock input to the transmitter clock.

LocalLink Swap

The LocalLink Swap module provided allows the LocalLink interface from the pattern generator to be connected to either EMAC0 or EMAC1. This module also provides the loopback function that connects the Rx FIFO to the Tx FIFO so the user can inject frames at the physical interfaces and see them echoed back. Optionally, the loopback function can swap the source and destination addresses of the Ethernet frame to provide a basic *ping* type function.

Microprocessor System

Figure 2 illustrates the microprocessor system, which is based on a soft MicroBlaze™ v7.10d processor and PLB v4.6-based subsystem containing 16k RAM, UART (uartlite), interrupt controller, LMB (for block RAM connection), GPIO, and custom peripherals for connecting the MACs through the LocalLink interface.

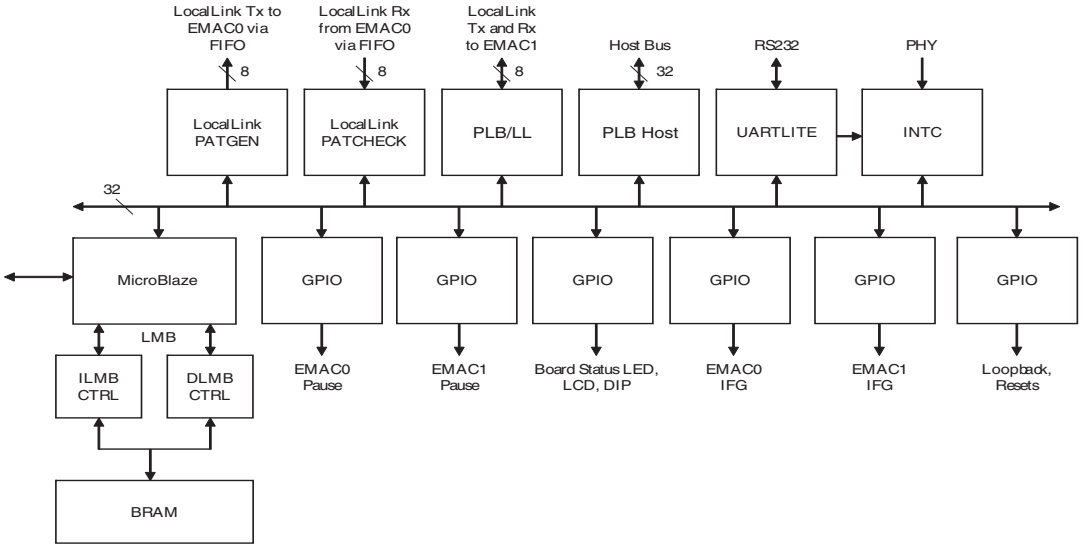

Figure 2: **Microprocessor System Block Diagram**

Custom Peripherals

- **Pattern Generator** (plb_ll_patgen). The Pattern Generator peripheral holds all the logic required to access the client side transmit data stream of the Ethernet MAC core, allowing data to be written to the transmitter-side FIFO. The patterns are stored in dedicated block RAM which can be loaded by the processor and read by logic and the data passed to the FIFO through the LocalLink interface.
- **Pattern Checker** (plb_ll_pat_chk). The Pattern Checker peripheral logic, when configured in the user mode, uses the local link interface to access the client side received data from the Ethernet MAC. The pattern checker must be configured in user mode (the default mode) if one wishes to read the data received. The pattern checker can be configured in auto mode to run line speed tests. In the auto mode, the pattern checker verifies the received data for pre-determined sequence of data bytes.
- **PLB/LL** (plb_ll). The pattern generator is connected to the one EMAC selected by the GUI. The unused EMAC is connected to the PLB v4.6 through a simple byte wide interface. It is not used by the GUI or demonstration platform.
- **Host** (plb_host). The Host peripheral allows the Management Interface of the Ethernet MAC core to be accessed. Through this interface, the MAC configuration can be read and

modified, the statistics counters can be read, and the MDIO transactions can be made to control the physical layer.

Microprocessor Software

The microprocessor runs software that monitors the serial interface and responds to commands issued by the GUI. On startup, the microprocessor detects what design is operating on the FPGA, configures the logic on the FPGA appropriately, and then waits for and processes commands from the PC.

Setting up the Demonstration Platform

ML505/ML507 Configuration

Power Supply

Power is supplied to the ML505/ML507 from a 5V jack input.

Clock Selection

The demonstration platform uses the on-board clock synthesizer to provide the 125 MHz differential reference clock. SW6 - SUPR CLK FREQ (close to the Compact Flash socket on the bottom of the board) must be set as defined in Table 1.

Table 1: **DIP Switch Settings for Super Clock Generator (SW6)**

Switch	Setting
1 (N0)	OFF
2 (N1)	OFF
3 (N2)	ON
4 (M0)	ON
5 (M1)	ON
6 (M2)	OFF
7 (SEL1)	ON
8 (SEL0)	OFF

DIP and Jumper Settings

SW3 is used to control the FPGA configuration. Table 2 defines the DIP switch settings required to enable both SystemACE and JTAG download. SW8 is not used by the design and the setting is not critical.

Table 2: **DIP Switch Settings for Configuration (SW3)**

Switch	Setting
1	OFF
2	OFF
3	OFF
4	ON
5	OFF
6	ON

Table 2: **DIP Switch Settings for Configuration (SW3) *(Continued)***

Switch	Setting
7	OFF
8	ON

The Jumpers defined in Table 3 should be installed. See the *ML505/ML506/ML507 Evaluation Platform User Guide* (UG347) for detailed information about the function of each jumper. Any jumper not described in the table should be left open.

Table 3: **Jumper Settings**

Jumper	Setting
J9	Jumper between pins 1 & 2 and between pins 3 & 4
J14	ON
J17	Jumper between pins 1 & 2
J20	Double Jumper between pins 1 & 2
J21	Jumper between pins 1 & 2
J22	Jumper between pins 1 & 2
J23	Jumper between pins 1 & 2
J56	ON
J62	Jumper between pins 1 & 2
J63	Jumper between pins 1 & 2
J81	Jumper between pins 1 & 2
J82	ON

RS-232

Connect a Null-Modem cable between the RS-232 port on the board, the 9-pin DIN plug, and an additional serial port on the PC. (This is used to connect to the PC.)

LCD

The LCD provides basic information about the status of the MicroBlaze software.

On-board Tri-Speed PHY

The ML505/ML507 has a Marvell Tri-speed PHY built onto the board. This is connected to the MAC in the demonstration platform. A 1-Gigabit compatible RJ45 patch lead should be connected to the PHY. A crossover cable is required for direct connection to another Gigabit Ethernet device; otherwise, use a normal cable for connection to a Gigabit Switch.

FPGA Configuration

There are two ways to configure the FPGA:

- Using JTAG (using Platform Studio™ or iMPACT) to download the bitfile included with the application note source files. The bitfile is created using Platform Studio and can be downloaded by opening the Platform Studio project (`system.xmp`) included with the application note or by using iMPACT to download the file from the implementation

directory. Two bitfiles are created by platform studio `system.bit` and `download.bit`. Use `download.bit` containing the MicroBlaze software.

- Using a Compact Flash card and the SystemAce file included with the application note source files. A SystemAce file is included and instructions about how to create it are included in "Building the Hardware," page 19. The .ace file should be copied to a Compact Flash card using a suitable writer, after which the CF card can be plugged in to the ML505/ML507. The SystemACE controller on the board configures the FPGA on power up.

Graphical User Interface

The GUI lets the user control the Ethernet MAC wrapper core in the demonstration platform and the other components within the design, and is comprised of the following:

- Serial Port Selection and Interface Selection dialog boxes, displayed on startup
- Main screen, which contains several tabs for setting design options
- MAC Statistics Screen

Installing and Running the GUI

1. Extract the project files from the Application Note .zip file into a new directory on the PC (for example, `c:\xapp957`).

   Note: Do not download the demonstration platform to a directory path containing spaces; this causes errors when running the GUI.
2. Load the hardware design onto the FPGA.
3. Navigate to `c:\xapp957\code\pc\rtf\` (if you saved the zip file contents to `c:\xapp957`).
4. Double-click xapp957.exe to start the GUI.

Serial (COM) Port Selection

Figure 3 displays the Serial Port selection dialog box, used to select the COM port to communicate with the board.

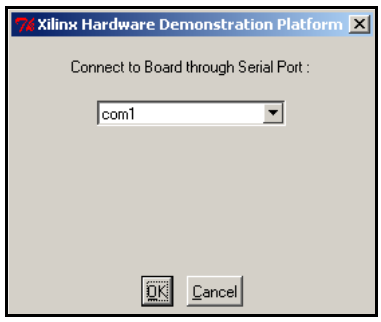

Figure 3: **Serial (COM) Port Selection Screen**

To select a port:

1. Select the appropriate COM port from the drop-down menu.
2. Click OK and wait for the link to be established with the board. After connection, the Interface Selection dialog box will appear.

   If a link is established with the board, the board and Ethernet MAC type will be displayed at the bottom of the screen.
3. If a connection to the board cannot be established, an error message will appear at the bottom of the main screen. Verify that the board is turned on, the FPGA is configured with

the correct bitfile, the serial cable is connected and is the correct type, and then restart the GUI.

Ethernet Interface Selection

Figure 4 displays the Interface Selection dialog box, which appears after selecting a COM port from the Serial Port dialog box.

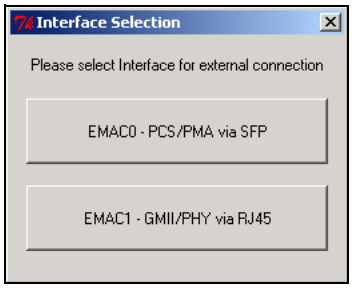

Figure 4: **Interface Selection**

To select an interface:

Click a physical interface. If the demonstration platform is to be used in loopback, either interface can be selected.

Main Screen

Figure 5 displays the main GUI screen.

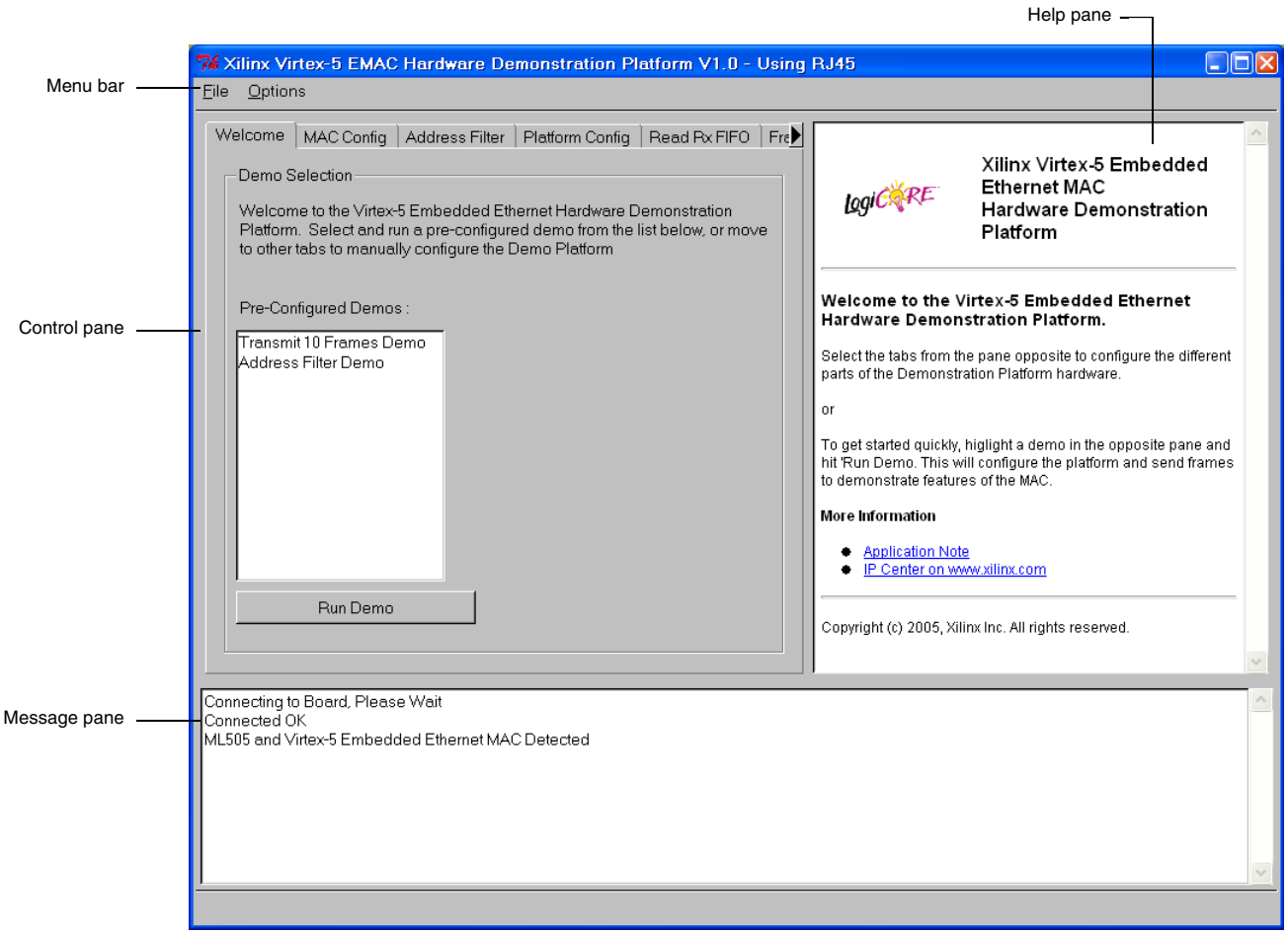

Figure 5: **Main Screen**

The Main Screen consists of four areas:

Menu Bar

A standard Windows toolbar across the top of the screen.

Control Pane

The control pane at the left side of the window displays several tabs for selecting design options:

- Welcome
- MAC Configuration
- Platform Configuration
- Read Rx FIFO
- Frame Editor
- Address Filtering
- PHY Control

Help Pane

Displays context-sensitive help at the right side of the window.

Message Pane

Displays debug and status messages at the bottom of the window.

Welcome Tab

The Welcome Tab (Figure 6) is used to run pre-configured demos. The available pre-configured demos are Transmit 100 Frames or Address Filter Demo.

To run a demo:

Click a selection in the Pre-Configured Demos area of the dialog box, then click Run Demo. After the demo runs, a message appears in the Message pane.

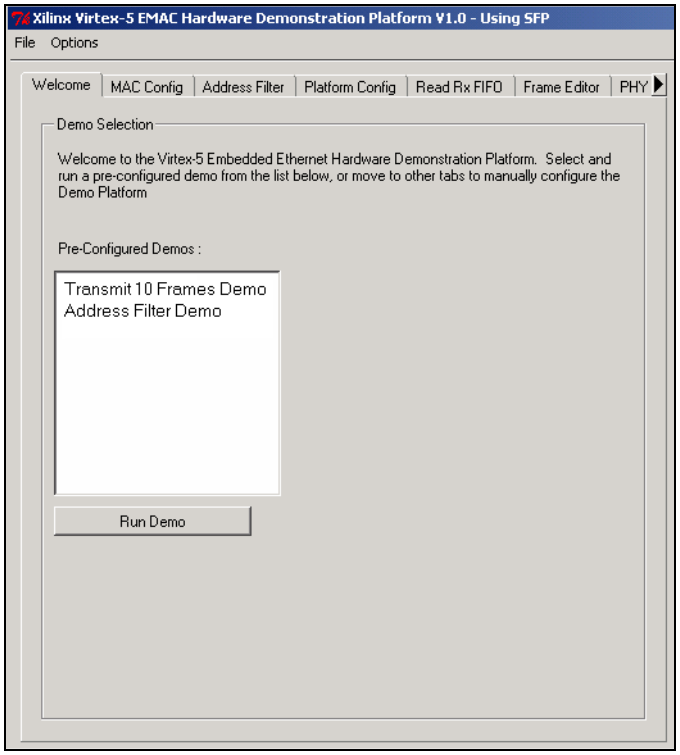

Figure 6: **Control Pane**

MAC Config Tab

The MAC Config Tab (Figure 7) modifies the configureable features of the Ethernet MAC cores. When the FPGA is configured, this screen displays the default configuration for the Ethernet MAC core.

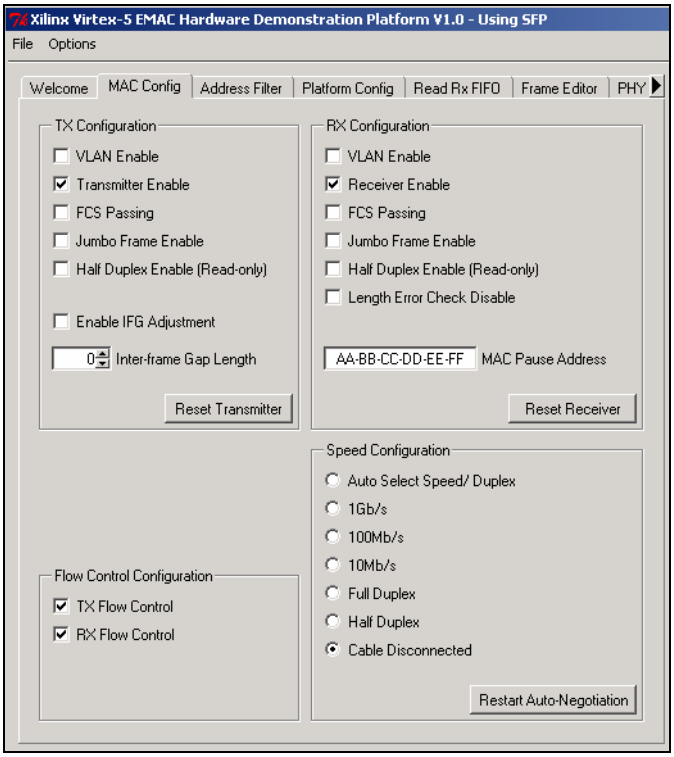

Figure 7: **MAC Configuration Tab**

Transmitter Configuration

TX (Transmitter) configuration options:

- **VLAN Enable**. When selected, enables transmission of VLAN frames.
- **Transmitter Enable**. Enables the transmitter to send frames.
- **FCS Passing**. Transmitter generates and appends the FCS to the frame.
- **Jumbo Frame Enable**. Enables transmission of jumbo frames.
- **Half Duplex Enable (Read-only)**. Enables half-duplex operation mode.
- **Enable IFG Adjustment**. Select this option to subsequently enter an inter-frame gap length.
- **Inter-Frame Gap Length**. When Enable IFG Adjust is selected, the value in this field is the inter-frame gap length used between frames while transmitting.

Receiver Configuration

RX (Receiver) configuration options:

- **VLAN Enable**. When selected, enables reception of VLAN frames.
- **Receive Enable**. Enables the receiver to receive Ethernet frames.
- **FCS Passing**. Receiver passes the received FCS to the client.
- **Jumbo Frame Enable**. Enables reception and passing of jumbo frames.
- **Half Duplex Enable (Read-only)**. Enables half-duplex operation mode.

- **Length Error Check Disable**. Received frames are not checked for length and CRC errors.
- **MAC Pause Address.** The pause address used in the generated pause frames.

Flow Control Configuration

Transmit and Receive Flow Control can be enabled or disabled by the user, allowing observation of the flow control functionality.

Speed Configuration

The speed configuration for the Ethernet MAC demonstration platform does not provide access to the speed registers in the MAC core; different operation speeds can be selected. The speed configuration selected depends on whether loopback is selected.

- **Auto Select Speed/Duplex**. Allows auto-negotiation to negotiate and select the speed and half-duplex / full-duplex mode.
- **1 Gb/s**. Sets the advertised link speed to 1 Gb/s.
- **100 Mb/s.** Sets the advertised link speed to 100 Mb/s.
- **10 Mb/s**. Sets the advertised link speed to 10 Mb/s.
- **Full-Duplex**. Sets the advertised mode to full-duplex mode of operation.
- **Half-Duplex**. Sets the advertised mode to half-duplex mode of operation.
- **Cable Disconnected**. Indicates that no cable is connected.
- **Restart Auto-Negotiation**. Causes the EMAC to initiate auto negotiation. Auto negotiation is automatically initiated by EMAC on power on, reset, or loss of link.

Loopback Mode Setting

When the interface is set to loopback mode (see "Platform Configuration Tab," page 14) configure the speed in the following way:

1. Select Auto Select Speed/Duplex to use the MAC core at the fastest speed or select a specific speed and duplex combination.
2. Click Restart Auto-Negotiation. The GUI speed/duplex options are deasserted until the MAC has been configured for the new speed.

Note: As the physical side is being looped back, only full-duplex operation can be used. Half-duplex mode fails.

External Device Mode

When an external device is being used and the physical interface is selected to use the external device, configure the speed in the following way:

1. Select either Auto Select Speed/Duplex to use the MAC core at the fastest speed, or select a specific speed and duplex combination.
2. Click Restart Auto-Negotiation. The GUI speed/duplex options are deasserted until the MAC has been configured for the new speed.

If Auto-Select Speed/Duplex is selected, clicking Restart Auto-Negotiation forces the PHY to advertise all speed/duplex combinations or to advertise only the requested speed/duplex combination. The PHY is then set to restart Auto-Negotiation with the external device.

After completion of Auto-Negotiation, the demonstration platform reads the PHY registers to determine the link speed/duplex and then appropriately configures the Ethernet MAC core. The GUI speed/duplex options are then reasserted to show the operating speed/duplex. If an error occurs during Auto-Negotiation, it is displayed in the message pane.

Note: If the link to the external device goes down at any time, the GUI speed/duplex options disappear from the speed configuration and return when the link is re-established.

Address Filter Tab

The Address Filter tab (Figure 8) controls address filtering on the receiver of the Embedded Tri-Mode Ethernet MAC. Address filtering can be enabled or disabled and addresses can be written into the table in Hex using the Address Filter tab.

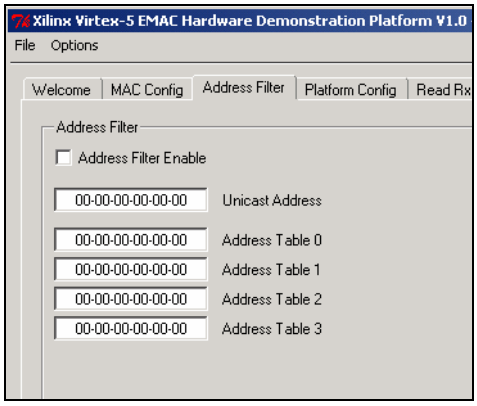

Figure 8: **Address Filter Tab**

If an external device is connected to the demonstration platform and is intended to receive frames from the device, the address filter must be configured with the MAC address the external device is using as a Destination Address. Otherwise, disable address filtering.

If loopback is selected on the Platform Configuration tab, the address filter should be configured with the address set in the frame editor, the destination address used by the pattern generator (11-22-33-44-55-66), or disabled.

Platform Configuration Tab

The Platform Configuration tab (Figure 9) controls the data sources in the demonstration platform.

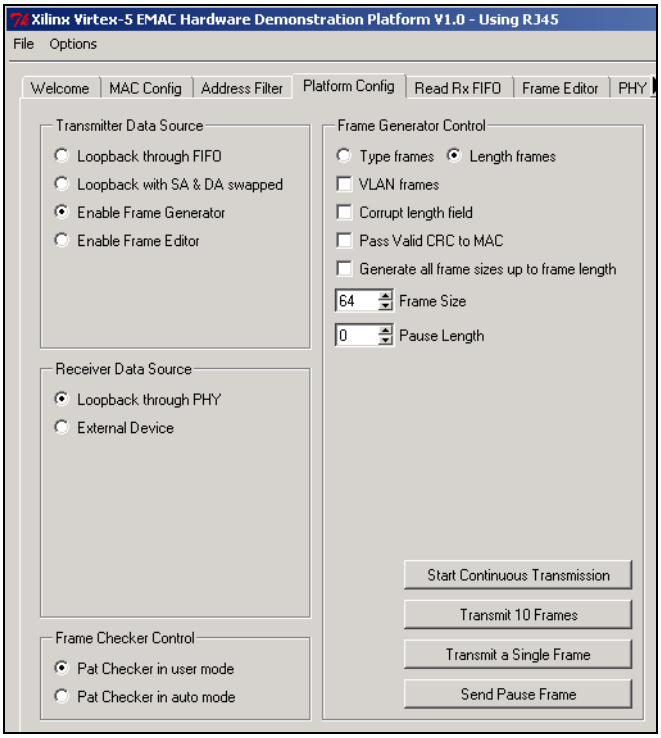

Figure 9: **Platform Configuration Tab**

Transmitter Data Source

The client side transmitter interface of the MAC can be connected to one of three data sources:

- **Loopback through FIFO** and **Loopback with SA & DA Swapped** (source address and destination address). Connects the data path from the MAC client side receiver to the transmit path. Data passes from the MAC receiver to the receive FIFO, then directly to the transmit FIFO and finally to the MAC transmitter.

  If Loopback with SA & DA Swapped is selected, the first 6 bytes of each frame are switched with the following 6 bytes, switching the source and destination address. This allows an external device to send frames to the demonstration platform and receive correctly addressed frames back.
- **Enable Frame Generator**. Connects the transmit FIFO to the hardware pattern generator. Options for controlling the pattern generator are also set in this tab and are described in "Frame Generator Control."
- **Enable Frame Editor**. Connects the transmit FIFO to the RAM based frame editor, allowing the user to enter and transmit any frame using the GUI. The Frame Editor is controlled in the Frame Editor tab; see "Frame Editor Tab," page 17.

Receiver Data Source

- **Loopback through PHY**. Connects the transmitter to the receiver through the PHY or PCS/PMA logic.
- **External Device.** Configures the receiver for loopback connected to an external device.

Frame Generator Control

The Frame Generator generates various frame types and lengths.

- **Type Frames**. The pattern generator inserts hex 0800 in the length/type field of each frame.
- **Length Frames**. The pattern generator inserts the length of the data contained in each frame into the length/type field of each frame.
- **VLAN Frames**. The pattern generator inserts a VLAN tag into each frame. The total length of each frame is unchanged but the length field is adjusted to indicate less data in each frame.
- **Corrupt Length Field**. The length field is set incorrectly, resulting in an errored frame.
- **Pass Valid CRC to MAC**. Complete frame along with the received CRC is passed to the client.
- **Generate all Frame Sizes up to Frame Length**. The pattern generator starts transmitting 19-byte frames (padded by the MAC core) and increase the length of the frame by 1 byte. When the length is equal to the length selected for Frame Length, the length is set back to 19 and the process repeats. Use this option for transmitting frames continuously.
- **Frame Size**. Sets the total size for frames sent to the transmitter. If the value is less than 60, the MAC core pads the frame unless configured for FCS passing.
- **Pause Length**. Sets the 16-bit pause length input to the MAC core, which is used if Send Pause Frame is selected. See the *Virtex-5 Embedded Tri-Mode Ethernet MAC User Guide* (UG194) for a description of the pause transmission mechanism.

The four buttons below the Frame Generator section send frames to the MAC transmitter and trigger transmission by the pattern generator.

- **Start Continuous Transmission**. Sends frames to the MAC until the same button, (now **Stop Transmission**) is clicked.
- **Transmit 10 Frames**. Enables one-shot mode, in which ten frames are sent.
- **Transmit a Single Frame**. Enables one-shot mode in which a single frame is sent.
- **Send Pause Frame**. Sends a single-pause frame from the MAC, which asserts the pause request input to the MAC. To enable these options, the Transmitter Data Source must be set to Enable Frame Generator.

PHY Registers Tab

The PHY Control tab (Figure 10) displays the information gathered by the Ethernet MAC core about the PHY (PCS/PMA or GMII depending on the selected interface) using the MDIO. On the Platform Config tab, selecting Loopback though PHY enables demonstration of the delivered design.

The other fields provide register status information only. See the *Virtex-5 Tri-Mode Embedded Ethernet MAC User Guide* (for the PCS/PMA registers), or the *8E1111 (Marvell Alaska*

Integrated 10/100/1000 Gigabit Ethernet Transceiver) Data Sheet for information about the MDIO registers.

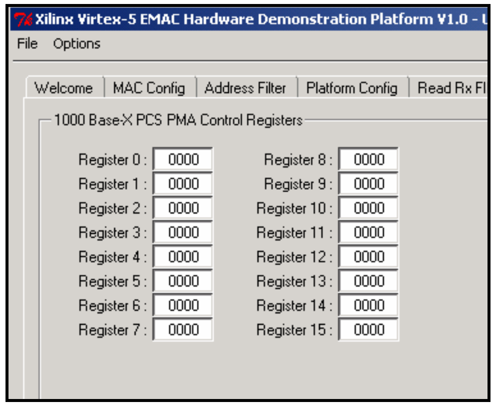

Figure 10: **PHY Registers Tab**

Read Rx FIFO Tab

The Read Rx FIFO tab (Figure 11) reads frames from the receive side FIFO received by the Ethernet MAC core. The FIFO cannot be read if the platform is configured for loopback through the FIFO.

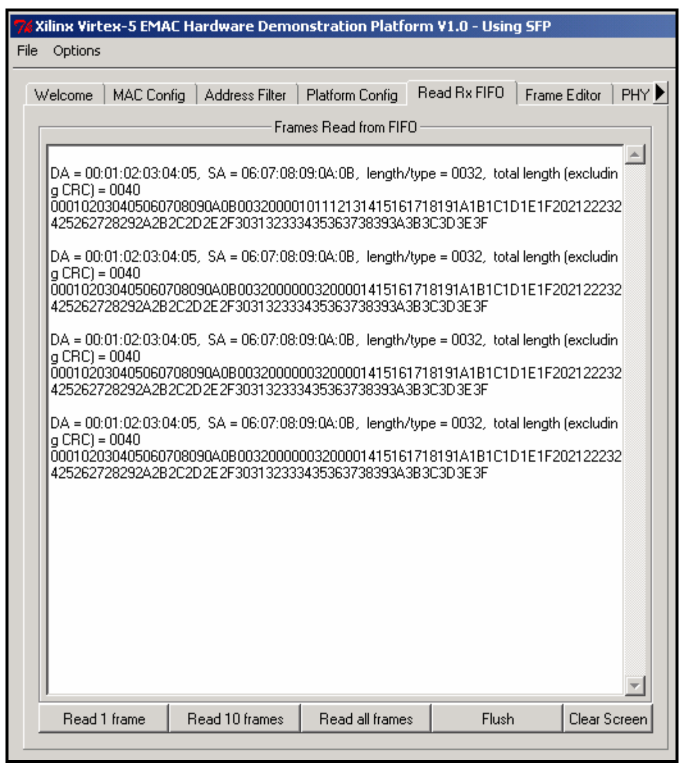

Figure 11: **Read Receive FIFO Tab**

Read frame options include:

- Read 1 Frame
- Read 10 Frames
- Read all Frames
- Flush

- Clear Screen

The FIFO stores all frames received from the MAC core until it is full. When full, the FIFO discards further incoming frames. The FIFO is 16 Kbytes deep, so it can store approximately 10 frames of 1518 bytes.

Care should be taken when mixing frame sizes. If the FIFO is nearing full and a frame larger than the remaining space is received, it is discarded. If the next frame is smaller than the remaining space, it is stored in the FIFO. This situation should not be confused with frame loss.

To read frames from the receive FIFO, flush the FIFO before initiating transmission to ensure that the FIFO has space to store the new frames. After transmission is started, the FIFO can be read. During continuous reception of frames, a snapshot of the data can be captured by flushing the FIFO, which will cause it to discard any stored frames, and store the next 16 kBytes of frame data received. The user can read a single frame or all available frames.

During continuous reception, the FIFO is unlikely to empty because data can enter the FIFO at a much faster rate than can be transmitted across the serial link to the GUI. To prevent an endless upload of frames to the GUI, a maximum of 64 frames can be read at once. To read more frames, continue clicking Read all Frames.

Frame Editor Tab

The Frame Editor tab (Figure 12) load frames into the RAM in the Pattern Generator, which can then be sent to the Ethernet MAC core for transmission. Enable Frame Editor must be selected on the Platform Config tab to enable this functionality.

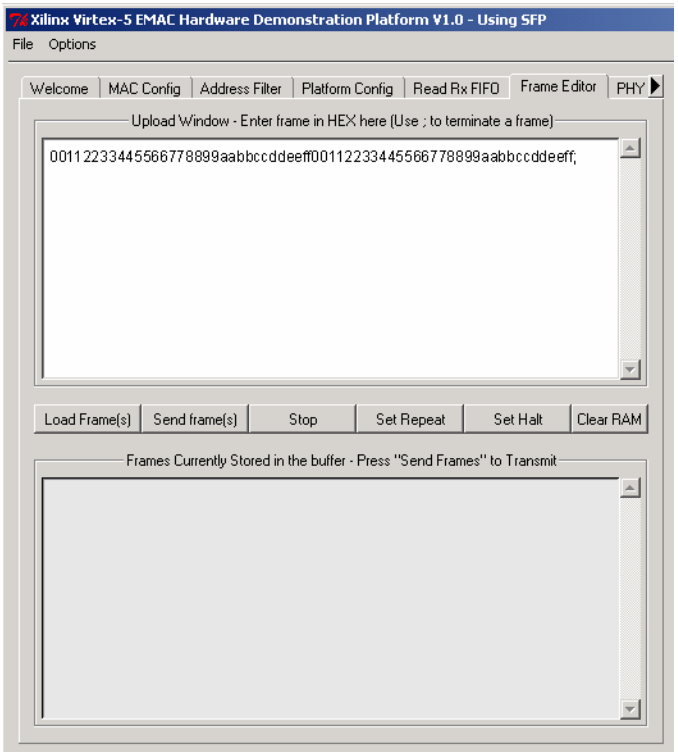

Figure 12: **Frame Edit Tab: Data Entry**

Frame data should be entered in hex in the Upload window, as displayed above. Multiple frames can be entered and uploaded at the same time.

Syntax for entering frames:

- White space and CR/LF characters are ignored

- Data must only contain the characters 01234567890ABCDEFabcdef
- Each frame must be terminated with a semicolon
- Each frame must have an even number of characters (nibbles)

Valid frame example:

```
1122334455667788990OAAbbCCddEE;
1 2 3 4 5 6 7 8;11223344; 2 35556;
```

Invalid frames example:

```
123456789; - uneven number of nibbles will give a warning
11223344rt56; - invalid hex characters
```

Example inaccuracies:

```
11223344556677889900
12345678123456789012;
```

The input is treated as a single frame. The carriage return is not treated as a frame terminate character.

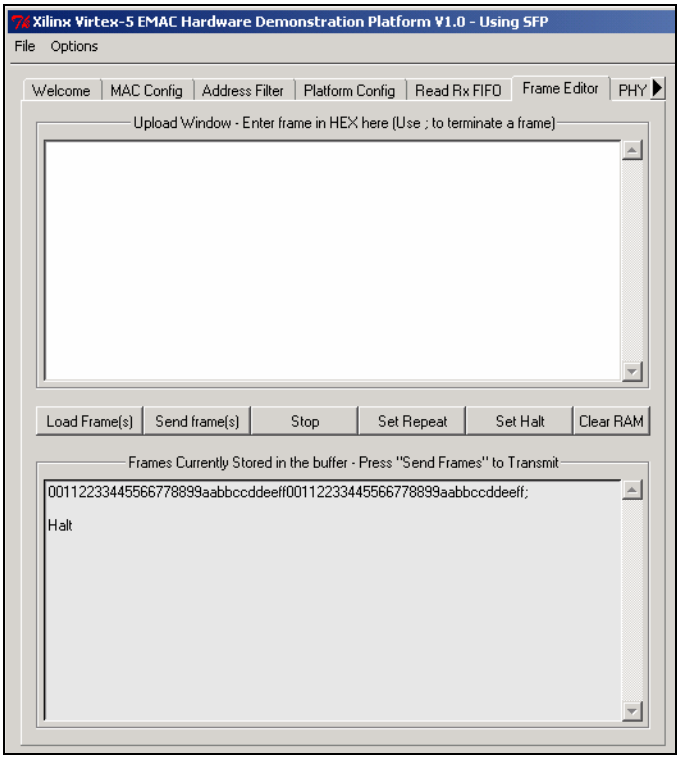

Figure 13: **Frame Edit Tab: Buffer Display**

- **Load Frames(s)**. Loads frames into the RAM. When an error is detected on an entered frame, it is highlighted to the user and the frame remains in the upload window. If the user enters three frames and the first two are valid but the third is invalid, the first two are uploaded and the third remains in the edit window. If an error occurs in the first frame, all three frames remain in the upload window.
- **Send Frames**. Starts sending frames to the Ethernet MAC core.
- **Stop**. Stops transmission of frames to the Ethernet MAC core.
- **Set Repeat**. Requests continuous transmission of frames by continuously looping through all the frames in the RAM.

- **Set Halt**. Requests that frames in the RAM are sent only once.
- **Clear Ram**. Clears the RAM.

MAC Statistics

Statistic counter values (Figure 14) are displayed by choosing Options > Show Stats.

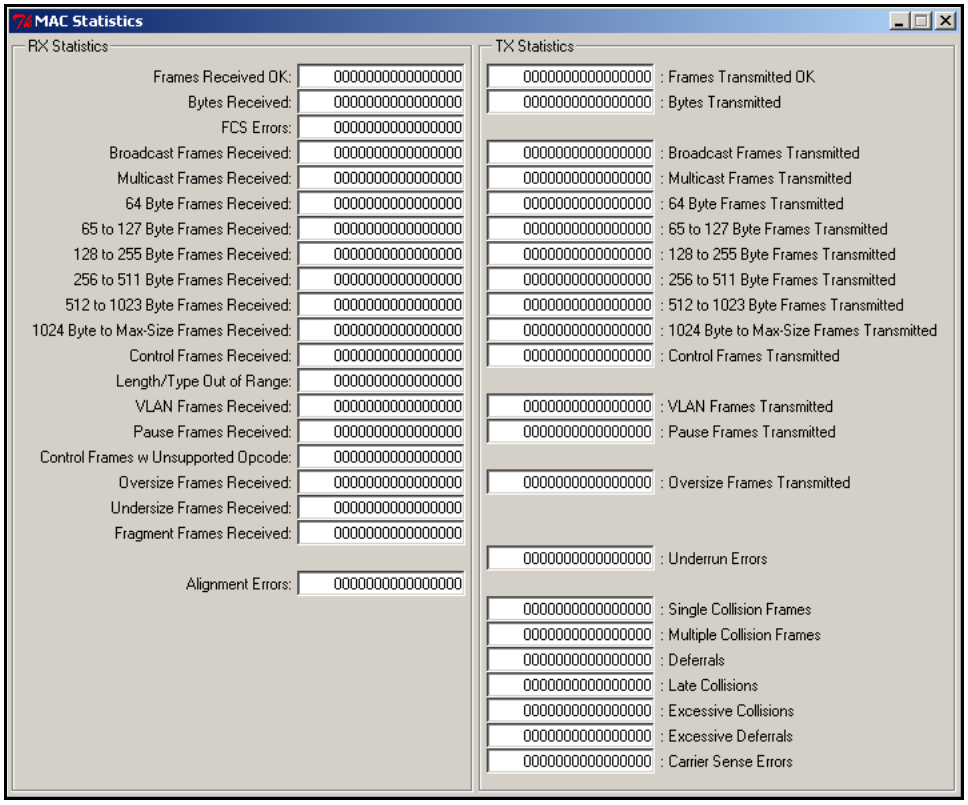

Figure 14: **Statistics Window**

Building the Hardware

A ready-to-use bitfile and SystemACE file are provided. Use the information in this section only if it is required to change the design.

The design consists of an XPS project containing the components used in the design. HDL source code and support files are provided for each component (pcore) unique to the design. Standard XPS peripherals are provided by XPS when it is installed; these are not included with the design source.

Source Code Directory Structure

The main components of the hardware design include:

- An XPS top-level project containing the MicroBlaze peripherals and software
- Multiple RTL and NGC based pcores containing the EMAC core and board specific logic

Figure 15 illustrates the organization of the source code directories. The system is built using XPS. For the ML505 board, the project is `system_lxt.xmp`, and for the ML507 board, the project is `system_fxt.xmp`. Both these projects are located in the top-level folder.

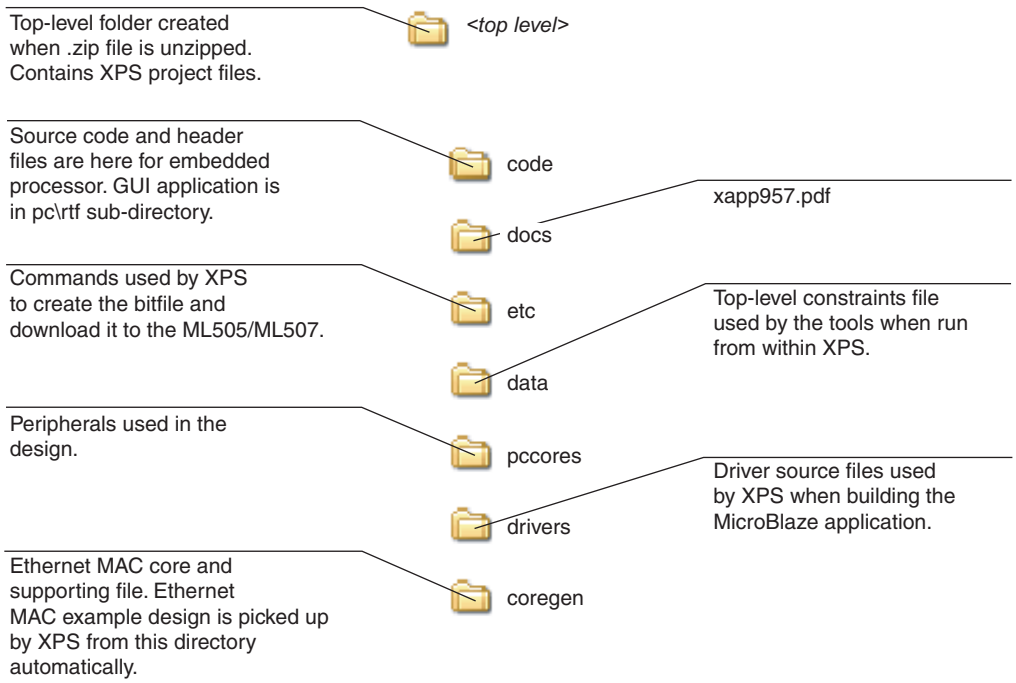

Figure 15: **Source Code Directories**

Building All Demonstration Platforms

CORE Generator IP

The Ethernet MAC wrapper and Ethernet Statistics cores are created using the CORE Generator in the CORE Generator directory. The CORE Generator output for the latest versions of both cores is included with the application note source .zip file. A CORE Generator project file and parameter files are also included in the CORE Generator directory. If either of the cores is regenerated, the same names must be used and the cores generated in the CORE Generator directory, because the XPS build process copies the source and NGC files from the CORE Generator directory each time the design is built from scratch.

XPS Project

To build the bitfile containing both hardware and MicroBlaze software, using EDK version 10.1i, do the following:

1. Using XPS, open `system_lxt.xmp` (ML505) or `system_fxt.xmp` (ML507). Although the Windows version can be used, only Linux and XP versions have been tested.
2. Choose Project > Clean all Generated Files to remove any old files.
3. From the Device Configuration menu, select Update Bitstream to build both the hardware and software.

For detailed information about the build process, see the EDK documentation, located at www.xilinx.com/ise/embedded/edk_docs.htm.

Configuration

JTAG

XPS can be used to download the bitfile directly using a suitable JTAG cable. From the Device Configuration menu, choose Download Bitstream.

SystemACE

To create a SystemACE file, open a command prompt and navigate to the implementation subdirectory created when the application note source zipfile was expanded (the implementation subdirectory is only created after XPS has run). Then type the following:

For the ML505:

```
xmd -tcl genace.tcl -jprog -hw download.bit -board ml505 -target mdm -ace xapp957.ace
```

For the ML507:

```
xmd -tcl genace.tcl -jprog -hw download.bit -board ml507 -target mdm -ace xapp957.ace
```

`xapp957.ace` can be copied from the implementation subdirectory to your Compact Flash card using an appropriate writer.

References

1. UG170, *LogiCORE IP Ethernet Statistics User Guide*.
2. UG194, *Virtex-5 Embedded Tri-Mode Ethernet MAC User Guide*.
3. UG347, *ML505/ML506/ML507 Evaluation Platform User Guide*.
4. UG340, *Virtex-5 Embedded Tri-Mode Ethernet MAC Wrapper User Guide*.

Revision History

Date	Version	Revision
1/15/07	1.0	Initial Xilinx release.
10/8/08	1.1	Updated to support the Virtex-5FXT device, the PLB v4.6 bus and EDK version 10.1.